Art Direction for the Web

Andy Clarke
A Hardboiled Web Design shot
Art direction for the web

A Hardboiled Web Design Shot by Andy Clarke
For Lemmy.

If you like to gamble,
I tell you I’m your man.
You win some, lose some,
all the same to me.

The pleasure is to play,
makes no difference what you say.
I don’t share your greed,
the only card I need is the Ace of Spades.

Ian Fraser ‘Lemmy’ Kilmister (1945–2015)
Born to lose, lived to win.
Contents

1. What art direction means  Page 21

2. One hundred years of art direction  Page 41

3. Art-directing experiences  Page 56

4. Art direction and creative teams  Page 70

5. Principles of design  Page 95

6. Directing grids  Page 129
Acknowledgements

To everyone at Smashing Magazine for their friendship and support through thick and thicker.

To the best editor in the business, Owen Gregory, for being Ronnie to my Reggie; and to Rachel Andrew for the technical editing of this book and everything she’s done for the web.

To my design hero Trent Walton for his foreword.

To Rachel Andrew (again), Jon Gold, Stephen Hay, Dan Mall, Mark Porter, David Sleight, and Rob Weychert for agreeing to be interviewed for this book and sharing their experiences.

To my favourite type foundry Dalton Maag. I made every design in this book using only their fabulous fonts.

To my friends; John Allsopp, Paul Boag, Ben Buchanan, Voltaire Carlos, Fiona Chan, Dan Davies, Brendan Dawes, Richard Eskins, Petra Gregorova, Scott Gruber, Jon Hicks, Leigh Hicks, Dylan Jones, Luci McCullough, Drew McLellan, Mandy Michael, Harry Roberts, Mike Sharp, Joe Spurling, Tim Shuttleworth, Jared Spool, and Jesse Yuen. And to Dr. Alex Clarke for typesetting this book and making me the proudest dad.

Finally, to my wife, Sue. 2019 is our 30th wedding anniversary. She famously threatened to hunt down and kill the next person who asked me to write a book, but without her smart thinking and encouragement, this one wouldn’t have been written.
About the author

Andy Clarke is a well-known designer, design consultant, and mentor. He's been called plenty of things since he started working on the web. His ego likes terms such as “Ambassador for CSS,” “industry prophet,” and “inspiring,” but he's most proud that Jeffrey Zeldman once called him a “triple-talented bastard.”

With his wife Sue, Andy founded Stuff & Nonsense in 1998, where they've helped companies around the world to improve their designs by providing consulting and design expertise.

Andy's written several popular books on website design and development, including Hardboiled Web Design: Fifth Anniversary Edition (Smashing Magazine, 2015), Hardboiled Web Design (Five Simple Steps, 2010), and Transcending CSS: The Fine Art Of Web Design (New Riders, 2006). He's a popular speaker, and gives talks about art direction and design all over the world.

Andy tweets as @malarkey and really, really loves gorillas.

1 https://stuffandnonsense.co.uk
2 https://twitter.com/malarkey
About the reviewers

Rachel Andrew lives in Bristol, UK. She's one half of web development company edgeofmyseat.com, the company behind Perch CMS. She's also editor-in-chief of Smashing Magazine, an Invited Expert to the W3C on the CSS Working Group, and a Google Developer Expert. Her day-to-day work can include anything from product development to DevOps to CSS, and she regularly writes about all of them.

Owen Gregory is a professional editor, copy editor and proofreader based in Birmingham, UK. Owen has experience of working with small, independent, and digital publishers, from manuscript to print and digital editions. His particular expertise encompasses all aspects of front-end web design and development, and he's worked with many well-known authors from the web industry.

3 https://rachelandrew.co.uk
4 http://fullcreammilk.co.uk
What this book’s about

Art direction has been part of advertising and print design for over 100 years, but on the web art direction is rare and there have been few meaningful conversations about it. This might be because we’ve been fixated on designing digital products. It may be because we’ve been preoccupied with making websites responsive. It might simply be that many designers and developers were never taught about art direction. In this book I’m going to explain art direction, what it means, why it matters, and who can do it.

Art direction can help you improve engagement and encourage customer retention. It connects your brand with an audience, improves conversions and brings your customers’ journeys to life. In this book, I’m going to show you how. I’ll also explain how art direction can maintain brand values and design principles by connecting touch points across marketing, product design, and websites.

This book is for web designers and developers who want to understand art direction and how to make it work for digital products and websites. It’s for businesses that want to make designs which engage audiences, and for experienced art directors and designers who work in print, who need to understand how to take their talent to the web.

What I describe is as relevant to people who design websites running on platforms like Shopify or Squarespace, as for those who handcraft every aspect of a digital experience. It’s as appropriate for people who design complex and multifaceted digital products as for those who create websites to help market them.
I’ve written for developers who want to collaborate with designers more effectively by teaching design principles so they can implement designs on any platform, from template-driven systems like Wix or WordPress to handwritten code. The art direction principles I teach also apply as much to people who use libraries like Bootstrap as they do to those who start every project by defining a bespoke grid.

I’ve written for other writers too, people who want their words to be more than—as my friend Brendan Dawes\(^5\) once said—“filling for rectangles.”

I’ve written for people who lead marketing or product teams, or run businesses, and who want their stories to be more effective at communicating, persuading, and selling.

I also wrote *Art Direction for the Web* for me because, quite frankly, I’m weary of seeing websites that feel soulless and look identical to one another. I hope this book can help change that.

The content of this book is based on my twenty years’ experience of working with clients, plus the expertise of the art directors and designers I interviewed. I learned an incredible amount from talking with them, and I hope that through this book you will too.

\(^5\) [http://brendandawes.com](http://brendandawes.com)
Foreword by Trent Walton

N early ten years ago, my phone rang, and it was Andy Clarke. My Paravel cohorts and I had recently published an infographic-heavy website devoted to our love of cinema. We pulled out all the stops, and we were pretty damn proud of our creation. Having pushed ourselves with layout, typography, and illustration to see what we were capable of, we were still unsure if our peers would think it was, well, cool. Would they get it? Would it resonate?

For years, Andy had been writing, speaking, and teaching about design and CSS—one of those online fixtures who seem to have been a key part of the web community from the beginning. I deeply admired his work, which had (and still is) centred on his willingness to experiment with new tech, comprehend its capabilities, and ultimately produce some enviously clever implementation of it.

When I answered that phone (what seems like a lifetime ago in web years), Andy asked to use our site as an example in a book he was working on. I was honoured and felt validated because somehow the thing we built struck a chord with someone we aspired to emulate. My belief in the power of the web as a medium was reinforced. Not only is it able to transmit information exceptionally well, but when we are intentional about design choices (regarding layout, typography, imagery, copy, and so on) when we build, we create a web that can inspire, excite, and connect people too.

Now, with Art Direction for the Web, Andy provides a framework for harnessing the web’s potential in this way. With historical context and real-life examples, Andy inspires each of us to be more purposeful about the choices we make. And true to form, he follows up all that inspiration with demos and the practical knowledge needed to see our ideas manifest online.
Art Direction for the Web is as close as we’ll get to seeing the web through Andy’s eyes: a web that has soul, style, and purpose—just like the man himself.

About Trent Walton

What can I say about Trent Walton? Yes, he’s part of the three amigo web design studio Paravel Inc., based in Austin, Texas. Yes, he’s been responsible for some of the best examples of responsive web design at scale, including his work for Microsoft. Yes, he’s written some of the most thoughtful pieces about designing responsively. Yes to all this. He’s also one of the humblest people I know. He’s my Wild West hero. Let’s just say that.

6 https://trentwalton.com
7 https://paravelinc.com

Introduction

Study these two stories for a moment. I’ll wait.
How DEA Agents Took Down Mexico’s Most Vicious Drug Cartel

...And how this helped give rise to the criminal empire of Chapo Guzmán

**CASE STUDY AND POLICIES**

January 20, 1990

For 14 months, the first thing Dave Herrod, a special agent with the Drug Enforcement Administration, did every morning was boot up his laptop and begin tracking a 45-foot yacht with black boxes painted on the hull.

In the summer of 2005, the DEA had dismantled a marijuana ring in which members of a Mexican drug cartel known as the Arellano Felix Organization were buying a yacht in California. Herrod and his colleagues studied the classified ads in yacht magazines and determined that the boat Holiday was the boat the Arellano Felix Organization was using to transport drugs. DEA agents had managed to get on board and install tracking devices before the sale went through. That’s when Herrod started tracking the boat on his laptop.

Since the early 1990s, the Arellano brothers—the inspiration for the Ochberg brothers in the movie *Traffic*—had controlled the flow of drugs through what was perhaps the single most important point for Black cocaine in the world: the border crossing from Tijuana to San Diego. With the Arellanos’ success derived from its production and distribution. The cartel employed a crew of “narcotics” who would hang victims from structures like trees and leave them to die in the heat. Meanwhile, the Spanish word for a traditional Mexican wine, the AFOs’ euphemism for a method of hiding highprofile victims—in this case, one of the crew’s heads—was often buried in a barrel of heavy concrete and left for 24 hours until only their teeth were left, then thrown down the trash.

**CASE STUDY AND POLICIES**

DEVA, Deals and the DEA

Why Chapo Guzmán was the biggest winner in the DEA’s longest running drug cartel case

daniel spain, ProPublica

December 15, 2012

This story was co-published with The Atlantic.

For 14 months, the first thing Dave Herrod, a special agent with the Drug Enforcement Administration, did every morning was boot up his laptop and begin tracking a 45-foot yacht with Dock Holiday painted on the stern.

In the summer of 2005, the DEA had dismantled a marijuana ring in which members of a Mexican drug cartel known as the Arellano Felix Organization were buying a yacht in California. Herrod and his colleagues studied the classified ads in yacht magazines and determined that the boat Holiday was the boat the Arellano Felix Organization was using to transport drugs. DEA agents had managed to get on board and install tracking devices before the sale went through. That’s when Herrod started tracking the boat on his laptop.

Since the early 1990s, the Arellano brothers—the inspiration for the Ochberg brothers in the movie *Traffic*—had controlled the flow of drugs through what was perhaps the single most important point for Black cocaine in the world: the border crossing from Tijuana to San Diego. With the Arellanos’ success derived from its production and distribution. The cartel employed a crew of “narcotics” who would hang victims from structures like trees and leave them to die in the heat. Meanwhile, the Spanish word for a traditional Mexican wine, the AFOs’ euphemism for a method of hiding highprofile victims—in this case, one of the crew’s heads—was often buried in a barrel of heavy concrete and left for 24 hours until only their teeth were left, then thrown down the trash.

**CASE STUDY AND POLICIES**

DEVA, Deals and the DEA

Why Chapo Guzmán was the biggest winner in the DEA’s longest running drug cartel case

daniel spain, ProPublica

December 15, 2012

This story was co-published with The Atlantic.

For 14 months, the first thing Dave Herrod, a special agent with the Drug Enforcement Administration, did every morning was boot up his laptop and begin tracking a 45-foot yacht with Dock Holiday painted on the stern.

In the summer of 2005, the DEA had dismantled a marijuana ring in which members of a Mexican drug cartel known as the Arellano Felix Organization were buying a yacht in California. Herrod and his colleagues studied the classified ads in yacht magazines and determined that the boat Holiday was the boat the Arellano Felix Organization was using to transport drugs. DEA agents had managed to get on board and install tracking devices before the sale went through. That’s when Herrod started tracking the boat on his laptop.

Since the early 1990s, the Arellano brothers—the inspiration for the Ochberg brothers in the movie *Traffic*—had controlled the flow of drugs through what was perhaps the single most important point for Black cocaine in the world: the border crossing from Tijuana to San Diego. With the Arellanos’ success derived from its production and distribution. The cartel employed a crew of “narcotics” who would hang victims from structures like trees and leave them to die in the heat. Meanwhile, the Spanish word for a traditional Mexican wine, the AFOs’ euphemism for a method of hiding highprofile victims—in this case, one of the crew’s heads—was often buried in a barrel of heavy concrete and left for 24 hours until only their teeth were left, then thrown down the trash.

Two examples of art-directing a news story about the US DEA’s longest-running drug cartel case. Left: “How DEA Agents Took Down Mexico’s Most Vicious Drug Cartel” from The Atlantic. Right: “Devils, Deals and the DEA” from ProPublica.
These pages both contain the same written copy and many of the same photographs. In the first interpretation, published in The Atlantic, information is presented in an intentionally matter-of-fact manner to make a reader think The Atlantic is balanced and reliable. After all, journalism is—or at least should be—concerned with remaining neutral.

Over at ProPublica, illustrations convey violence, with an intensity that would’ve been difficult to stomach using photography. These illustrations make people feel tense and terrified and, you know, that’s OK. Readers should experience these emotions while reading about shoot-outs and melting drug dealers in acid.

The process of evoking those feelings is called art direction.

These stories were art-directed with two very different goals in mind, and the decisions taken depended on factors which I explore in this book.

Now, I realise not everyone designs news stories. Some of us help charities raise money for important causes, and we need people to feel affinity with a cause. Some of us work for governments where the aim is to inform people and sometimes reassure them. Some of us design digital products and focus on the experience of using them. Others make websites which sell physical products, like those fancy watches I dream of spending my money on.

8 https://smashed.by/atlanticidea
9 https://smashed.by/propublicadea
Selling watches needs more than developing a frictionless e-commerce flow; it’s also about making people want the product. (Courtesy of Panzera.)

The pleasure of buying a product like an expensive watch comes from how we feel while purchasing it, not just owning the item. Creating positive experiences by stirring people’s emotions and motivating them to buy, that’s art direction too.

This is a book about why art direction matters and how you can art-direct compelling and effective experiences across devices and platforms. I’ll explain how to use design principles and techniques, and then implement art-directed designs using the most up-to-date technologies.
The madness of Frankie Fraser

Frank Fraser ("Fraser"), better known as "Frankie" Fraser, was a British fascist, race and English nationalist and criminal who spent 12 years in prison for various offenses. In the early 1960s, Fraser met Charlie and Eliza Tchurner at the National Front meeting held at the Knyvet House, according to him, helped him achieve the "Great Front Highest" by "Invincible". He later claimed to have met Hitler and was present at the Nuremberg rallies, but it is unclear if these claims are true. Fraser was a member of the National Front in the 1960s and was sentenced to 12 years in prison for various offenses, including murder, attempted murder, and conspiracy to commit murder. He was released in 1972 after serving 12 years in prison.

The Richardson Gang

The Richardson Gang was an English crime group based in South London, England, in the 1960s. They were known for their violent and brutal tactics, including murder, armed robbery, and extortion. The group was led by Reginald Richardson, who was sentenced to life imprisonment for the murder of a police officer. The Richardson Gang was responsible for several high-profile crimes, including the murder of a prison guard in 1968 and the murder of a police officer in 1969.

1915

16 December 1915. Delightful Aerobatics was established - one of a number of small airfields around London that were created for the purposes of Zeppelin airship raids during the First World War.

1924

Following the Imperial Airways flight from London to Karachi, the first major civil aviation accident, conditions at Croydon were under criticism from the public inquiry.

1928

The terminal building and control tower were completed and the wooden air traffic control and customs building demolished. The new buildings began operations on 25 January and were opened on 2 May 1928.

1930

Amy Johnson took off from Croydon on 5 May 1930 for her round-the-world flight to Australia. She returned home around 13 months later.

1940

On 15 August Croydon Airport was attacked by the first major air raid on the London area. The airport was destroyed, the civil air terminal building was badly damaged by cannon fire and shot.
Get acquainted with Wheel Man

To help me describe how to art-direct designs for products and websites, I've made **Wheel Man**—an app for booking a fast car and a getaway driver—plus a companion website, full of advice on how to ensure you get away clean. The Wheel Man website also tells readers about some of the infamous London criminals and gangs of the 1960s, and the murders, robberies, and violence they committed. I'd be happy if you learn from the Wheel Man examples, but I don't suggest you follow in these criminals' bloody footsteps.
Explaining art direction

"The truth isn’t the truth until people believe you, and they can’t believe you if they don’t know what you’re saying, and they can’t know what you’re saying if they don’t listen to you, and they won’t listen to you if you’re not interesting, and you won’t be interesting unless you say things imaginatively, originally, freshly."

— Bill Bernbach
What art direction means

What do you think of when you hear the term “art direction”? If you work in advertising, or you’ve watched a few episodes of Mad Men\(^\text{10}\) on TV, you’ll have some idea of what an ad agency art director does, or at least did in the 1960s. At Don Draper’s agencies, art directors like Sal and Stan collaborated with copywriters like Peggy to develop creative concepts. They also art-directed photo shoots and supervised designers and illustrators.

Ask what art direction means to developers, and they might answer: using the `<picture>` element or `sizes` attribute in HTML for responsive images; presenting alternative crops, orientations, or sizes at various screen sizes. They’ve become useful tools, but there’s more to web design than using tools.

Fifty years after Mad Men’s fictional Sterling Cooper Draper Pryce pitched a new ad campaign with the tagline “Pass the Heinz,” real-life agency David Miami ran it in 2017. This proves to me that great communication really can be timeless. (Images copyright AMC and David Miami.)

\(^{10}\) [https://smashed.by/madmen](https://smashed.by/madmen)
Trent Walton adapts the design of his blog entries to communicate their subject matter more clearly.

Page speed and load times are the foundation for a positive user experience on the web.

Let’s face it, if your page fails to load in time, all the effort put towards information architecture, content strategy and interaction design will be for naught. During my formative front-end coding days, I did a fair bit of assuming that as internet connections transitioned from dial up to DSL I’d be able to make compromises. Using sprite-less PNGs upon redundant CSS upon tons of HTTP requests became commonplace, and a few seconds for a page load seemed worth it.

Not Seconds, Milliseconds

But that’s not really how things are. We close out the windows of a tabbed link if the load bar logs. We leave news and shopping sites early if they feel unresponsive and sluggish. If a site is slow, the amount of time we’re willing to continue to invest decreases exponentially with every second... every millisecond. Bing and Google agree that slow pages lose users. Each company can similar experiments where they deliberately delay server response time by a varying number of milliseconds finding that the longer users wait the less engaged they are with a page and the more likely they are to leave.

But good things come to those who wait. Right?

It’s tempting to want to build in enhancements - if all those extra...
Web designers may mention people like Jason Santa Maria\textsuperscript{11} and Trent Walton,\textsuperscript{12} who sometimes give their writing their own distinctive images, layout, and typography. This gets us closer to understanding art direction, but pictures, layout, and typography are only its result, not its purpose.

So if art direction isn’t exactly those things, what is it? One interpretation of mine is that art direction:

> Uses design techniques to intentionally evoke an emotional response from someone when they read an article, use a product, or visit a website.

**Facts and feelings**

Let me relate that interpretation to “Devils, Deals and the DEA” from ProPublica. When I design a news story about a drug cartel, I need to lay out who, what, where, when, how, and why:

- who was involved
- what occurred
- where it happened
- when the events took place
- how it came to pass
- why what happened, happened

I can state those facts in a manner that’s, well, matter-of-fact, but there’s usually more to telling stories than relaying facts. When someone’s reading about a Mexican drug lord, do I want them to feel calm or terrified? I’ve plotted those emotions on a scale:

\textsuperscript{11} [http://jasonsantamaria.com](http://jasonsantamaria.com)

\textsuperscript{12} [https://smashed.by/wheretostart](https://smashed.by/wheretostart)
Calm — Relaxed — Uncomfortable — Agitated

Anxious — Worried — Scared — Terrified

Encouraged — Motivated — Driven — Obsessed (We need to catch this guy!)

It helps me decide how I want someone to feel and the intensity of those feelings by plotting them on an emotional scale.

I ask myself how I want someone to feel before I start art-directing every new design. Of course, terrified wouldn’t be appropriate when I write advertising, marketing materials, or product notifications.

“Your file didn’t upload—be terrified!”

Instead, I’d choose a different set of emotions, more appropriate to the content and context. MailChimp’s Voice and Tone Guide\(^\text{13}\) contains a helpful list of example emotions:

### NEGATIVE FEELINGS

<table>
<thead>
<tr>
<th>Anger</th>
<th>Annoyance</th>
<th>Apprehension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boredom</td>
<td>Confusion</td>
<td>Criticism</td>
</tr>
<tr>
<td>Fear</td>
<td>Frustration</td>
<td>Helplessness</td>
</tr>
<tr>
<td>Shock</td>
<td>Skepticism</td>
<td>Stress</td>
</tr>
</tbody>
</table>

\(^{13}\) https://styleguide.mailchimp.com
## POSITIVE FEELINGS

<table>
<thead>
<tr>
<th>Anticipation</th>
<th>Confidence</th>
<th>Excitement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exhilaration</td>
<td>Joy</td>
<td>Pride</td>
</tr>
<tr>
<td>Relief</td>
<td>Surprise</td>
<td>Trust</td>
</tr>
</tbody>
</table>

## NEUTRAL

| Curiosity/interest | Determination | Optimism |

I wasn't exaggerating.

- Drug barons are **wicked**.
- They do **appalling** things.
- They're out there.

When art-directing their version of the drug cartel story, ProPublica chose a more dramatic direction because they wanted people to feel appalled while reading it. Their choice of Tim McDonagh's emotive illustrations of chaos, violence, and death helps to stimulate those emotions and shift readers up the emotional scale from interested towards being appalled about what happened. If, like The Atlantic, they’d chosen a more conventional photomontage, that needle might’ve moved less.

The Atlantic and ProPublica approached their interpretations of this story with different intents. The Atlantic approach is very much matter-of-fact, whereas ProPublica wanted to provoke a strong emotional reaction to their version. This is art direction, and as SuperFriendly Dan Mall explained:

---

14 [https://smashed.by/alaartdirection](https://smashed.by/alaartdirection)
Art direction brings clarity and definition to our work; it helps our work convey a specific message to a particular group of people. Art direction combines art and design to evoke a cultural and emotional reaction. [...] Without art direction, we’re left with dry, sterile experiences that are easily forgotten.

If you’re anything like me, whether you’re working on a product or a website, designing email newsletters, login screens, notifications, or an on-boarding process, the last thing you want people to feel is that your designs are dry, sterile, and quickly forgotten. Art direction can prevent that from happening.

**Lasting impressions**

Whereas the art direction on an individual story may last only a short time in someone’s memory, the art direction of an entire product or website can leave a mark on a business and its customers which lasts much longer.

Successful designs often have styles all their own, which are created when art directors choose colours, illustration and photography styles, layouts, and typography. These come together to form what I call *foundation styles*. They are what give a design its distinctive personality.

When I choose typefaces I keep several factors in mind. Legibility and readability are essential, especially for body copy, but I also consider the impression I want to make and how I’d like an audience to react. As Tim Brown wrote in his guide to combining typefaces:¹⁵

---

¹⁵ [https://smashed.by/combiningtypefaces](https://smashed.by/combiningtypefaces)
Finding emotion in type is simple for us typographers: we look at type specimens, read what others have to say about the face(s), and make our own notes (mental or otherwise) about our impressions. It’s a very visual, visceral activity. It’s lucky for us that, unlike texts, typefaces are most always ready and waiting to be appraised for their potential emotional contributions to an experience.

For example, when I’m directing the design of a product aimed at the financial sector, I would choose an anchor typeface like Aktiv Grotesk by Dalton Maag. This is not only clearly legible and easy to read, but was designed deliberately to not draw attention to itself. I choose this typeface because I’d like my audience to feel comfortable, reassured, and secure, and not be distracted by my typography.

Back in 2011, Douglas Bonneville wrote an excellent step-by-step guide to choosing typefaces.16

Whereas, if I’m art-directing a note-taking app that’s designed for a more casual audience, and I want my product to reflect the brand’s informality, I may choose a typeface with fewer straight lines and less symmetry, like Ideal Sans by Hoefler & Co. Pick the wrong typeface, and I run the risk of people feeling differently to how I’d intended.

16 https://smashed.by/choosetypeface
When I’m art-directing, my work doesn’t stop at selecting typefaces. It extends to the hierarchy, weight, and scale of everything from running text to headlines, image captions, numerals, pull-quotes, and tables.

The same is true for the colour palettes I make and the grids I create. Because these styles will stick around throughout a design’s lifetime, they become the basis for regular iterations as a product or website is redefined, and are the foundations for countless interactions between businesses and their customers. They define the personality of a product or website, and ultimately an entire company.

Although Apple’s native Notes app, Day One, and Vesper (RIP) do much the same thing, they all have different personalities because of the typefaces chosen.
My choices of foundation styles will be based on several factors:

- audience composition
- business model
- brand values
- current or intended position in the market
- design principles
- media: digital, print, and video

Major design projects often happen at pivotal moments in a business's, product's, or website's lifetime. Perhaps the goal of a company is to:

- reach new audiences
- enter new markets
- define a new category or disrupt an existing one

Make the correct choices and designs can turn these goals into success. Get them wrong, and they can break a business. If a design doesn't break enough new ground, the business may not succeed in its goal of reaching new audiences. Current customers may be alienated if design choices are incompatible with their needs and sensibilities. That's why redesigns should always be approached with the utmost care, perhaps by starting small and experimenting with real customers to mitigate any potential risk. Mark Porter\(^\text{17}\) said to me:

> Somebody once said to me that redesigning a newspaper or a magazine or a website is like going into somebody's house in the middle of the night and rearranging all their furniture. They come down in the morning and think WTF, but if you've rearranged it in a way that works and improves the experience, then people get used to it.

\(^{17}\) https://markporter.com
There are many factors which contribute to the success or failure of a business and art direction is just one of them, although it can be the most visible. That’s why people who consider art direction should always have a connection to customers and be part of a team which leads a transformation. After all, no one wants their redesign to be remembered for the wrong reasons.

Memorable experiences

I need to confess. I’m a sucker for advertising, but not everyone shares my fondness for it. In Purple Cow,18 Seth Godin wants us to:

> Stop advertising and start innovating.

Yet he acknowledges that it’s probably impossible to read through a list of successful brands without either picturing one of their commercials, remembering their taglines, or hearing their jingles ringing in our ears. Advertising has given us some of the most memorable creative work, and the mark of great advertising is that it stays with us long after a campaign is over.

Successful advertising always provokes an emotional response in us as consumers, and as Don Draper said in the first episode of Mad Men:

> Advertising is based on one thing: happiness. And do you know what happiness is? Happiness is the smell of a new car. It’s freedom from fear. It’s a billboard on the side of the road that screams reassurance that whatever you are doing is OK. You are OK.

18 https://smashed.by/purplecow
What art direction means

For a time, GOV.UK used icons intended to help people understand the type of page they were on. Testing revealed these icons didn’t help people, and they also failed to make pages more memorable, so GOV.UK removed them.19

A ____________________________

When I asked Mark Porter that question, I was surprised when he mentioned GOV.UK.20 I’ve often joked that GOV.UK isn’t known for its flamboyance or flair. I was only half joking, but I’d completely missed that GOV.UK demonstrates good art direction. Mark explained it like this:

19 https://smashed.by/retiringicons
20 http://www.gov.uk
GOV.UK is about getting information out in the clearest, most usable, intuitive form possible. They set themselves a target of creating the best experience for the widest audience. They’ve done that by creating a visual language, hierarchical information structure, and behaviours, and by taking accessibility seriously. It’s a wonderful thing.

Art direction means asking “What do we need this to be?” “What does our audience require?” “What’s the best way to use our imagination, design skills, planning, research, strategy, and thinking to create the best experience?” This is something GOV.UK has done incredibly well.

The fact that GOV.UK is art-directed well seems obvious to me now. I won’t find chaotic layouts, powerful images, or deconstructed type because those things wouldn’t feel right for that website.

The layouts on GOV.UK aren’t chaotic because they shouldn’t be. There are no unnecessary images added only for visual impact. Typography is legible, readable, and well structured. The design of each and every element on GOV.UK is appropriate to the site’s goals and the people who use it.

The experience of visiting GOV.UK is memorable for all the right reasons. I can find something easily, then quickly pay my car tax, renew my passport, or set up a business. I don’t enjoy paying taxes, but when I leave GOV.UK after a satisfying experience, my lasting memory is that it served me well.

Experiences of products and websites can be memorable for many different reasons. Airbnb is famous not only because they designed a frictionless experience across apps and website, but because they tell stories about the experience of going somewhere new, like Sydney.
Giving people an experience that’s consistent, engaging, and delightful across multiple platforms is especially challenging when a company needs to balance inspiration with the utility of booking a place to stay.

As Alex Schleifer, VP of design at Airbnb explained: \(^{21}\)

"We think in stories and in narratives. [...] A large part of what we do is really creating trust between two strangers. We do this by more than just building interfaces. We make sure to tell the right stories. Whether that’s advertising on traditional media or over social media but also, more and more, within the product itself. Telling our community stories — real stories from real people — making sure we properly communicate what we’re about to cities and neighbourhoods and the folks who live there."
Like Airbnb, Slack tells stories, but they are about how more streamlined communication can make you feel “more productive, less stressed, and just a little bit happier.” A good example is a story of how by using Slack, NASA’s Jet Propulsion Laboratory keeps all its teams’ communications in one place.

The Slack product is designed to make people feel good. Its colour palette feels friendly and playful, and so does the animated logo seen while the app waits to connect with Slack’s servers. The “Get up and stretch once in a while” messages from “Your friends at Slack” show they understand how people spend most days, and that those messages can make them smile.

Slack has been successful not just because the product connects people, but because their art direction connects those people with their brand. This is the humanity which art direction can bring to a product, even one which has been designed primarily as a utility.

Using Slack may not be as memorable as a weekend in Sydney, but they’ve worked hard to reduce the frustrations we often experience with other forms of communication.

https://slack.com
Anyone can art-direct

By now you might be wondering what the difference is between art direction, creative direction, and design? There's definitely an overlap between them, so it shouldn't be a surprise when people use these terms interchangeably.

Art directors can design, and they very often do. Designers can and do art-direct, but when you're art-directing, your role is different from designing.

Dan Mall explained how he sees the difference between art direction and design when he wrote:

> Art Direction [is] the visceral resonance of how a piece of work feels. In other words, what you feel in your gut when you look at a website, app, or any piece of design work.

Whereas in the same article, Dan described good design as “precision:”

> Design is the technical execution of that connection. Do these colours match? Is the line-length comfortable for long periods of reading? Is this photo in focus? Does the typographic hierarchy work? Is this composition balanced?

Reading Dan’s definition, you might get the impression that designers are skilled at implementation — needing a good eye for following guidelines — but the real art is in direction.

Mainly because of his magazine and newspaper experience, Mark Porter looks at any distinction differently:

https://smashed.by/creativedirection
Part of an art director's job is creating visual identities. Some of the decisions you make are about typographic systems. I think it’s really hard to say whether that’s art direction or design. I know that working with a photographer to create a set of beautiful images is art direction. I know that being a junior designer on a magazine and being given pictures and type to lay out is design. But in the middle, there’s an area that’s much more blurred, where it’s really hard to separate the two.

Digital design leader Phil Coffman said that:

Design is about problem-solving, whether you are a designer or an art director. The two roles differ in that the designer is more concerned with execution, while the art director is concerned with the strategy behind that execution.

The answer is more nuanced than both Dan's and Phil's definitions, mainly as the boundaries of what we consider to be design have expanded enormously.

**RENDERING INTENT**

The incorrigible Jared Spool explained design as “the rendering of intent.”

The designer imagines an outcome and puts forth activities to make that outcome real.

When he wrote “Design is the Rendering of Intent,” Jared contrasted signing up to two US government websites: Global Entry; and We The People, an online petition system. He explained that:

---

24 https://smashed.by/alaartdirection
25 https://smashed.by/jaredspool
Both [...] were designed by teams of government employees. Both teams wanted their constituents to engage successfully with the design. Yet the two designs approach that in a very different way.

The Global Entry sign-up, though well-meaning, is less friendly than interacting with We The People. Assuming each had similar resources and constraints, why would the two designs appear to be so different? The answer lies in the teams’ intentions and how they rendered them.

How those two processes look and work differently is design, but why they’re different is down to their art direction. They were directed with different intentions, and therefore it’s not surprising the results are different too.

Everyone should art-direct

With such a diverse group of people involved in the business of doing creative and technical work for the web and the crossover between what used to be separate disciplines, it’s now almost impossible to know what someone does from their job title.

Where once it was easy to understand the difference between a designer and a printer, today it’s much more difficult to understand the distinction between a designer, a product designer, and a UX designer. You’ll still find art directors in advertising, film, and publishing, but that’s not a title you’ll often see in a business, charity, government, or indeed a digital product company team.

Is this because there’s no room for an art director’s talents in those organisations? Of course not. Every organisation needs people who are good at making sure they communicate well with customers, partners, or other stakeholders throughout everything the organisation produces.
Does that mean every organisation needs someone with the title art director? No, because the role an art director plays is something which can and should be done by everyone.

Graphic designers, product designers, UX designers, and web designers should art-direct. General and product marketers can art-direct too. Even the CEO can influence art direction when they define strategies and are concerned with how people feel while they experience the business.

Art direction needn't be a separate consideration from normal activities either. Just like accessibility and performance, art direction should be a part of everything you do day-to-day. Of course, art direction takes a good eye, but everyone can be taught. It takes practice, but no more than becoming an accomplished designer or developer.

**OPPORTUNITIES FOR ART DIRECTION**

<table>
<thead>
<tr>
<th>Commercial</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Articles, blog entries, and news stories</td>
<td>Customer sign-up</td>
</tr>
<tr>
<td>Brochures and catalogues</td>
<td>Emails and notifications</td>
</tr>
<tr>
<td>Customer stories</td>
<td>Help and support</td>
</tr>
<tr>
<td>E-commerce checkout</td>
<td>Interface design</td>
</tr>
<tr>
<td>Emails and newsletters</td>
<td>On-boarding process</td>
</tr>
<tr>
<td>Product or service pages</td>
<td>Product marketing</td>
</tr>
<tr>
<td>Social media</td>
<td>Voice and tone</td>
</tr>
</tbody>
</table>
Do you need to convince your boss or your clients that art direction should be something to take seriously?

No, just do it.

Bring art direction into what you design or develop every day and people will notice the difference.

**MY NEPHEW, THE ART DIRECTOR**

One thing I find fascinating about the distinction between art direction and design is that something can appear poorly designed but still be art-directed well. That is, of course, as long as the choices made were intentional.

For example, I might look at garish colours and handwriting-style text and think they constitute poor design. Yet in the context of a birthday party invitation, they demonstrate proper art direction because to my 14-year-old nephew they feel bright, exciting, and fun.

Like many of the people I work with, my nephew can’t give me qualified feedback on complementary and tertiary colours, leading or tracking type, or choosing between grotesque and humanist sans serifs because he isn’t a designer—yet. However, he can offer advice on whether something feels bright or dull, loud or quiet.

It was Jared who tweeted: 26

> Anyone who influences what the design becomes is the designer. This includes developers, PMs, even corporate legal. All are the designers.

26 [https://smashed.by/jaredtweet](https://smashed.by/jaredtweet)
I don't think that “all are [...] designers,” but I do believe that anyone can influence the art direction a design expresses. This is because developers, project managers, corporate legal—even my nephew—can all tell us what a design feels like to them.

Whatever the differences between art direction and design, it’s true that both roles have changed as new media and technologies have emerged. Just as they did when the advertising industry moved from print-based work to embrace television, we've seen a similar change demanded by the web and everything that's associated with it.

As a designer, I now consider aspects of design, like performance and responsiveness, that often weren't concerns when I began designing websites twenty years ago. Similarly, when I art-direct I consider the many factors which influence how someone feels when they encounter my work.

How someone perceives the speed of a product while uploading a document, that's art direction. How people feel when reading an error message, that's art direction too. Do they smile when an animation bounces? Also art direction. When they close a product or leave a website and are feeling happy about using it. That's when art direction has done its job.
One hundred years of art direction

Bradley, Brodovitch, Brody, and Feitler—together, their names sound like a Mad Men-era advertising agency—all appreciated the broader arts and art history. Their interest in graphic design, photography, and typography gave their work extra dimensions. The knowledge of how to combine them enabled their work to have a more significant impact. Today we all need a similarly broad interest in art and design, as well as areas such as experience and interaction design.

Before the start of the twentieth century, advertisers were already using commercial illustrators to help them promote and sell their clients’ products, and Will Bradley was one of them.

Bradley learned about advertising, layout, and typesetting while working in print shops, before becoming a freelance designer. He was a keen illustrator and printmaker, and before long he was using those talents to design advertisements, books, and magazine covers, including Vogue. Bradley served as art editor for several publications, including Century, Good Housekeeping, and Hearst’s International where he designed typographic layouts and even new typefaces.²⁷

Throughout his career, Bradley maintained a keen interest in many aspects of the arts and design. Understanding image making, typography, and how to bring them together to communicate intent and tell stories became the fundamental skills of the art directors who followed him. In the first 50 years of the twentieth century, there was no better place to show off those skills than by art-directing magazines.

²⁷ https://smashed.by/bradley
Bulova Watch Co. paid for the first television commercial. It ran in New York before a Brooklyn Dodgers baseball game in July 1941. (The Dodgers lost that game to the Philadelphia Phillies.)

Print quality improved rapidly into the 1930s, and art directors were quick to take advantage of it. At the same time, the US saw an influx of design talent from Europe and elsewhere, and these designers brought a modernist approach to design, with its asymmetric compositions and blending of images with typography into what László Moholy-Nagy\textsuperscript{28} termed “Typofoto.”

One immigrant was Mehemed Fehmy Agha,\textsuperscript{29} whom Condé Nast brought to the USA to become art director at Vanity Fair and Vogue in 1929. There, Agha introduced the first double-page magazine spreads. His innovative processes also saw designers cutting preprinted pages of type and arranging them to create dummy layouts. The paste-up\textsuperscript{30} method of producing publications survived until the start of the desktop publishing era in the 1990s.

\textbf{My encounter with art direction}

I’m not old enough to remember Alexey Brodovitch’s\textsuperscript{31} work on Harper’s Bazaar magazine from 1934–1958. I’m not fashionable enough to have read Man About Town\textsuperscript{32} (later Town) men’s magazine during the 1950s and ’60s, so didn’t see Tom Wolsey’s work first-hand.

\textsuperscript{28}https://smashed.by/moholynagy
\textsuperscript{29}https://smashed.by/agha
\textsuperscript{30}https://smashed.by/pasteup
\textsuperscript{31}https://smashed.by/brodovitch
\textsuperscript{32}https://smashed.by/manabouttown
Man About Town was a popular men’s magazine, and Tom Wolsey worked with then up-and-coming (now famous) photographers, including David Bailey, Terence Donovan, and Don McCullin. Wolsey’s typography and use of images created designs which inspired a generation of magazine designers.
These talented art directors made work that’s not only stood the test of time, but has also become iconic. However, it would be foolish for us to merely mimic work from a previous era and medium. So what can someone who worked decades ago in the worlds of advertising and magazine art direction, teach us about our thoroughly modern medium?

1930s, Alexey Brodovitch

What can I say about someone who pipped Pablo Picasso to win first place in a poster competition? Too much to squeeze into this short book. Brodovitch was more than an art director; he was an artist who worked across fabrics, jewellery, and graphic design. He had a keen interest in painting and was a skilful photographer, teaching, among others, Diane Arbus and Richard Avedon. These skills served him well when he took on his most famous role as art director of Harper’s Bazaar.

In 1934, Carmel Snow, who’d just been appointed editor-in-chief of the magazine, came across Brodovitch’s work. Of it, she said:

“I saw a fresh, new conception of layout technique that struck me like a revelation: pages that “bled” beautifully cropped photographs, typography and design that were bold and arresting.”

He instinctively knew how to combine photographs with written content, often turning text into shapes that contrasted with or mirrored the forms in his photography. Brodovitch’s sketches show clearly how for him, photography and writing were intimately connected, something that’s no less relevant today than it was throughout the 1930s–50s.

---

33 http://smashed.by/carmelsnow
It’s evident, even today, that it was Alexey Brodovitch’s knowledge of photography that gave his work its classic feel.
I find Brodovitch's layout sketches most informative because I often learn more about how someone thinks by looking at their work-in-progress rather than at a finished result. They clearly demonstrate how he connected writing with photography.
Alexey Brodovitch’s art direction for the Harper’s Bazaar fashion magazine in the early part of the twentieth century has influenced designers ever since.
1960s, Bea Feitler

Who was Bea Feitler? Let’s begin with:

- She was one of Harper’s Bazaar’s first female art directors, together with Ruth Ansel.  
- In 1965 she cast Donyale Luna, the first black model featured on the cover of a mainstream fashion magazine.
- She worked with photographers Richard Avedon, David Bailey, and Annie Leibovitz.
- She kissed Andy Warhol.

In the early 1960s, and while still only in her 20s, Feitler was one of the publishing industry’s youngest art directors. With her bold use of colour, Feitler’s work on Harper’s Bazaar characterised the optimistic mood of the United States in that period.

35  https://smashed.by/ansel
36  Luna was also the first black model to appear on the cover of British Vogue in 1966, photographed by David Bailey. https://smashed.by/luna
Bea Feitler’s 37 Rolling Stone cover — featuring a photograph of naked John Lennon, wrapped around fully-clothed Yoko Ono, by Annie Leibovitz 38 — is one of the most iconic images of the former Beatle. Sadly, Bea Feitler died from cancer before her final work — the first issue of a revamped Vanity Fair magazine — was published in 1982.

37 https://smashed.by/feitler
38 https://smashed.by/leibovitz
Bea Feitler’s creative partnership with Ruth Ansel, and her close collaboration with photographers resulted in Avedon’s pink space helmet — modelled by Jean Shrimpton, an icon of Swinging London — and Leibovitz’s naked John Lennon. These are undoubtedly two of the most memorable magazine covers. But perhaps it was Feitler’s conviction and the bold choices she made during her career — not least her depiction of Donyale Luna — that we can learn most from.
1980s, Neville Brody

In the mid-1980s, I was obsessed with music, so it was hard not to know about The Face magazine. I knew there was something special about it, although at the time I didn’t know who Neville Brody was or what an art director did. The Face was a culture, fashion, and music magazine published in Britain until 2004 and Brody worked as its art director until 1986.

It was unlike anything that came before it. In an interview with The Guardian in 2017, Brody said of The Face: 39

> The Face was a living laboratory where I could experiment and have it published. Our golden rule was to question everything. If a page element existed just as taste or style, it could be abandoned. Page numbers could be letters or shapes increasing in size. We could start the headline on the page before. We had disasters and near misses every issue. We had two weeks to art direct everything, then a week to lay it out. It was pre-computer so everything was traced by hand. […] It certainly wasn’t a nine-to-five job. You had to be obsessed to make it work.

Brody’s work was heavily inspired by Dada and by constructivist 40 graphic design. Over five years, his deconstructed typography and thought-provoking layouts transformed The Face. In the years after The Face, Brody worked on global advertising and identity campaigns for brands including Dom Perignon and Nike. In 2010, Brody’s Research Studios were instrumental in designing the BBC’s Global Experience Language (GEL), 41 which paved the way for the corporation’s current visual identity. GEL is often lauded as one of the earliest and most successful design systems, and this work has had an enormous influence on how we design for the web today.

39 https://smashed.by/brodyinterview
40 https://smashed.by/dada
41 https://smashed.by/gelbbc
I remember Neville Brody’s\textsuperscript{42} art direction for The Face\textsuperscript{43} in the 1980s, and I’m still inspired by it every day. The Face under Brody’s creative stewardship helped redefine the British music press from the 1980s onwards. It became a pop culture mirror to the political and social turmoil in Britain at that time.

The Face maintained punk’s rebellious attitude to the establishment and represented a contemporary pop-culture equivalent of the Dada movement’s opposition to everything bourgeois.

\textsuperscript{42} https://smashed.by/brody
\textsuperscript{43} https://smashed.by/theface
Even twenty-five years after he created them, Brody’s pages from The Face magazine are still remarkable designs.
Many magazines and newspapers have struggled to bring the quality of design and art direction from their print publications to the web. One exception is ProPublica. It faces the same editorial, creative, and technical challenges as other news organisations, but has solved them by developing a process that engages art directors and designers early in the editorial cycle. Helped by art director Rob Weychert, ProPublica’s team also developed modular page templates and a custom story rig, which allows them to add specific styles to art-directed articles. This allows enormous flexibility without compromising consistency with their regular, templated articles.

Lost Mothers
An estimated 700 to 900 women in the U.S. died from pregnancy-related causes in 2016. We have identified 134 of them so far.

by Nina Martin, ProPublica, Emma Cillosens and Alessandra Freitas, special to ProPublica
July 17, 2017
“Lost Mothers” is a series about maternal mortality in the United States and the rates of women dying due to pregnancy-related complications. For “Lost Mothers”, Rob Weychert not only used the dots grid as navigation to a story about each woman, but also to illustrate just how many women have not been identified. As Rob explained to me, “The idea behind the design was not to shock people necessarily but to give you a sense of how little is really known or understood about what’s going on.” Each vignette also includes photographs of a lost mother’s family and loved ones to give their story more emotional impact.

The appearance of the “Lost Mothers” gallery was partly inspired by the floor of the elevator in ProPublica’s building, which has a dot grid. Rob told me, “I see it every single day, and one day I thought, ‘Oh! That could be an interface.’”
Art-directing experiences

Now, some of you might be wondering about skipping this chapter because you think your work doesn’t involve telling stories—but don’t give up. Whether we write fact or fiction, sell or make products, the way to engage people, create desire, make them happy, and encourage them to stay that way, is by creating narratives.

When he wrote “Art Direction Goes Digital,” his chapter in *Art Direction Explained, At Last!*, Khoi Vinh said:

> Good art direction […] is good storytelling. Throughout its long history, what we’ve valued most in the practice of art direction is this: the ability to control the myriad elements of design in such a way as to produce a sustained, cohesive, and distinctive narrative experience.

If you aren’t the studious type, a narrative is “a spoken or written account of connected events; a story.”

What’s important about that definition isn’t so much that an account is spoken or written, but that the events a story describes are “connected.” Those connections create pathways in someone’s mind. While they’re on those paths, you can influence how they feel through art direction.

Lessons on crafting narratives usually teach that before writing your first word you should decide on outcomes, a cast, their backstories, and a plot. In this example of a design narrative, the role of CEO of Panzera (a watch company), could be the author of this story.

---

44 https://smashed.by/adexplained
45 https://www.subtraction.com
46 https://smashed.by/narrative
47 https://www.panzera.com.au
1. What’s the outcome we want? How do we want someone to feel?

Panzera watches are the best in their class. If I were Panzera CEO, I would want people to feel happy to have discovered our brand and excited to purchase. We aim to make learning about, buying and then receiving our products, a fabulous experience. We want to turn casual visitors into customers, then ambassadors for our brand. (Those are some lofty goals.)

2. Who are our characters?

(Did I say “characters”? We mean our customers, readers, or users.) They’re mostly between 30 and 55 years old, or people who buy unique gifts for them. They have some disposable income, but not unlimited spending power. They’re prepared to spend more because wearing a great watch is important to them. The watch they wear makes a statement about them, and they love to talk about it.
3. **What are their backstories? Think of these as user personas.**

“Reggie’s in his mid-thirties, and he’s the owner of a night club in Knightsbridge, a fashionable part of London’s West End. He often visits other business people and wears watches partly as a symbol of his success and also as a talking point. Reggie’s collection includes watches from mainstream luxury brands, but Reggie wants his next timepiece to be something different. He’s aware of our brand, has visited our website several times, but we’re only one of the manufacturers he’s considering.”

4. **What do we want to happen?**

Some narrative tutorials recommend flowcharts to help keep track of a plot, in much the same way as I might map a customer journey. Think of plot points as touchpoints, and I hope you see just how similar designing user experiences is to writing stories.

1. Reggie revisits us and sees our latest edition.
2. He spends several minutes browsing photographs and reading.
3. Scarcity matters and Reggie checks availability.
4. With only a few watches remaining, he decides to purchase.
5. Checkout is smooth, and Reggie feels confident, happy.
6. He smiles while reading the confirmation email we sent him.
7. A few days later, we deliver Reggie’s watch.
8. He takes a sharp intake of breath when he opens the box.
9. Before he’s even put the watch on his wrist, Reggie posts a photograph to his social media feeds.
10. At his next meeting, Reggie can’t wait to show off his new watch.
Art direction helps to tell stories like this, and the experiences that artefacts, interactions, and systems create turn these stories into reality.

With stories now told across a variety of media, how can art direction affect someone’s experience across email, product and packaging, social media, and website so they feel the way we intend and both get the outcomes we’re looking for?

**Stories on social media**

Ignoring the importance of social channels is impossible. They’re now frequently a brand’s main media for conversations with customers, and their most regular experience of it.

---

Panzer uses Facebook, Instagram, and YouTube to share stories about its watches. *(Courtesy of Panzer.*)
In the previous story, Reggie’s already familiar with the Panzera brand, and that’s no accident. The company has spent years art-directing their social media channels. Everything they post is intentional and immaculate. They publish photographs on Instagram regularly. These are all shot by professional photographers and are carefully art-directed to maintain a consistent message.

As well as the content and style of photography and video, art direction also influences the tone of voice and language a company uses when writing on its social and other marketing channels.

Panzera has been making watches only since 2009, but watchmaking history resonates with customers. Their vocabulary emphasises words like “heritage” and “tradition” concerning the company, and “modern vintage” and “neo-classic” whenever they talk about their designs.

Panzera writes about itself and its products using a consistent tone of voice and a vocabulary that aligns with its message. (Courtesy of Panzera.)
Stories on websites

The Panzera website is well designed, and it’s easy to navigate, but there’s more to a compelling page than an attractive design.

Art direction helps ensure a website design fully supports a company’s values through the choice and style of photographs or illustrations. It connects visuals with written messages to tell stories, and through them accomplish a company’s goals for their website.

Watches are an important talking point for Reggie. To appeal to people like him, the Panzera website emphasises the most distinctive aspects of their timepiece design. Copy focuses on their crown and hand designs, reinforced by close-up photography.

We don’t just art-direct images and type; we can direct experiences too. (Courtesy of Panzera.)

The watch is only part of the story, though. Equally important is what people say about it, so the company includes reviews and testimonials to increase its products’ appeal. Some people might describe insights like this as “psychology”, but I’m no psychologist. To me, this is just knowing a little about what makes people tick. And, of course, art direction.
Intentionally unremarkable stories

As the web matured, people developed expectations and mental models of how parts of a website should function and sometimes how they look. Product, price comparison, and checkout pages are among them.

When I buy something, I want to feel reassured a website is secure and a company will use my personal information carefully. I like to know there won't be hidden delivery costs, and how quickly what I've bought will be delivered. Above all, I want to be delighted, even before an item's been dispatched.

Reggie didn't feel uncomfortable while paying for his new watch. The checkout didn't ask for more information than it needed to process a payment. The site was secure and, all in all, the checkout process was unremarkable. Just how it was designed to be.

When Reggie heard an email notification and checked his inbox, what he found wasn't a typical confirmation. Instead, he saw something that looked like a personal message, thanking him for his order. The email included a summary of Reggie's purchase, links to exclusive videos, and an invitation to join a private owners' area containing information about using and maintaining his new watch.

All of this was art-directed to make Reggie feel first reassured, then delighted he'd chosen to buy from the company.

Turning stories into reality

The Panzera CEO's goals are to “make receiving (our) products a fabulous experience” and “turn casual visitors into customers, then ambassadors for our brand.”
Achieving these objectives means a well-designed experience shouldn’t stop when someone’s handed over their credit card number. The experience of opening a product for the first time can be art-directed too.

Countless videos are a testament to how effective Apple’s unboxing experience is. Apple reportedly has a room dedicated to designing and testing new packaging.

_No one gets excited by “You’ve Got Mail!” any more, so it’s even more critical to design emails people don’t hate you for sending._ (Author’s design.)
As Adam Lashinsky recounted in *Inside Apple: How America’s Most Admired—and Secretive—Company Really Works*:

> For months, a packaging designer was holed up in this room performing the most mundane of tasks—opening boxes.

Reggie’s watch arrived in a sturdy outer box, so he wasn’t concerned how it had fared on its journey to him. Inside that was a cardboard sleeve, embossed with a silver Panzera logo. This sleeve was included not for added protection but to deliberately build Reggie’s anticipation.

When Reggie slid off the sleeve he revealed a soft leather case with another logo imprinted on it. The hinges had just the right amount of stiffness, and when he opened it, Reggie got the first glimpse of his new watch.

Before he’d even put the watch on his wrist, Reggie took out his phone and posted photos of it to Facebook for his friends to see. When they liked his pictures, Reggie felt even happier about the watch he’d bought.

Every time Reggie interacted with Panzera—through social media, its website, notification emails, and finally, how his watch was delivered—the experience the company gave him was art-directed. This experience turned him from visitor to customer, then brand ambassador.

Just as the company had intended.

---

48 https://smashed.by/insideapple
CLEVER COPY MAKES EXPERIENCES MEMORABLE

Throughout the 1950s and ’60s, advertising legend David Ogilvy was a strong advocate of long copy—sometimes very long—and he wrote ads that contained more words than you’ll find in many of today’s books about web design. Ogilvy wrote in his own book, *On Advertising*:

> All my experience says that for many great products, long copy sells more than short.

He cited one successful advertisement for Merrill Lynch in the New York Times that ran to 6,450 words. His Ogilvy and Mather agency once made an ad for US Trust that contained 4,500 words. Another for World Wildlife Fund was 3,232 words. Ogilvy wrote:

> Advertisements with long copy convey the impression that you have something important to say, whether people read the copy or not.

This runs counter to what many people would expect, particularly on websites, but Ogilvy was apparently from a school of advertising which believed that—as Dr Charles Edwards, Graduate School of Retailing, NYU wrote—“the more facts you tell, the more you sell.” We might flinch at the thought of reading thousands of words in an advertisement today.

Things changed in the late 1950s with DDB’s (Doyle Dane Bernbach) widely cited “Think Small” and “Lemon” ads for Volkswagen of America, art-directed by Helmut Krone.

---

49 https://smashed.by/ogilvy
50 https://smashed.by/onadvertising
51 https://smashed.by/krone
Think small.

Our little car isn’t so much of a novelty any more.

A couple of dozen college kids don’t try to squeeze inside it.

The guy at the gas station doesn’t ask where the gas goes.

Nobody even stares at our shape.

In fact, some people who drive our little flivver don’t even think 32 miles to the gallon is going any great guns.

Or using five pints of oil instead of five quarts.

Or never needing anti-freeze.

Or racking up 40,000 miles on a set of tires.

That’s because once you get used to some of our economies, you don’t even think about them any more.

Except when you squeeze into a small parking spot. Or renew your small insurance. Or pay a small repair bill.

Or trade in your old VW for a new one.

Think it over.

141 words, eight selling points—not including probably the most critical fact that the VW is free of college kids. Having had a kid at college, I can tell you how important that is.

From “Think Small” and “Lemon” onwards, the Volkswagen of America ads were consistent in their format: ample space for a photograph of the car, and a smaller one for everything else—tagline, body copy and logo. The car didn’t always fill the space and was mostly shot against a plain background.

Clever copywriting and art direction portrayed the Beetle as a smart choice.
In 1969 they ran “It’s ugly, but it gets you there.” That tagline, a photo of the Apollo 11 lunar landing module, and a VW logo. I love “And if you run out of gas, it’s easy to push” and these other classic taglines:

- It makes your house look bigger.
- The only water a Volkswagen needs is the water to wash with.
- After we paint the car we paint the paint.

DDB’s campaign for the VW Beetle wasn’t memorable just for its clever copywriting and distinctive art direction, but also because it told the truth about Volkswagen’s product.

The Beetle was noisy and small, but it was also well built and reliable. That was the truth. DDB’s advertising didn’t hide it, and customers responded to that and the messages the advertising conveyed.

A Beetle was a smart choice, and people aspired to feel smart about choosing one. In many ways, the ads said, “This car is smart and individual, like you.”
Art-directing the Apple App Store

Most tech product companies seem satisfied with an illustration aesthetic that makes them appear interchangeable. On the other hand, Apple’s redesigned App Store has become one of the best places to find a variety of styles commissioned from a wide range of illustrators. These illustrations do more than just decorate the App Store. Their sheer variety helps draw people in and set the tone for a particular theme or app category.

Apple’s investment in editorial design extends beyond its use of illustrations to the ways images are used to bring stories about apps and their developers to life. Crops use unusual shapes and transforms, matched with a carefully considered approach to typography.
If you're looking for an example of how art direction and an editorial design style can benefit a product company, you won't need to look too much further than the App Store.

Khoi Vinh: Two Very Different Kinds of Illustration: subtraction.com/2018/01/03/two-different-kinds-of-illustration
Art direction and creative teams

Bill Bernbach\textsuperscript{52} didn’t begin his first job in advertising until he was 30, but what a start he got. In his first agency job, Bernbach met a then young graphic designer, Paul Rand,\textsuperscript{53} who’d joined the fledgeling Weintraub agency as head of art.

Rand influenced Bernbach’s trademark taste for simplicity, and both believed that advertising visuals could do more than illustrate written copy—they could be equals in conveying ideas. This was a revolutionary notion; at the time, advertising copywriters turned propositions into copy, and then commercial artists, or “visualisers,” illustrated it. Rand’s ideas made sense to Bernbach because, as art director Bob Gage\textsuperscript{54} explained:\textsuperscript{55}

\begin{quote}
Two people who respect each other sit in the same room for a length of time and arrive at a state of free association, where the mention of one idea will lead to another idea, and then to another. [...] The combination of the visual and the words [...] forming a third bigger thing.
\end{quote}

Bernbach took this idea to DDB—the agency he co-founded in the 1940s—where art directors and copywriters worked together in creative teams. Out of these pairings DDB produced some of the most iconic work of the mid to late twentieth century, and while making it, Bernbach changed the way great advertising is made.

\textsuperscript{52} https://smashed.by/bernbach

\textsuperscript{53} https://smashed.by/rand

\textsuperscript{54} Gage worked with Bernbach on several iconic campaigns and later became the first art director in advertising inducted in the Art Directors Club Hall of Fame.

https://smashed.by/gage

\textsuperscript{55} https://smashed.by/mirrormakers
It's this spirit of collaboration that we should embrace today. We should form teams who follow strategies built around common goals. Those teams can include; brand owners, graphic, interaction, and product designers, developers, project managers, and writers.

**Teaming up with brand owners**

Some people often conflate branding with an organisation's visual identity—a colour palette, logo, tagline, and typography—and perhaps its tone of voice. But branding is more than what people hear or see; it's about personality, purpose, and above all the relationship an organisation has with other people.
Brand values

A business depends on good relationships with customers. Howard Schultz, who twice was the CEO at Starbucks, said:

“If people believe they share values with a company, they will stay loyal to the brand.”

Many organisations start by basing brand values on how they’d like to be perceived by customers. They think of words like “friendly,” “professional,” and “reliable.” However, those words could apply to any business operating in the same sector.

I know some shady people, but have you ever met someone who intentionally didn’t want to be perceived as professional or trustworthy?

Brand values are different from company to company. For example, Adobe’s values are “Genuine, Exceptional, Innovative, and Involved.” Build-A-Bear’s values perfectly match the playfulness of their brand; “Di-bear-sity, Colla-bear-ate, and Cele-bear-ate.” Perhaps my current favourite set of values comes from Squarespace. Its values include: “Be your own customer,” “Design is not a luxury,” and “Good work takes time.”

Brand values will form your strategy for art direction and help keep that strategy in mind when making decisions about design.

WHEEL MAN BRAND VALUES

As the owner of Wheel Man, what do I want customers to say about my service? It’s important not to overthink it. Instead, I thought about three or four values I wanted people to see in my organisation.

- **Fast**: Quick to use, quick to get you away.
- **Honest**: Even criminals want to know who they can trust.
- **Reliable**: Who wants to be left standing outside a bank with a sawn-off?
Brand values should be consistent throughout everything you do, but what should they be?

- **The Knowledge**: We know the best back-road routes to escape down.
- **Have courage**: To drive fast and get away clean.
- **Be adaptable**: You never know what’s around the next corner.

How do those values translate into art direction for Wheel Man? They inform its messaging and why people should use them to get away, summed up in the straightforward tagline:

“Get away with it.”

Having decided on your organisation’s brand values, don’t hide them away in a PDF. Instead, get creative, and design posters to display them for everyone to see.
Values should also influence the tone and voice of everything Wheel Man writes, from the content on the website to the style of notifications and other microcopy in the app. This copy is an ideal place for Wheel Man to show its distinctive personality by including labels like “Burner email” and “Set heist time.”

Successfully translating brand values into art direction is essential to ensure any product or website reflects the organisation that makes it. Have you defined brand values for your organisation? If you haven’t, spend some time considering what you want customers to say about your organisation, product, or service and how you’d like them to feel about it.

**Design principles**

Whereas a creative brief tells you about the goals of an individual design project, design principles help you maintain a consistent brand experience across many products and websites, and helps you develop ways to judge whether a design reflects your organisation’s brand, goals, and values.

Many well-known companies have now made their design principles available to learn from, and just like brand values, these principles are different from company to company. For example, Google’s “Ten things we know to be true” include:

- Focus on the user, and all else will follow.
- Fast is better than slow.
- You can make money without doing evil.

The UK Government Digital Service’s design principles are “intended for people building digital services for the GOV.UK domain” and they include: “This is for everyone,” “Design with data,” and “Do the hard work to make it simple.”
**WHEEL MAN**

When I was designing Wheel Man, the single word principles I kept in mind were:

- **Bold**
- **Fearless**
- **Timely**

The form your design principles take will depend on how much you want to clarify the meaning of your words. You might, as I did, use just a single word, like “Fearless.” Or you might expand that one word into a phrase to provide a little more clarity:

Having decided on Wheel Man’s brand values, I don’t want to stash them away. Instead, I create posters and display them on my walls.
“Make bold choices which get the job done.”

“Don’t be afraid to make tough decisions for our customers.”

“Give people only what they need, when they need it.”

You might also bring those formats together like this:

**Fearless:** Don’t be afraid to make tough decisions for our customers.

Boldness needn’t be merely about prominent colours or bold type; it can also be represented through a choice of layout. A layout which breaks with convention and uses a full-height image instead would also be a bold choice.

UK Government Digital Service design principle posters by Paul Downey.
Of course, principles like “bold,” “fearless,” and “timely” won’t be relevant for every organisation, so you should spend time considering the best principles to guide your designs. Don’t do this alone, because often the collaborative process of deciding on your organisation’s design principles can be as valuable as the principles themselves.

Design principles should influence the art direction, design, and development of every aspect of a product or website, but that can only happen when designers and developers keep them in mind. I found the best way to ensure everyone involved with a design does this is to turn them into posters for the walls of a studio or in a client’s office.

This is something the Government Digital Service did very successfully, and their posters make it obvious that in a field like ours, we can make our tools as creative as our work.

**COLLABORATING WITH SMASHING MAGAZINE**

A few years ago, the owners of Smashing Magazine asked me to help during the concept design phase of their new website. Redesigning a famous brand—particularly one that’s so well-known to its audience—isn’t a job I undertake lightly.

Even the most detailed brief is unlikely to articulate precisely how an owner perceives their brand, so there’s no substitute for spending time with them.

After several weeks of workshops and dozens of design iterations, I delivered a set of concept designs, ideas, and prototypes so the Smashing Magazine team could communicate what they needed from a new designer much more clearly.

Armed with my concepts, the Smashing team turned to Dan Mall and asked him for fully-fledged designs which “felt right” for
Smashing Magazine’s distinctive personality and unique positioning. The team used words like “community,” “quirky,” and “vibrant,” to describe their brand and wanted designs which were “turned up to eleven.”

Clarifying what words like “vibrant” mean visually is something Dan does exceptionally well. To ensure everyone shared a common understanding, he created a mood board which was full of visual examples of Smashing Magazine’s design principles. To use Dan’s words:

*The Smashing community is a vibrant community so let’s have a colour pallet that feels vibrant. Let’s use bright reds, oranges, and blues that are very vibrant and then darker colours to offset them.*
Smashing Magazine design principles by Dan Mall. Quirky? Yes. Vibrant? Absolutely!

**VISUALISING DESIGN PRINCIPLES**

Smashing Magazine's events are quirky, and the experiences people have while attending them are “turned up to eleven.” Dan playfully represented this by giving rounded elements a radius of precisely eleven pixels. And, if you haven’t spotted it already, he rotated their avatars and other features by exactly eleven degrees.
These discussions around design principles helped Dan create a design that's distinctively Smashing Magazine. He applied their signature orange liberally and chose two typefaces: one for readability, the other for its quirky letterforms.

Whereas similar sites display small avatars of their authors, Dan chose to emphasise the people behind Smashing Magazine's content by using large photographs, placed above, not below their writing.

Design principles like “vibrant,” “quirky,” “dimensions,” and communities “turned up to eleven” are useful tools for judging whether a design feels right.

They’re also helpful when designing experiences which span channels and media, including emails, social posts, and website. They even help ensure that events, offline merchandise, and sales collateral all reflect the personality of a brand.

**Teaming up with editorial and marketing**

Mark Porter is perhaps best known for his work on magazines and newspapers, including The Guardian. Mark knows that it is written content—in the form of headlines, body copy, and pull-quotes—that draws readers to a publication.

In our conversations, Mark’s made the importance of designing with authentic content abundantly clear:

> If we’re designing a newspaper or redesigning a magazine from almost the very beginning, we’re using real pictures, real headlines, because we can’t judge an effective piece of design unless there’s real content.
In what’s now become an infamous example of art-directed influence over someone’s understanding of a piece of writing, David Carson56 used ITC Zapf Dingbats—a font that contains only symbols—as a statement on what he thought was a boring interview with Roxy Music frontman Bryan Ferry in Ray Gun. (Author’s recreation of the original design.)

Understanding content is essential to art direction, and David Hillman57 (coincidentally also formerly of The Guardian) said that:

“In its best form, [art direction] involves the art director having a full and in-depth understanding of what the magazine says, and through design, influencing how it is said.

This means when we are art-directing, we must work closely with editorial teams to organise, order, and structure written content. Even though art direction is typically responsible for look-and-feel, and editors have responsibility for the written material, everyone sees the best results when they’re considered equal partners.

56 http://davidcarsondesign.com
57 http://studiodavidhillman.com
For many organisations, a collaboration between art direction and other parts of a business isn’t easy to accommodate. It requires strict planning, flexible systems, and above all an understanding between people with different experiences and talents. In product companies, art direction should be considered by general and product marketing, sales, and others.

COLLABORATION AT PROPUBLICA

David Sleight\textsuperscript{58} is design director at ProPublica,\textsuperscript{59} a non-profit investigative journalism newsroom, which publishes stories in the public interest. When I asked David about collaboration between designers and editorial at ProPublica he told me:

\begin{quote}
For us, everything starts as words, so a lot of email and talking. We do one-on-one meetings where we’re trying to suss out opportunities by talking to people. “This sounds a really interesting story. Tell me more about it.” Our role’s like group psychologists. “Tell me what you’re thinking about this story.”
\end{quote}

At ProPublica, their agile design process often starts by sketching ideas in a notebook and having conversations like: “Here’s what I think our opener might look like. Here are a couple of things I’m thinking about.” When the time between design and publication is short, ProPublica’s designers don’t spend it making high-fidelity visuals; instead, they switch from sketching to code quickly and this gives everyone more opportunities to iterate on a story right up to publication.

This mixture of designing and developing means that when designers create a layout for a ProPublica article, they’re often the ones who develop it using CSS and HTML. This is what happened when ProPublica worked on “Devils, Deals and the DEA.”

\textsuperscript{58} http://stuntbox.com
\textsuperscript{59} https://www.propublica.org
For this story about violence, Sleight chose to commission illustrations by Tim McDonagh to communicate violence with an intensity that would’ve been difficult to stomach using photography. David explained how they decided to illustrate this story:

We talked to David, who is the reporter, and he was giving us details of the story. Most of them were like, “Holy jeepers! This is guns, shoot-outs, and crazy shit. They’re melting people in barrels of acid. We asked, “What part of your story are you going to elevate?” and he said, “This is insane and I need to show that it’s an insane world.”

60  https://www.mcdonaghillustration.com
Devis, Deals and the DEA

Why Chapo Guzman was the biggest winner in the DEA's longest running drug case

by Michelle Zatlyn, ProPublica
December 23, 2015

This story was co-published with The Atlantic.

FOR 14 MONTHS, the first thing Dave Horrocks, a special agent with the Drug Enforcement Administration, did every morning was boot up his laptop and begin tracking a 45-foot yacht with Dock Holiday painted on the stern.

In the summer of 2013, the DEA had intercepted communications in which members of a Mexican drug cartel called the Jalisco New Organization discussed buying a yacht in California. Horrocks and his colleagues used the classified anti-kickback magazine and determined that the Dock Holiday was the boat the APD wanted. DEA agents then managed to get an invite and forced docking because the boat was insured. That's when Horrocks started watching the boat on his laptop.

Since the early 1990s, the APOs have been the inspiration for some of the top-notch drug trafficking operations in the world, and the stories of their activities never fail to amaze and terrify. But for everyone else, the story of the APOs' rise to power is the story of their ruthless and often violent tactics. Their success has come at a high cost: the lives of countless people, including women and children, have been lost as they have tried to escape the violence.

Discussion Recap

ProPublica's Dave Horrocks and former APD agent Josh Eiden, host a discussion about this story with Emily Martinez and Rob Weychert.

By now, the DEA had already been on the scene for over a year, and the agency was starting to feel the pressure. The APOs were using the boats to transport drugs, and the DEA needed to stop them. They had already caught the leader of the APOs, but the rest of the cartel was still out there. The DEA knew they needed to act fast if they were going to stop the APOs once and for all.

Eva was a fighter, a no-nonsense special prosecutor with a reputation for getting results. She was known for her quick thinking and her ability to get to the root of a problem. She was a woman who never backed down when she saw an opportunity to make a difference. She was the perfect fit for the DEA's case.

Meanwhile, the DEA had set up a hotline and put up posters at border crossings to encourage people to come forward with any information that could help them stop the APOs. They had also been working with the Mexican government to try to stop the APOs from moving across the border with their boats.

After a few months, the DEA had managed to catch the leader of the APOs and they were on their way to stopping the rest of the cartel. They had done it! They had stopped the APOs and saved countless lives. But the battle was far from over. The DEA knew they had to keep an eye on the cartel and make sure they didn't try to do it again.

"Devis, Deals and the DEA." Production by Emily Martinez and Rob Weychert. (Courtesy of ProPublica.)
“Busted.” Design and production by David Sleight, Rob Weychert, and Hannah Birch. (Courtesy of ProPublica.)

“I’m like, “OK, we’re going to get you insane art, something a couple clicks over the top that’s just going to reach out and grab somebody.”

Compare the design of that article to “Busted,” an exposé of false positives in police roadside drug-test kits. ProPublica emphasised the effects of being falsely accused, making the testing kit header image small to contrast with the devastating consequences.

These stories and others share ProPublica’s design DNA. There’s some visual consistency between them, but they often feel entirely different.

https://smashed.by/drugtest
Not every story ProPublica publishes has the same degree of art direction. In the production of everyday stories, an art director’s role often extends only to commissioning illustration and photography, then publishing them through the content management system.

When ProPublica designed its CMS, they knew it needed a system of templates capable of publishing thousands of stories, but which was also flexible enough to allow for art-directed pieces.

Instead of buying an off-the-peg solution, ProPublica developed a system in-house, one that’s tailored to suit their own needs. On top of standard templates, they created a “custom story rig” which enables them to customise colour, layout, and typography styles for their art direction.

As Dan Mall explained to me:

“When you read a magazine, not all articles, especially feature articles, look templated. A lot of them ask for custom layouts [...] but when you understand them, you realise they’re not actually custom. They’re just modular enough so you can have the semblance of being custom.”

Combined with frameworks, libraries, and systems, an agile process for design has proved successful at ProPublica for enabling art direction at scale. It can work equally well in businesses, governments, and product companies.

**Teaming up with product designers**

Is art direction relevant to product design? Of course. When you’re designing a product your design thinking may be different, but the principles of art direction stay the same.
The term *art direction* doesn’t always resonate with people who design products, even though considering how people feel can make the difference between a good product and great one. According to Stephen Hay:62

> Good art direction always has a story. Even if you can’t see what the story is, it’s an undertone, a subtext. You feel it somehow. That’s what makes one product completely different from another.

**DESIGN SYSTEMS SHOULD BE BORING**

As designers’ conversations shifted increasingly towards products, so did talk about systems for design. This isn’t a book about design systems—Alla Kholmatova has already written that63—but their popularity does raise interesting questions about art direction which Alla doesn’t answer.

**BOOTSTRAP NEEDN’T BE BORING**

There are definite advantages to creating and maintaining design systems for products and websites, not least that they mandate a level of consistency and help deliver a unified experience. This is partly what led to the popularity of frameworks like Bootstrap.

Bootstrap has became so ubiquitous, it seems there’s barely a website which hasn’t incorporated at least some of its class values or patterns. Bootstrap itself isn’t boring, but plenty of products and websites based on it certainly are because people often fail to graft their own brand’s DNA onto Bootstrap.

---

62  http://www.the-haystack.com
63  https://smashed.by/designsystemsbook
I guess you’ve seen those “Which one of these two layouts are you designing today?” tweets, lamenting the current state of design on the web. Does Bootstrap create unimaginative designs? No, people do.

At the most basic level, a design system should prevent designers from needing to repeat work. After all, when I’ve designed the most suitable interface element for one part of a product, wouldn’t designing it again for another be boring? When Josh Clark wrote “The Most Exciting Design Systems Are Boring,” he thought the same:

> The more common the problem, the better. Design systems should prioritise the mundane.

If product design needs a system, and systems are boring, where does that leave people who don’t want to feel bored? What can art direction do for products?

**DESIGNING A CUSTOMER JOURNEY**

I always smile when I hear designers referring to someone using a product or website as a “journey.” After all, buying a cable or paying your car tax online is hardly the same as flying from London to Sydney. Still, journeys or not, mapping a customer’s experience can be a very helpful process to help you improve it. So what is a customer journey and how do you map it? I think that Paul Boag has the best explanation. He wrote:

> A persona focuses on the person, while a customer journey map focuses on their experience. A customer journey map is a story designed to provide insights into the customer’s journey.

64 https://smashed.by/jongold
65 https://smashed.by/boringdesignsystems
Customer journey maps give you a view over the entirety of a customer’s experience and show you where there might be opportunities to improve it through art direction. Journey maps can be powerful tools for art direction because they help you understand not only what someone’s trying to do, but also how they might be feeling while doing it. For example:

- How might someone feel after pressing a link in an email or a social post and before arriving at your website for the first time?

- What could someone be feeling while they’re signing up for your product or service?

- How might someone feel when they receive a confirmation email about something they’ve purchased or a booking they’ve made?
When someone sees a link in a social post or email they might be curious and interested. After finding out more about a product or service, they might feel relieved to have found a solution to their problem. While signing up, they could feel excited. After making a booking or purchase, we hope they’ll feel happy.

A customer journey map needn’t be a complicated diagram. In fact, the best way to illustrate what a customer might feel is to show you a map I created for Wheel Man (previous page).

I might sometimes be more specific about a particular area of a product or website which, in the case of Wheel Man, might include:

- Setting a heist time
- Picking a destination (car swap, private airstrip, or safe house)
- Choosing a getaway driver
- Agreeing to the take
- Confirming the job

Think about the touchpoints a customer may encounter, for example, when moving from buying to retaining. Your touchpoints might include sign-up forms, payment pages, login, and an on-boarding process. Whatever your map illustrates, take care not to over-complicate matters or confuse people by including too much information. Your maps should include only what’s needed to describe what someone is doing and how you want them to feel.

You should also design your customer journey map in a way which makes understanding it easy and relevant for the people who need to use it. Make it as colourful, engaging, or illustrated as you like, as your aim should be for people to pay attention to it, so they keep how customers feel at steps in the journey uppermost in their minds.
The pay-off

Art direction doesn’t only happen within the walls of an advertising agency, or for pages of a glossy magazine. It happens every day on the web when you consider how someone will feel when they read your content, buy something in your store, or sign up for your product. If that sounds a lot like UX design, it is, but it’s also much, much more.

Art direction should be as much part of product and website design as accessibility and performance. Like them, art direction needn't be someone specific's job and should be something which everyone involved in the planning, design, and development of a product or website should engage in.

Art direction done well will make a product or website—and someone's experience of using it—better for everyone. The famous art directors of the 1930s, '50s, and '60s all wanted to make their medium better for the times they were living in. We should want to do the same. I believe we have the knowledge and tools to match the achievements of those great art directors, but in our own way and for a medium that's for our time.

And that's now.
When I book a hotel, I want to see pictures of the type of room I’ll be staying in. I also need information about prices, location, and nearby transportation; but, like a kettle and a working elevator, those essentials don’t add to the experience of my stay. With so many hotels to choose from, giving a potential customer a taste of their experience online can make the difference between someone making a booking and moving to the next hotel. Just like in the offline world, attention to small details makes a big difference. No one except marketers like pop-ups, but Volkshotel in central Amsterdam annoy visitors in style with an illustration of an arm waving a special offers placard. Try to close this pop-up, and it cheekily moves away, making me smile, at least for a moment. This is a terrific example of art direction which leaves me with a positive feeling about Volkshotel, even if I’ve not stepped foot inside it. Yet.

https://volkshotel.nl
The essence of art direction is communication, and whether you enjoy its design or not, the owners of the Hans Brinker hostel in Amsterdam have decided who they're speaking to and what they want to say:

"Quite honestly not the best, but definitely the most memorable hostel in Lisbon and Amsterdam."

Their cheeky copywriting is just one example of the clever art direction on the website. This is a budget hostel which makes no pretence about having no “swimming pool, room service, a gym, [or] tiny bottles of shampoo.”

Their bold mixture of typefaces, bright colours, and photography style are all designed to send a clear message, one that says, “We’re cheap, but if you know what to expect, that’s OK.”

https://hansbrinker.com
Data is great at giving you information, giving you knowledge; but it doesn’t give you understanding and that is its great failing. What we need is greater creativity.

— Sir John Hegarty
Principles of design

Are the principles which have guided design in other media for generations relevant to the world of digital products and websites? Of course they are, dummy. You may not be able to control the shape of someone’s browser, but the principles of symmetry and asymmetry are relevant for every screen size. Ratios and scales are centuries old, but they are as relevant to art direction for the web as they were before it was invented.

Throughout the rest of the book, you’ll find notes like this which highlight what’s important in a story and suggestions on how to use design to emphasise it.

Balance

When design elements balance, a composition looks solid and somehow feels right. People stay engaged with a design as a whole because no single part demands attention more than another. A balanced design helps people feel comfortable; but when you want to put people on edge, make designs which are unbalanced. I’m going to teach you about three ways to balance elements in a design: symmetry, asymmetry, and radial balance.

Steven Bradley wrote seven fabulous articles on design principles where he covered symmetrical and asymmetrical balance, and far more besides.

SYMMETRICAL BALANCE

For a composition to be balanced symmetrically, there should be an equal amount of visual weight on both sides of a central axis. Those axes can be horizontal, vertical, and even diagonal. Steven explained visual weight well.

66 https://smashed.by/bradleybalance
67 https://smashed.by/bradleyweight
Every element on a web page exerts a visual force that attracts the eye of the viewer. The greater the force, the more the eye is attracted. [...] There’s no way I know of to precisely measure the visual weight of a design element. You use your experience and judgment to determine which elements have greater or lesser weight.

Left: Both circles are the same size and have identical visual weight. Right: Even though these circles are different sizes, this composition is balanced across its horizontal axis.

Likewise, the following two compositions are balanced symmetrically, this time across a vertical axis.

Left: These circles are symmetrical because I place them at equal distances from the central axis. Right: While the two sides of this composition don’t mirror each other, the group of small circles has equal weight to the single large one, so the overall layout feels balanced.
ASYMMETRY

You can create asymmetry by arranging elements so there's no relationship between them, or so one side of an axis carries more visual weight than the other.

Left: One circle isn’t just bigger, it crosses the line of symmetry. Right: This overall composition is asymmetrical, although it contains one element which is symmetrical across the horizontal axis.

Although white space technically doesn’t weigh anything, it does add to the weight of elements close to it.

Left: This composition is asymmetrical because when you include the white space, there’s more visual weight on the right than on the left. Right: When there’s no relationship between colour, size, shape, or white space, designs feel like they’re full of energy.
DIAGONAL SYMMETRY

Angles can help you make a layout look less structured and feel more organic. The absence of obvious structure encourages the eye to roam freely around a composition. This movement also causes an arrangement to feel energetic.

Left: Although these circles are symmetrical across a diagonal axis, if you folded this layout along the dotted line, the two halves of the circles wouldn’t align. Right: While this composition has equal visual weight on both sides of the diagonal axis, the layout itself is not technically symmetrical.

Wheel Man: harmony or tension?

In the 1960s, two rival criminal gangs fought for control of the London underworld; the Krays in the north, the Richardsons south of the river. In my next design, a horizontal axis cuts the layout in two, just as the Thames divides north and south London.

Although it’s not explicitly stated in this next article’s text, the Krays dominated London’s criminal underworld in the 1960s. It’s important your design reflects that in some way.

These two gangs were bitter rivals. But this composition feels harmonious because both sides of the river are equally weighted and their content is almost symmetrical.
Above: The images and text describing the two gangs are weighted equally, and this symmetry suggests equality between them. (The River Thames label and compass help people unfamiliar with the ‘Smoke.’) Right: Although I arrange the elements differently for a small screen, I’m able to maintain that balance.
If you’re familiar with gangland history, you’ll know that the Kray twins, Ronnie and Reggie, were a larger firm than their adversaries south of the river. I changed the visual weight in the north of this layout — increasing the headline and image sizes — to portray this imbalance.

Don’t think that layout is only for arranging content. You can communicate as much about your content as through the pictures and words themselves.

To emphasise the imbalance of power, I make more space available at the top of this design and increase the size of images and type.

On the web, it’s rare to find pure symmetry, where every shape on one side perfectly mirrors the other. Neither of these articles below about the notorious south London enforcer ‘Mad’ Frankie Fraser is purely symmetrical, but because the visual weight of the heavier images on one side is offset by the lighter columns of text on the other, the design feels balanced.
Above: Break up large images into smaller units to balance columns of text and add rhythm to your compositions.

The first of the two designs on the right comes closer to pure symmetry by pulling the alignment of both columns towards a central vertical axis, then placing images on opposite sides.

Communicating through your choice of layout is just as relevant on store product pages. Focus someone’s attention on the most important information by placing it in the largest space.

Below: One downside of a symmetrical design is that it can feel static and predictable. When something lacks energy, it won’t keep anyone engaged for long. Bottom: Create movement by using asymmetric columns and hold people’s interest for longer.

The madness of Frankie Fraser

The first of the two designs on the right comes closer to pure symmetry by pulling the alignment of both columns towards a central vertical axis, then placing images on opposite sides.
SMALL SCREEN, BIG IMPACT

Full-height images make big impressions, especially on large screens. Delivering the same impact on small screens can be challenging, but maintaining a similar weight hierarchy helps create connected experiences across screen sizes.

Right: Viewport units make filling the height of a small screen with images easy, but that might not always provide the best experience. Far right: Filling 60% of this small screen feels more compatible with the large screen design, but there are no hard-and-fast rules, and sometimes you need to use your intuition.

Above: This design is symmetrical across the vertical axis. To balance the weight of a full-height image, I use two columns of text and a generous amount of white space.
According to Roman philosopher Cicero, ‘The face is a picture of the mind as the eyes are its interpreter.’ Readers can’t know what was in Frankie Fraser’s mind when he committed his awful crimes, but they should be able to look into his eyes.

The diagonal in this design draws the eye from the upper-left to the lower-right.

Angles and diagonals

I see designs set around horizontal and vertical axes every day, but rarely anything based on a diagonal. Every once in a while, I spot an angled element—perhaps a banner graphic with its bottom sloping—but it’s rarely essential to a design.

The dynamic nature of the digital canvas makes pure diagonal symmetry impossible.

Adding extra visual weight to the top reinforces the Krays’ dominance over the Richardsons at the bottom.
If you were able to fold your expensive screen along a diagonal, the two sides would never align. That doesn’t mean you shouldn’t use diagonals, as they’re useful for encouraging the eye to flow across and around elements in a composition.

The results can be more compelling when you combine diagonals with changes in scale. Compare the two designs below:

Because the weight of this headline and the white space around it balance the full-height image, there’s very little tension in this design.

By contrasting a large image with a small headline, I make this composition intentionally unsettling.
Principles of design

Radial balance

A design is said to have radial balance when elements radiate from a central point. Radial designs can be particularly effective when your content is more graphical, because they focus attention on critical areas.

Ronnie and Reggie Kray were at the centre of 1960s London gangland activity. Next, the twins stay dead centre while their associates radiate from them.

Radial navigation

Radial balance can also create stylish navigation elements, and a radial navigation menu ties in beautifully with the branding of my Wheel Man website.

Over the past few years, our phones got bigger, but our hands didn't. Sometimes reaching distant elements can be a stretch, but radial navigation puts links within easy reach.
Flow along horizontal, vertical, and two axes

Controlling flow is one way to lead someone's eye around a composition. It lets people know where to start, then guides them from one part to another in the direction you want them to travel. To get flow correct, you need to think hard about your content's structure and what you want people to see and do first, second, and after.

Use bold headlines and dominating images to guide the reader. Their visual weight draws attention towards a focal point; then lines, text blocks, and other directional cues let people know the direction to travel.
I use the combined visual weight of a headline and image to draw attention to the top-left corner of the composition on the right. A series of repeating horizontal text blocks then lead the reader across the page.

I include a line to reinforce the horizontal reading direction and make it easy to scan from left to right along the flow. To interrupt that movement and change that flow, use a line which runs perpendicular to it.

This vertical line forms a barrier between two areas, reinforcing the fact they contain different content.
One narrow column of running text makes the vertical flow in the this design visible, but there's another line which gives this composition energy. By matching the visual weight of headline and image, and placing both at opposite ends of the copy, I create a dynamic diagonal.

Both previous examples have one directional flow, but occasionally including two directions helps people understand the structure of a complex page. This next page (opposite) tells two stories. One—about south London's Richardson Gang—is told along a horizontal axis. The other—about Frankie Fraser, a member of their firm—is conveyed on the vertical. People could find more than one flow confusing, so provide obvious focal points and plenty of visual cues to avoid that.
A dominant headline acts as a focal point at the top of this text. To make its vertical axis more apparent, I pull the entire column up into the white space. Setting type in this column using a darker, medium-weight, sans serif makes the contrast between these two stories even stronger.

**Contrast**

For contrast to be effective, one element must look sufficiently different from another. When two elements carry the same or similar visual weight there will be no focal point. I might also use different background colours to help people distinguish between content areas.
In these two designs there’s an obvious contrast between the right-hand panel and the rest of the page. To help both sides feel connected, I match the headline colour with the panel background.
Principles of design

The typographic contrast between this headline and running text is lower. Understated typography plays second fiddle to the large image which dominates this design.

Don’t make people work hard to find the most recent news article or a featured product in a store. Instead, use contrast to make them obvious.

TYPE SIZE CONTRAST

When your aim is to make people feel at ease, a typographic design should support your content, not make a statement of its own. A lower contrast between text sizes makes this next design feel academic and conservative. At other times, make a feature of your type. Add drama by using a steeper typographic scale.

You shouldn’t underplay the impact gangland enforcers like Freddie Foreman had on London’s criminal underworld. The contrast between headline and running text is much stronger here to reinforce that.
CONTRASTING TYPEFACES

One of the most effective ways to add contrast is by using strikingly different typefaces for headlines and running text. Choose a headline typeface which makes a bold statement and reflects your content. Smaller type sizes—in some cases, much smaller—can make as significant an impact as large type. I reduce the size of this headline, but it’s the size of the text opposite which tells the real story.

Freddie Foreman is a convicted English gangster. He has been tried in two separate occasions for murder and has convictions for armed robbery and for disposing of a body. He was involved in the disposal of the body of Jack “the Hat” McVitie and was sentenced to ten years in prison for the crime.
This design becomes more interesting when I make that paragraph smaller and centre it horizontally and vertically.

Harmony and tension

Although infamous for their occasionally violent confrontations, twins Ronnie and Reggie Kray were inseparable until both were sentenced to 30 years in 1969. They spent the rest of their lives in prison.68

Balance creates harmony; contrast adds tension. This layout brings the twins face-to-face to portray the closeness of their relationship. Solid blocks of running text and a vertical headline combine to form a solid unit which represents their bond.

68 Reggie was released on compassionate grounds after serving more than his minimum 30 years life sentence. He died from cancer two months later.
Over 30 years in prison separated Ronnie and Reggie. The edited version of their photograph opposite makes people feel uncomfortable because the twins now face in opposite directions. To dial up the tension, I shift my vertical headline to leave a void between them.

How the deep bond between Ronnie and Reggie Kray contrasts with spending most of their adult lives apart, was one of the most fascinating aspects of the twin’s story. Your design should explore that.
While their actions brought about their eventual downfall, it was Detective Chief Superintendent Leonard “Nipper” Read who finally came between the twins.

I demonstrate how Read forced Ronnie and Reggie apart by placing him at the centre of this next layout.
I use a sans serif typeface for Read’s story and a serif for the Krays to emphasise the difference between their stories.

Inspector Leonard “Nipper” Read of Scotland Yard was promoted to the Murder Squad and his first assignment was to bring down the Kray twins. It was not his first involvement with them. Eventually, a Scotland Yard conference decided to arrest the Krays on the evidence already collected, in the hope that other witnesses would be forthcoming once the Krays were in custody. On 8 May 1968, the Krays and 15 other members of the Firm were arrested. Exceptional circumstances were put in place so as to stop any possible cooperation against any accused.

Ronnie Kray shot and killed George Cornell—a member of the rival Richardson gang—at the Blind Beggar pub in Whitechapel on 9th March 1960.

The day before, there had been a shoot-out at a nightclub in Catford involving the Richardson gang and Richard Hart—an associate of the Krays—who was shot dead. This public shoot-out led to the arrest of almost all the Richardson gang. Cornell was not present at the club during the shoot-out.

Whilst visiting hospital to check up on his friends, he visited the Blind Beggar which was only a mile away from where the Krays lived.

Ronnie was drinking in another pub when he learned of Cornell’s location. Ronnie went into the pub with Ian Barrie, walked up to Cornell and shot him in the head. Barrie—confused by what happened—fired five shots in the air.

Just before he was shot, Cornell remarked, “Well, look who’s back.” He died at 6:00am. Ronnie Kray was already suffering from paranoid schizophrenia at the time of the killing.

After his wife Frances’ suicide, Reggie Kray was allegedly encouraged by his brother Ronnie to kill Jack “the Hat” McVitie, a minor member of the Kray gang who had failed to fulfil a £1,000 contract—half of which was paid to him in advance—to kill Leslie Payne. Shortly after Reggie’s death, Freddie Foreman revealed that McVitie had a reputation for leaving carrage behind him due to his consumption of drugs and heavy drinking. He had in the past threatened to harm the twins and their family.

McVitie was lured to a basement flat in Evering Road, Stoke Newington on the pretence of a party. Upon entering, he saw Ronnie seated in the front room.

Ronnie let loose a barrage of verbal abuse and cut McVitie below the eye with a piece of broken glass. It’s believed that an argument broke out between the twins and McVitie. Reggie pointed a handgun at McVitie’s head and pulled the trigger, but the gun twice failed to fire. Reggie was handed a carving knife and stabbed McVitie in the face and stomach.
Scale

Some design elements stand out because they’re far bigger, others because they’re much smaller. Knowing how scale influences someone’s understanding of a story is essential to telling it well.

Are you really looking at a massive baby? Or is he just closest to you? Is the Mini a full-size car or a toy? Is a Mini bigger than an elephant? Not last time I checked, but if you’re telling a story about the Mini’s spacious interior, the message becomes more important than reality.

Making an element large lets people know it’s important and attracts attention. It also creates a visual hierarchy which helps people understand the order and structure of your content. Sheer size is rarely enough on its own, so to make a significant impact, largeness must be relative to the size of other elements.
To provide scale, I contrast the size of this Mini Cooper with smaller than average text about its inventor.

Left top: The images of these two cars are identical, but the impact the one at the top makes is lessened because the Mini and text are equivalent sizes. Left bottom: I make the Mini appear much more significant by reducing the amount of running text.

Right: To maintain the scale when space is limited, I use a different large image and include a horizontal scroll.
This mini Mini makes an impact far more significant than its diminutive size. In fact, it's the perfect counterpoint to my large block of running text.

**PROPORTIONS**

Don't leave relative sizes to chance; base proportions on a system instead. A little maths can go a long way to ensuring that your design's proportions look natural and feel right.

You can divide any page into sections using proportions which have served architects and artists for centuries. Don't turn the page. I'm not planning on teaching Ancient Greek architecture. I won't mention the hypotenuse, not even once.
RATIO-BASED DESIGN

When you're working out the most natural-looking relationships between elements, root rectangle proportions are incredibly useful. You can divide them into smaller rectangles which have the same ratio of height-to-width as the original. That might sound complicated, but these rectangles are simple to work out.

Start with a square and then draw a diagonal line from the bottom-left corner. Imagine that line is a piece of string. Swing it in an arc and where it meets the bottom of the rectangle, that’s the edge of your new root-2 rectangle.

Jay Hambidge’s book, The Elements of Dynamic Symmetry is the definitive guide to geometry and ratios — if you have a PhD in mathematics. If you don’t, enter a length into Scott McDaniel’s Dynamic Symmetry Calculator and it will do the maths for you.⁶⁹

Left: This running text fits neatly into the narrower column. I determine its width by making a root-2 rectangle. The image and text alongside are perfectly proportioned. Top right: Root-2 rectangle ratio: 1:1.41. Bottom right: Root-2 rectangle with smaller root rectangles inside it.

⁶⁹ https://smashed.by/dynamiccalc
A root-3 rectangle with a ratio of 1:1.73
Two columns of running text suits this ratio perfectly. Did you notice I scaled the heading's font size using the same ratio?

You can use that root-2 ratio to determine a relationship between the width of an image and a column of text alongside.

Root-3 rectangles are longer. To make one, draw a new line diagonally between the corners of that first root-2 rectangle and then arc it down again. Where the line meets the bottom of the rectangle, that’s the edge of your root-3 rectangle.
You shouldn’t be surprised to learn that to create an even longer root-4 rectangle, you can follow the same pattern. What could be a better use for these three perfectly proportioned columns than to fill them with text?

**GOLDEN RATIO**

Unless you’ve been doing a 4,000 year stretch at Her Majesty’s pleasure, you’ll know about the golden ratio. It’s where one dimension is 1.61 times longer than another. It’s common to see this ratio illustrated as a 1:1.61 rectangle with a square inside it. This is where maths gets interesting. When you remove a square from a rectangle which has golden proportions, what you’re left with is another, albeit smaller, golden rectangle. Do that again, the same result. And so on.

The golden ratio is a popular way to determine relationships between elements. Working out the proportions of a golden rectangle is easy as there are plenty of tools online to help you with your 1.61 times multiplication. If you fancy making your golden rectangles the old-fashioned way:

1. Divide a square in half.
2. Draw a line from the bottom-centre to the top-right corner.
3. Swing the line down in an arc.

*According to Luca Pacioli—a contemporary of Leonardo da Vinci—“Without mathematics, there is no art.”*
Where the line meets the bottom of the rectangle, that’s the edge of a golden rectangle.

**Golden rectangle ratio, 1:1.61**

Use the larger of these two columns on the left, or reverse this arrangement by placing the smaller column on the left to inspire an altogether different layout.

Left: Alternating between layouts of the same proportions keeps people engaged, but be careful not to confuse them by moving important functionality. Right: You might spot that the width of the Mini’s specifications table is the same as the columns of running text.

The golden ratio can also help when you’re designing for portrait orientations. These layouts fit tablets’ medium-size screens perfectly, but they can define spaces on a smartphone screen too.
RULE OF THIRDS (FOCAL POINTS)

To help people focus on the most important parts of a composition, you can borrow the rule of thirds technique from photography. This principle involves dividing a layout into nine segments by drawing two horizontal and two vertical lines at even distances apart. A composition's focal points are where those lines meet.

Left: I use the bottom two-thirds of this medium size layout for my headline and running text. Centre: By flipping the layout vertically and placing both headline and image into the upper two-thirds, this design has a very different look. Right: The golden ratio defines space even when it's limited. This crop-top Mini occupies the upper third.
Focal points are equally useful in portrait orientations. Combined with a little imagination, they can often inspire unusual designs.
FIBONACCI

Despite having a name like a mafia mobster, Fibonacci was an Italian mathematician of the early Renaissance, famous for his number sequence. In Fibonacci’s sequence, every number is the sum of the two previous two numbers, which goes like this:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144…

And so on. Fascinating, if you’re a mathematician.

Designers have every reason to be fascinated by the Fibonacci sequence too, as its closeness to the golden ratio has inspired them, artists, and architects for hundreds of years.

You might have seen this spiral overlaid on everything from Twitter’s logo to Donald Trump’s comb-over.

You’ve probably seen Fibonacci’s sequence expressed as a spiral, but it’s his numbers I find most useful. The sequence’s similarity to the golden ratio makes it easy to determine proportions which are visually appealing for everything including font sizes and leading, gutters and margins, and of course, the structure of an entire page.

If you overlay a Fibonacci spiral over a grid made up of squares, you’ll find as the curve opens up, you can fit more squares underneath it.
Left: The quantity of squares increases in the same order as Fibonacci’s sequence. Now that is fascinating. Right: These squares make many unique layouts possible.

These perfectly proportioned squares offer enormous flexibility for making designs where relationships between elements feel natural. Use as many squares as you need and arrange them in every combination you can think of.

The starting point for this unique layout was one of the (probably) millions of potential combinations. This Fibonacci-inspired grid is much larger than the visible area, and this alone makes it seem full of energy.

Left: This approach to layout can result in designs as playful as the original Mini Cooper. Right: There’s more to small-screen layout than a single column. All you need to make mobile designs engaging is a little imagination and an 800-year-old Italian.
I use the two largest squares to define the size of the photographs of my Mini's front and rear. Copy then fits neatly into a smaller square alongside. Content needn't be confined within a single square. I use the L-shape combination of three squares to determine the size and position of this Mini Cooper headline.

We needn't lose playfulness just because there's less space available on small screens. In fact, on a small screen, a playful design can engage people just as much. For this mini design, I reposition that same arrangement of squares to give me the unusual spaces I need for my content.
Directing grids

Grids have a long and varied history in design, from the earliest books, through movements like constructivism and the International Typographic Style,\(^7\) right up to the present-day popularity of grids in frameworks like Bootstrap and material design.

Layout systems

A generation of product and website designers have grown up with grids from Bootstrap, 960 Grid System\(^7\) before it, and Blueprint before that. In frameworks like these—and in plenty of work built on them—grids are used mostly for just aligning content to the edges of columns.


\(^7\) https://smashed.by/its
\(^7\) https://960.gs/
When you use grids imaginatively, they do much, much more than align content. A grid brings cohesion to a composition. It helps people understand the stories you’re telling by suggesting hierarchies. Grids inform people what to read first, then next, and how much attention to give it.

They define the position of valuable information or a call to action. A thoughtfully chosen grid leads to a wealth of possibilities and any number of exciting designs.

I’m going to teach you there’s more to grids than twelve or sixteen evenly sized columns. I’ll show you how varied layouts based on one, two, and multiple columns can be. I’ll explain how to use modular and compound grids, which are two or more grids on one page. Finally, I’ll demonstrate how to get incredible results from an off-the-shelf framework. Before I do that, I want to explain grid anatomy and terminology.

**Grid anatomy**

There are several components to a grid: columns, fields, flowlines, gutters, margins, modules, and spacial zones.

*Every grid contains the same components. How you use them will depend on the goals for your design.*
**Columns** are perhaps the best-known grid components. They’re vertical. They can, but needn’t, be the same width. You can use as few or as many of them as you need. That choice will depend on your design goals.

![Columns diagram]

**Horizontal fields** are known as rows in CSS Grid—when your writing mode is set to horizontal, top to bottom,—and in HTML table parlance. By default, they run left-to-right and perpendicular to columns. Their heights can be intrinsic—defined by the content inside them—or you can give them an explicit height.

![Horizontal fields diagram]
**Flowlines** divide space into horizontal sections. You can use them to guide someone from one content area to the next. When flowlines are solid and extend to the outer edges of the page, they make people pause before starting a new section. You can anchor blocks of content to either the bottom or top of a flowline using CSS Grid or Flexbox alignment.

Gutters are horizontal and vertical gaps between columns and fields. You can leave them empty or incorporate them into the size of elements on the grid. Gutter height and width can have an enormous impact on the clarity of your layouts. Make gutters too small, and you run the risk of making your copy dense and difficult to read. Too large and you might lose the visual connection between columns.
Unlike CSS terminology—who every element has margins on all sides—grid margins are the spaces between the outer edges of your layout and the browser chrome, page, or screen. Don’t think these margins need to stay empty though, as they can be useful spaces for placing image captions and small text.

When you hear the term module, do you immediately think of a reusable atomic web design component? Grid modules are different. They’re individual units—most commonly rectangles or squares—which repeat horizontally and vertically. Modules can be any size: based on your content, the aspect ratios of images, and even advertising sizes.

In CSS Grid, spacial zones are called grid-template-areas. Whatever you choose to call them, they’re adjacent modules bound together to form larger areas.

You needn’t set the same margin on every side. A deeper margin at the top or bottom, or a wider one on the left or right, can make a big difference to the feel of your design.
Grid types

When I teach design classes, I ask my students to draw what a grid means to them. Nine out of ten of them sketch columns, but it’s important to know that column-based designs are only one of several options.

Columns offer enormous possibilities, so it’s little wonder they’re so many peoples’ go-to grid components. You can use as few or as many columns as you need and you can make them any width. Not every column needs to be as wide as its siblings, you can bind columns together to form wider containers, and you can also leave columns empty.

SINGLE COLUMNS

A single column is the simplest form of grid. While its roots reach back to the design of early manuscripts, more recently it’s become a popular way to avoid the complexities of responsive layouts.

The width of one column will depend either on the length of its contents, or the browser, device, or page around it. There may also be times when you’ll choose an unnaturally narrow column plus extra-wide margins to create drama.

A single column, one-eighth of the available width of my layout.
Often you’ll choose column widths which make your text comfortable to read. Not too narrow so jumping between lines becomes tiring; not too wide so finding the start of the next line is difficult. The ideal width—or measure—of a single column is something book designers have considered for generations, so it’s not surprising that designs which are based on them look classical.

In this single-column design, running text holds around 75 characters per line, which is the upper limit for reading comfortably. I always adjust type size when column widths change, making text smaller in narrower columns and larger in wider ones.

**TWIN COLUMNS**

Grids with two symmetrical columns feel orderly. They can hold a tremendous amount of content without becoming overwhelming, especially when you set those columns within wide margins. If you have plenty of running text, twin-columns solidify it into blocks, giving your page visible structure.
While twin-column grids might feel conservative, you can add playful touches, perhaps by pulling elements into margins or staggering the start of each column by padding its content.

I want this next design to feel more dynamic, so I pull the image left by the same width as gutters between columns. This creates a diagonal that draws people towards where they should start reading.

UX designers are concerned with making a product or website easy to use, so make it obvious where you want people to look and where to start interacting or reading.

Adapting a twin-column layout to tiny screens is straightforward. Just allowing the second column to drop under the first makes for an uninterrupted reading experience.
Directing grids

Two asymmetrical columns

While symmetry helps large amounts of running text feel approachable, an asymmetrical two-column grid — where one column is wider than the other — often works better when your content has more variety.

Vanish for good

A good cover can be essential to your new life on the run. Cover identities can take months of planning and preparation to pull off.

It’s a new life that isn’t easily tried your way, device or a date of birth with country you’re from and how you’re traveling — all details you’re used to remember when you’re crossing borders. Choose a cover you can easily relearn if you’re questioned. You should be able to converse as your cover without any difficulty.

Tell a story of your birth date, your new profession, where you live, your future. You should be able to converse without hesitation. You should be able to discuss your cover’s profession fluently. If you’re a web designer, you should be able to discuss the language of HTML and CSS. If you’re new in business, you should be able to discuss balance sheets and I.T.

Don’t be afraid to ask people or share your experiences, asking where it works with your cover. The most convincing stories are the truth. Telling the truth whenever you can gives you a sense of you’re more believable.

Vanish for good

This arrangement is ideal when images and captions need to connect to the story. You might notice that to emphasise this connection, I aligned the top of the suitcase with the first line of my running text.

Vanish for good

A little space makes a big difference to even the simplest designs.

To add drama to this symmetrical, twin-column design, I pad the start of one column, then the end of the other.
A larger column offers plenty of space to tell a longer story. A smaller one allows image captions or notes to sit alongside so they can support your narrative. A narrow column is also a fabulous place for supporting information: content which is different from, but still relevant to your story.

Whether you decide a narrow column should sit on the left or right will depend on your content. I place a large block of running text in the wider column on the right, then supporting information in its narrower sibling on the left.

Moving the narrow column to the right gives my design a very different feel. This difference is important because here the content in this narrow column is separate from the story.

MULTI-COLUMN GRIDS

Symmetrical multi-column grids have become a staple largely because twelve columns can be easily divided into thirds and quarters. In fact, they’re now so ubiquitous that starting a new design without sketching three or four columns can be hard.
Just because so many people choose multi-column grids doesn’t necessarily mean designs based on them need to be predictable. There are plenty of ways to make multi-column layouts look unpredictable. Above, this oversized image and three asymmetrical columns give the impression that the bulging suitcase is pushing my content apart. The layout then switches to three symmetrical columns under an invisible flowline.
Remember, you needn’t fill every part of every column. Leaving columns empty can turn a highly structured layout into one which feels dynamic.

**MULTI-COLUMN GRIDS AND FLOWLINES**

On the web, you rarely know how tall someone’s window on your content will be. Tall columns of text can quickly grow beyond the viewport, which makes reading difficult. You need to think carefully about content structure before setting long passages of running text in columns.

Flowlines separate a composition into sections, but they are also useful for ensuring that blocks of running text are more likely to fit within one screen. You can use images and text elements as flowlines to define these sections.

**SYMMETRICAL LAYOUTS**

When a design calls for two or more columns, you’ll need to decide whether to arrange them asymmetrically or symmetrically. This decision will largely depend on the content you’re delivering and the complexity of your design. It will also depend on how you’d like someone to feel, because symmetry brings a calmness which makes someone feel comfortable.

Symmetrical layouts suit simpler designs with less varied content because symmetry makes it easier for someone to view your composition as a whole. That doesn’t mean every part of a layout must mirror another precisely. Deliberately misplacing some elements will make designs more compelling.

I place this larger standfirst paragraph diagonally opposite my headline to contrast it with the blocks of running text below.
ASYMMETRICAL LAYOUTS

Asymmetry helps organise a wider variety of content in complex arrangements. It can also direct someone's attention to specific areas in a page and help them make sense of your content. Asymmetrical layouts allow for a wide variety of sizes and shapes. Asymmetry brings energy to compositions and makes them feel active.

Solid blocks of copy and images add visual weight and draw the eye. To make a design more interesting, alter contrast between larger, heavy elements and lighter, smaller ones.

This next design includes various types of content. There's a headline, image, running text, plus two pieces of supporting information. I want the content hierarchy to be obvious, so I devote more space to running text and image, less to supporting information. I choose four columns and arrange them in an asymmetric 1|3|2|2 pattern, where each unit is an eighth of my layout.
These proportions work even better when I reorder those columns into a 3|1|2|2 pattern. Moving the vertical headline into the centre of this composition further separates the story from its supporting information.

By now, you could be wondering how good asymmetrical layouts look on smaller screens. Even in landscape orientation on a tablet's screen,
There's still room in this layout for a fabulous vertical headline. I resize it to match the viewport height.

the narrowest column isn't wide enough to give my headline the visual weight it needs. Instead, I place a horizontal version of that headline in the space at the top of the page. With space in portrait orientations at an even greater premium, I dropped the supporting information below my main story.

**Compound grids**

A compound grid is two or more grids of any type—column, modular, symmetrical, and asymmetrical—on one page. They can occupy separate areas or overlap. In this design, six columns overlay four to create interesting ways to align and size content.

The flexibility of a compound grid becomes obvious by making the grid lines behind this layout visible. The interplay of two grids—and how that affects the position of a headline and the width of my small text—makes this compound layout more interesting than one grid in isolation.
A 6&4 compound grid, famously used as the basis for Karl Gerstner’s work on Capital magazine in the 1960s. This grid makes an incredible variety of compositions possible.

In this version of the same article, I split running text across two of the four columns, then move the supporting information to one of the narrower columns on the six-column grid.

A good cover can be essential to your new life on the run. Cover identities can take months of planning to pull off.

Vanish for good

Pick a name that won’t easily slip out of your mind. Choose one that is easy to remember when you’re creating a cover. You should be able to remember it as easily as your own name. If you’re a writer, you should be able to discuss your cover’s effectiveness. If you’re a writer, you should be able to discuss your cover’s effectiveness.

How to disguise yourself

Lining up your trip after confirmation? No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid. No need to be afraid.

Vanish for good

Before you leave for an airport, get away disguised as a business person, think about how someone who fies every week would deal with building a plane, or going through security. They’re likely to be the same person, and it’s a lot of work and it took them time. Look around in the airport, and think about how they behave. Watch them, and you can’t be fooled.

Learn as much as possible

The best way to maintain a cover is being able to avoid suspicion. If you can’t change, convince a fellow traveler that you’re traveling for business, they may ask about your job and how long you’re in the area. If someone asks, “Where are you going?”

I align the headline to the start of a column on the six-column grid. The small text on the left also gets its width from one of those six columns.
With a compound of two grids, you might use widths from one or the other. Or you could combine widths from both to form columns which don’t conform to either. You can use these new widths to inform the sizes of images and text. This pull-quote gets its width from starting on the four-column grid but ending on the six.

In addition to the pull-quote, this headline has an unusual width because it begins on a line from one grid but finishes on the other.
Above, I set the running text in two out of four columns. However, when I place pull-quotes in the outer two columns, they feel disconnected from the story. I can improve this design by aligning the edges of these quotes to lines in the six-column grid so they feel part of the story.

You can afford to be creative when designing block and pull-quotes. I’ll teach you how later.
Compound grids make informed decisions about the widths of images and captions easier. Below, these columns of running text derive their widths from the four-column grid, as does the banner image. On the other hand, I decided the width of this image caption should be the same as one column from the six.

Within a symmetrical design, small details such as image captions can make layouts feel more active.
STACKED GRIDS

You can combine column grids with hierarchical and even modular grids. When pages contain two separate subjects or different types of content, stacking grids can be a great way to make that difference more obvious.

At the top of this next page is a story about creating a new identity after pulling off a robbery. Underneath there’s an altogether different story about why the Mini Cooper is your best choice for a getaway car. To leave someone in little doubt these stories are separate, use a different grid for each. I base the top story on a four-column grid, the bottom on a six.

I make the difference between these two stories obvious by placing the second against a grey background. I also use paragraph spacing instead of first-line indentations.

Modular grids bring order to complex content and are a fabulous choice for the timeline of Mini models at the bottom of this stacked design. The modular timeline layout contrasts beautifully with the columns in the story above.
STACKING ON SMALL SCREENS

When people use small screens, you can’t rely on complex compositions. You’ll need to find other ways to make people notice differences between content types. The position of these two images emphasises the difference: for the first, the image bleeds off the right of the page; in the second, it bleeds off the left.

You might choose different typefaces, styles, or weights to emphasise different types of content. Varied alignments and spacing can also be incredibly effective.
Modular grids

A module is a rectangular or square unit of any size. Modules repeat horizontally and vertically to form a modular grid. At first glance, modular grids can seem complicated. However, they’re easy to work with, so it surprises me so few web designers use them. Modular grids are excellent for bringing order to large amounts of varied content and you can also use them to create visually appealing layouts when there’s very little content. When you use modular grids thoughtfully they can fill your designs with energy.

In this next design, I bind several modules together to create three larger spacial zones. These define the position and shape of my content. The zones inform the size of headlines, images, and even the detached image caption which occupies a single module over on the right.

Because it contains fewer, more substantial, spacial zones, this next composition (opposite top) feels altogether different. The smaller of two zones informs the position and size of a headline and standfirst combination. Whereas in the previous design I use a single module to determine the size of a caption, here that module provides dimensions for a small image.
As well as using modules to inform image proportions, you can also use them to determine the size of decorative text elements, including drop caps.
ADVERTISING SIZES

If a product or website relies on advertising, you might choose to base its module proportions on Interactive Advertising Bureau (IAB) unit dimensions so that ads integrate into your composition seamlessly.

Banner ads are sometimes a necessary evil, so I base the design of this modular grid on a 970×90 IMU (pushdown).
ASPECT RATIOS

Large images needn’t always run horizontally across a page; portrait orientations can make a design look different and distinctive. I base this grid on a 16:9 aspect ratio, but this time stood the image upright by binding nine modules together to form a large, vertical spacial zone.

A good cover can be essential to your new life on the run. Cover identities can take months of planning to pull off.

Vanish for good

A smart name that won’t easily trip you up. Decide on a date of birth, which country you’re from and everyone’s handling—will details you’re going to make your cover look right, your border. Choose a cover you can easily maintain your position. It should be able to turn your cover without any difficulties.

Details about your birth, date of your new creation; when you like your favorite drink. It should be able to your name—profession, profession. If someone receives; the language of HTML, and photography. If you’re new or in business, you should be able to discuss below sheets and PDF. Don’t see what text we can impose our previous.

Vanish for good

Even with an image so large, this modular grid still gives me the luxury of dividing my running text into two symmetrical columns.
For a tablet's medium-size screen, reorientate the image to landscape format and make the best use of the available space on those devices.
WHICH GRID TO SELECT

How do you choose the grid type which best suits the design you're making? Some grids provide structure and they order content; others feel more organic. Sometimes a grid will be obvious, other times it will fade into the background. Images are occasionally more critical than text, and you may want to display them at large sizes for maximum impact. Your choice of grid will depend on all these factors and more, but in general:

- **Single columns** have a classical feel and are suitable for long passages of running text. Ideal for blog entries, news articles, and help text.

- **Twin columns and multi-columns** are the most flexible and best for helping people make sense of varied content. Perfect for digital products and stores.

- **Modular grids** are fabulous for making diverse designs with plenty of drama and energy. Useful for portfolio and product pages.

Frameworks needn’t mean boring

There’s no doubt that Bootstrap and many design frameworks like it are popular for getting a project started. Even if people don’t use Bootstrap’s .row and other classes, its twelve symmetrical columns are where many people begin.

When he wrote “Beyond The Boring: The Hunt For The Web's Lost Soul”, Josh Johnson thought that frameworks mean we've “reached a stagnation point where unique layouts are a lost art.” But frameworks don’t make websites boring. Designers do.

If you want a framework-based design to be more interesting, start by simplifying its grid. You can easily streamline twelve narrow columns into six wider ones, which is good because I can’t think of many reasons to use all twelve.

[73](https://smashed.by/beyondboring)
When you stop thinking of framework columns as alignment guides, you can start using them more imaginatively.

I find using fewer columns helps me focus on the whole composition. To create a feeling of rhythm in this framework-based design, I split two of my simplified columns to form a repeating 1|1|2|2 pattern.

Experiment with splitting, then recombinining columns to create more compelling layouts.
These narrow columns don’t contain content of their own. Instead, they augment the width of other columns and inform the size of images and captions. The sizes of this suitcase and image caption are informed by adding one narrower column to a wider one. The main content is one narrow column plus two wider ones.
Sometimes, all it takes to get a different feeling from an asymmetrical design is to flip the grid. In that last example, this forms a 2|2|1|1 repeating pattern that I use to size a pull-quote and make it stand apart from the columns of running text around it.

**Diverse grids**

When I described compound grids, I showed how the interplay between two or more grids can lead to interesting designs. This type of flexibility is essential because sometimes a design requires a layout that’s not possible from a single category of grid.

**GRID COMBINATIONS**

My next design breaks a page into three horizontal sections. Within each section are different numbers and widths of modules. When I switch on the lines which turn those modules into columns, the possibilities of this grid appear.

> While I take an organic approach to creating space, the result still maintains a geometrical structure.

Sometimes, to make a design engaging, you’ll want to break with convention altogether by encouraging people to explore a layout and discover content instead of laying it out for them.
My goal in this design is that no two blocks of content should be the same shape or size, but there should be a relationship between them. These content areas look randomly scattered even though their placement was deliberate.

These grid combinations suggest irregular positive and negative spaces in and around my content.

I start this design with a five-column symmetrical grid, then layer a root-3 rectangle, the width of four of my columns. The next step is to duplicate that rectangle and place it diagonally opposite to create two overlapping rectangles. I then copy the first rectangle a third and final time and rotate it by 90 degrees.

These three intersecting rectangles create a variety of shapes and sizes and countless layout permutations.
Still a fan of Bootstrap’s columns? You can create diverse grids from that too.

**Do disappearing right**

To do disappearing right, you need a good “go bag” to make your break. Change your appearance. Dye your hair, if you’re a natural blonde go brunette and vice versa. Ditch your favourite t-shirt and jeans. Go to a cheap clothing store and buy outfits you’d never normally choose.

Adjust the size of the root rectangles to match the width of eight of Bootstrap’s twelve columns and make spacial zones which feel balanced on its grid.

Reposition those root rectangles anywhere on the grid to open up a wealth of new opportunities from Bootstrap and frameworks like it.
ROTATING GRID COMPONENTS

When you want a design to have an off-kilter feel, rotate one or more elements. I base this group of images on a modular grid derived from their 16:9 aspect ratio. Then I turn each image by a different number of degrees.

Using images of various sizes and layering them on top of each other creates a sense of depth and gives your design a more realistic look.

Try rotating one of two columns from a twin-column symmetrical grid to give you the perfect balance between order and playfulness.
Lo-Flo Records’ website is artfully directed in so many dimensions, it’s difficult to know where to start describing it. The modular grid layout has a musicality which perfectly suits its subject matter. Its images and text elements have been arranged in patterns which are structured, but at the same time unexpected. However, it’s when you interact with those elements when the design comes to life. Each page is filled with well-orchestrated animations which bring their design to life.
Animations like these can quickly become repetitive if they’re not carefully considered. On Lo-Flo Records, these subtle enhancements move in a way which perfectly suits the company’s tagline, “Go with the flow.” The changing geometric shapes and even the use of sound when hovering over a link all make someone want to explore the website with their ears as well as their eyes.

http://loflorecords.com
When you want even more drama, a modular grid’s structure is ideal for rotation. This grid includes two spacial zones and a single module. I rotate the entire composition just enough to achieve a dramatic effect, without making reading too difficult.
Directing type

In the previous chapter, I taught you there’s usually no reason to fill every pixel with content as white space is equally crucial to the success of a design.

Designing white space

White space makes a design feel luxurious, suggesting you needn’t use every pixel. Combine white space with multiple columns and your designs will immediately take on an editorial feel, reminiscent of a quality print publication. White space can also direct people to and then through your content. That’s why it’s important you understand the impact this negative space can make.

HEAD MARGINS

A page with large blocks of solid copy can sometimes be off-putting to a casual reader. One way to help people feel comfortable is to add deep margins before the start of your content.

Left: The small margins around these columns make the composition feel serious and their content a little daunting. Right: By deepening the margin above this content, this design now feels more casual.
WHITE SPACE UNDER HEADLINES

You can use tall spaces to separate headlines from running text, which will encourage people to pause, preparing them to read what’s next.
Fleeing to another country

Thanks to the dark web, buying a fake passport has never been easier.

The UK Home Office defines extradition as “the return of a wanted criminal from a country where he or she is found, to the country where he or she is accused of, or has been convicted of, a criminal offence. It prevents criminals escaping punishment by fleeing to another country."

Plan to escape to somewhere that doesn’t require an entry visa and has no extradition treaty. Countries like Egypt, Iraq, and Pakistan are not politically friendly with the UK, but may not be your first choice of places to live out your life on the run.

The UK has extradition agreements with 105 countries. There are 33 countries with which Britain doesn’t have an extradition agreement, so these would be great destinations: Afghanistan, Armenia, Azerbaijan, Bahrain, Belarus, Bhutan, Cameroon, China, Dominican Republic, Egypt, Ethiopia, Georgia, Iran, Japan, Kazakhstan, Kuwait, Kyrgyzstan, Madagascar, Mongolia, Namibia, Oman, Pakistan, Qatar, Russia, Saudi Arabia, South Korea, Tajikistan, Turkmenistan, UAE, Ukraine, Venezuela, Yemen.

Escaping to a non-extradition-treaty country may not mean you’ll be free as a bird, as you’ll be subject to the laws of your new country.

It’s important to know that there’s a big difference between forging documents like birth certificates, drivers license, and passports, and “ghosting” someone’s identity. Forged documents will only get you so far and won’t help you get a new social security number.

Readability

I love designing with type. But this isn’t a book focused on web typography – Richard Rutter’s already done a fabulous job with that. I’m going to explain how typography relates to art direction. First, some basics.

74 https://smashed.by/webtypographybook
MEASURE AND COLUMNS

What you probably know as line length, print designers and typesetters call the measure. Have you ever wondered why? Fun fact: this term comes from hand-setting type, and a tool called a composing stick. Typesetters use these sticks to set out a precise number of metal characters to fit their columns. Because column widths change, composing sticks have one adjustable end, and a scale to measure line lengths.

You can set the measure using any CSS absolute length unit—millimetres, centimetres, inches, picas, pixels, and points—or relative units (ch, em, ex, rem, vh, and vw).

To ensure type is comfortable to read, there's an ideal width for a measure that's defined by the number of characters in a line of text. In his book Web Typography, Richard recommends we aim for between 45–75 characters; although he suggests that research also indicates line lengths up to 100 characters don't negatively affect readability.

To demonstrate just how awkward a too-long line of text can be, here's a wide measure, 120 characters, set at a font size of 18px. I've included an image for scale.

On the formation of Imperial Airways—Britain's first national airline—In 1924, Croydon became the new airline's operating base. It was from Croydon that Britain first developed its European and long-haul routes to India, Africa, the Middle and Far East, Asia, Africa and Australia. The Croydon Aerodrome Extension Act led to large scale expansion, redevelopment and construction of an improved new airport. The airport was greatly enlarged between 1926 and 1928, with a new complex of buildings being constructed alongside Purley Way, including the first purpose-designed airport terminal and air traffic control tower, the world's first airport hotel, and hangars.

With a measure that's 120 characters wide, reading more than a few lines quickly becomes tiring. The eye has a long way to travel before starting a new line.

By doing nothing other than increasing text size, you can make reading a wide column a little more comfortable, but still not ideal.
Directing type

105 characters (including spaces) set at 20.49px.

Increase text size again, and this column now measures a comfortable 75 characters. Someone's reading experience just improved dramatically.

75 characters (including spaces) set at 28.34px.

Increasing text size is just one way to optimise a measure. At smaller sizes you can set type across multiple columns too.

50 characters (including spaces) per column, set at 20.49px.
The general rule of thumb is to reduce the size of your text when you make columns narrower. This helps maintain a measure that’s comfortable to read.

50 characters (including spaces) per column, set at 20.49px.

**TYPOGRAPHIC SCALE**

By how much you change text size will depend on the typographic scale you’ve chosen. These scales help make a typographic design feel balanced as every scale has a ratio which defines how large or small one text size appears in relation to others.

Working out the sizes in any given scale isn’t tricky. There are several tools available, including Tim Brown’s Modular Scale, and Type Scale, which has a handy preview for experimenting with different scales.

Choose a shallow scale when you want a conservative feel. Other times, a steeper scale will add drama. Knowing which scale is appropriate to tell your story is one of the most essential skills for a typographer.

Contrasting headline and body sizes is essential when establishing a typographic hierarchy. Lower contrast feels more academic and authoritative; higher contrast looks more adventurous, somehow daring. Designs based on these scales feel dramatically different.

---

75 http://modularscale.com/
76 https://smashed.by/typescale
Amy Johnson flew solo from the top of the world to down under. Her daring flight was not (then) an everyday occurrence and deserves more than an everyday design.

Left: The shallower curve in this typographic scale feels academic and conservative. Right: By switching to a steeper curve this design feels more confident and even a little daring.

While this running text stays the same size, headlines increase according to your chosen scale.

When using the steepest scale (3), type sizes quickly escalate. In the most gradual scale (1) there’s less difference between the smallest and largest sizes across the same number of increments.

Earlier I taught you how increasing type size makes reading text set in wider columns more comfortable, and why this is especially important for the variety of screen sizes we deal with. When making text larger, one option is to stick to the same scale, but move up by one or more increments.
Chapter 7

(A) Smaller text suits the measure of this narrow column. (B) Staying on the same gradual scale (bottom), this type size increases by one increment.

Croydon Airport

On the formation of Imperial Airways—Britain’s first national airline—in 1924, Croydon became the new airline’s operating base. It was from Croydon that Britain first developed its European and long-haul routes to India, Africa, the Middle and Far East, Asia, Africa and Australia.

Croydon Airport

On the formation of Imperial Airways—Britain’s first national airline—in 1924, Croydon became the new airline’s operating base. It was from Croydon that Britain first developed its European and long-haul routes to India, Africa, the Middle and Far East, Asia, Africa and Australia.

Another option is to switch to an altogether different scale. You might use a gradual scale for mobile devices and a steeper one for larger screens. To make the most significant impact, switch to a scale where text sizes change dramatically between increments.

(A) The text in this narrow column hasn't changed. (B) This type is larger, staying at the same position but moving to a less gradual scale.

(A) Again, the text in this narrow column stays the same. (B) There's now a much larger difference between sizes of body copy between smallest and largest screens.
TYPOGRAPHY FOR SMALL SCREENS

Readability matters on every device, but small screens bring other challenges, not least the age-old issue of making as much valuable content as possible visible above the fold.

When content contains blocks of running text, consider using first-line indentations rather than paragraph spacing. First-line indentation removes most, if not all, the space above new paragraphs and replaces it...
with indentations. A rule of thumb is to match indents with line heights to form a square.

If the typeface you’re working with includes condensed or semi-condensed versions, consider using them for small screen designs in place of regular styles. 

Left: An indented version of this small screen typographic design.
Right: Using a condensed alternative fits more lines on screen without altering a design’s personality.
READABILITY OF INVERTED COLOUR SCHEMES

Pay particular attention to readability when inverting a colour scheme; for example, setting light-coloured text against a dark background in a product’s night mode. Unless you’re careful, the high contrast of white text on a black background can be tiring, but there are ways to avoid eye strain. Start by decreasing the weight of your type: if your normal copy is set Regular, choose Light for when it’s inverted.

Lightweight letterforms contain more white space, which helps reduce the contrast in inverted colour schemes.

In print, the space between lines of text is called leading because type-setters used strips of lead to separate them. On the web, we know this as line height.

For the best readability, adjust your leading for different measures. The narrower a measure, the less leading you’ll need. As a measure increases,
more leading will be required. For most typefaces, the best leading is somewhere between 120% and 150% of the type size. On inverted colour schemes, you can improve readability enormously by increasing the distance between lines of text.

The text on the left has line spacing of 150%, 1.5, but over on the right — against a black background — I increase leading to 160%, 1.6 to improve readability.
Creative type

People often say that typefaces have personalities, but, I prefer to think of them as having different voices. Every voice has its unique combination of tone, texture, and timbre. That's why one narrator is more appropriate for telling a particular story than another.

A narrator's voice doesn't just complement a story; it brings it to life in ways that no other voice can. Richard Burton's narration of *Jeff Wayne’s Musical Version of The War of the Worlds* is a magical example of this. The characteristics of a voice play a massive part in our perception. However, there's a danger a distinctive voice might distract from the story itself. The same is true of typefaces.

Type designer Eben Sorkin wrote that “The voice of a typeface can influence behavior”, which makes choosing the most appropriate typeface the single most critical factor on which a brand is held together. That is as important for a product as it is in editorial design.

Just as an actor's appearance and idiosyncrasies play a part in the success of their performance, so does the direction they’re given. Think of the typefaces you choose as actors cast for a role, and consider carefully how to direct them.

Headlines

Headlines demonstrate hierarchy and signal the importance of content, but they can do much more than demand attention. Headlines set a scene, so choose a typeface which complements your subject matter. This next story about getaway driver Georgia Durante perfectly suits black italics in this modern sans-serif typeface.

77  https://smashed.by/richardburton
78  https://smashed.by/voicesoftype
I often highlight important words in a headline by changing their colour. Typeface: Prometo Black Italic by Dalton Maag.

You needn’t always position a headline before a story starts. Placing one after a few paragraphs turns them into a prologue.

I emphasise page structure by inverting my main content and pushing a headline down to meet it. Typeface: Prometo Black Italic by Dalton Maag.
The design of this display typeface was based on a grid and it echoes the structure of the columns of text below. For more drama, rotate a headline and position it partly outside the viewport.

Use tight leading and tracking to create overlapping shapes and turn a headline into a distinctive visual component.
Headlines needn’t be large to attract attention. Instead, combine small type with large amounts of white space to contrast against solid blocks of running text. This quantity of white space feels luxurious, but don’t forget to keep that white space in proportion to screen sizes.

You don’t always need images to add visual interest. Deconstructed text can sometimes be even more interesting. Use SVG and media queries to make sure everybody sees a design appropriate for their device.
Standfirsts

In newspapers and magazines, online and in print, a standfirst is the first few lines of an article and is often designed to stand out. Don’t confuse a standfirst with a lead (or lede) paragraph, as a standfirst should stand apart as well as stand out. It should be separate from running text in an article, unlike a lede, which is part of it.

You might also see standfirst paragraphs referred to as “decks,” “kickers” (because they kick readers into the content), “riders,” “summaries” (as they often summarise the content of an article), or — my personal favourite — “the sell” because one of its jobs is selling content to a reader.

While a standfirst paragraph is often larger, bolder, or in capitals, that’s not necessarily how it always looks. There’s no rule book to dictate its size or style, or even its position. That’s something important to bear in mind when you’re designing layouts for many different devices or screen sizes. The design of a standfirst should help it do its job, and should:

• Be designed to catch someone’s eye
• Encourage people to read on
• Give people an idea of what an article contains
• Improve understanding and shareability

INSPIRED STANDFIRST DESIGNS

It’s common to see a standfirst span multiple columns of running text and positioned between an article headline and content. When you’re following that convention, it’s essential a standfirst paragraph be typographically distinct from both a headline and body copy, perhaps by increasing its size or contrast using a heavier or lighter weight.
Laws are meant to be broken. To make sure you catch someone’s eye, place a standfirst above, instead of below, a headline. You might also add extra interest by limiting its width to span fewer columns.

Give a standfirst greater emphasis by adding thick underlines. When the default `text-decoration:underline` isn't strong enough, several experimental CSS properties offer greater control over underline styles.
These include, `text-decoration-color`, `text-decoration-skip`, and `text-decoration-style`.

If you’re looking for something stronger, combine thick underlines with even thicker borders. Keep those border widths in proportion to your underlines and your typeface. For example, a bottom border might be two, three, or four times the width of your underlines.

Borders add structure and the underlines in this standfirst help define the edges of my block of running text.
Borders define space. I outdented this border to prevent a mismatch between the standfirst and running text below.

Add a thick border to the left of a standfirst to define its position and prevent it from drifting into space. Outdenting this standfirst by the combined border and padding width also means its contents align with columns of running text below.

When you position a standfirst close to columns of running text, it needs to look different enough for readers to see the distinction between it and regular paragraphs. As well as increasing the size and weight of standfirst text, add heavy borders to the top and bottom.
Chapter 7

To bring an abstract feel to this layout, add an oversized margin to the left of a floated standfirst. This limits its width, forming a column which pulls the eye towards the content. The image at the top of this column provides a focal point.

How a standfirst paragraph looks is often determined by the overall style of a publication, but it can sometimes be influenced by an article’s subject.

Standfirsts offer a fabulous opportunity to be creative with text. So experiment and, most importantly, have fun.
Its exaggerated leading turns this standfirst into a signature design element.

This offers a fabulous opportunity to be creative, perhaps by deconstructing its content and rotating each line to turn a standfirst into a strong visual element.

Exaggerated leading can be particularly effective when turning a standfirst into a signature design element. The large, centred paragraph in the example above balances the visual weight of a full-height image opposite and its lightweight style makes it look even more distinctive.

Placing a high contrast standfirst over a banner creates a feature which demands attention.
An article about a real-life getaway driver deserves a high-impact design. Placing a high-contrast standfirst over an image which fills the screen creates a feature which demands attention and gets a reader’s adrenaline flowing, ready for what comes next.

Finally, a standfirst should also make readers curious, and this next design becomes hugely more interesting when I turn this centre-aligned standfirst on its side. The loose leading and uppercase style also turn this paragraph into a strong visual element.

There’s no doubt that when written well, a standfirst can connect readers with your content. Its design can also help people understand what’s coming next, and offers us an opportunity to connect them to the brand of a magazine, product, or website.

**Drop caps**

Study manuscripts from centuries ago and you'll find that one of the most distinctive aspects of their design was the use of large capitals which were sometimes even *illuminated* with gold to bring the light of
Directing type

Top left: Adjust the size of drop caps across breakpoints. On larger screens, use any extra space by pulling caps into it. Top right: Why be satisfied with plain drop caps when you can add decorative borders? Middle left: Make a feature of the cap space by replacing caps with symbols or illustrations. A modern twist on illustrated religious texts. Middle right: Use drop caps in more than one paragraph to break up large blocks of text and draw the reader’s eye to a new section.

Bottom left: Use pseudo-elements to insert decorative caps and position them absolutely behind your content. Bottom right: Initial caps sit on, rather than below the baseline. Use the same typeface as your body copy or contrast it with one that’s completely different.
God to a reader. Other times these caps were illustrated with a figure or scene from the text.

Drop and initial caps can be both decorative and useful in that they mark where someone should start reading your text. What’s the difference between them? Initial caps sit on the baseline and drop caps fall below it. Both are now easy to accomplish using CSS but are rarely seen on the web.

You can use initial caps — also known as versal letters — for more than simply indicating the start of a paragraph: they can add personality to headlines too.

Experiment with typefaces that contrast, perhaps by mixing a traditional serif initial cap within a contemporary sans serif headline.

First lines

On magazine and newspaper pages, you’ll often find the first word, three or five words, or even a phrase in the first paragraph emphasised in some way. Sometimes set in bold, other times uppercase.
You can also use pseudo-class selectors to apply that treatment to an entire first line, no matter how long or short that is.

### Quotes

Block quotes and pull-quotes are useful at punctuating solid blocks of running text. They’re also two of the best typographic elements for acting as visual landmarks to catch someone’s eye.

There are no rules about how long or short, big or small quotes need to be, and none for how they look. Quotation marks can look traditional, or move beyond the ordinary and use other symbols which reflect your content and design.

Left: We often contrast quotes with running text. Using the same size, typeface, and weight implies both are equally important. Right: In contrast, use a different type treatment and positioning to place quotes at the centre of someone’s reading experience.
Left: Art director Neville Brody often used varying type sizes to emphasise words, which can be particularly effective for quotes. Right: Hanging, or negatively indented quotes and marks can point the way into your content.

Modern CSS makes it easy to position quotes visually while maintaining an appropriate source order. Indenting quotes into the right edge of your running text will quietly interrupt the reading experience.

Georgia Durante’s life has taken some unexpected turns. She was a photographic model for Kodak, a “Kodak Girl”—who went on to work as a stunt driver. She was the stunt double for Cindy Crawford in a commercial for Pepsi.

Going from modelling to television isn’t unusual, but what Hollywood film crews didn’t know was that, in between, Durante’s driving skills were honed during a career as a getaway driver in New York City at a Mafia-run club called Sundowner. She says she never knew exactly what went on at the after-hours club but, the driver’s side of the mafia’s Underworld. Suddenly, the owner of Sundowner threw Durante a set of keys. They put the man in the back of the car and Durante found herself in the driver’s seat. They dropped the injured man off in front of the hospital and Durante sped away. That incident earned her a reputation. She started off picking up and delivering packages to locations around the city. Then they moved her up to what were called “pickups.” But when a mob war broke out, Durante knew she had to leave that lifestyle. She wanted to take her daughter away from their abusive household, but she could never escape her husband. Then one day, she grabbed her daughter, got in the car and started driving. They ended up in Los Angeles. Durante arrived with only $7 in her wallet. She and her daughter lived out of their car and stole food from convenience stores. She desperately needed a job, but she had to lay low. And the only work experience she had came from modelling and driving cars.

Think carefully about indenting quotes from the left. It’s easy to make it harder for a reader to find the start of a new line.
You can summarise parts of running text or write new copy which makes scanning stories easier. Either way, they can’t be left to chance. It takes collaboration between designers and editorial teams to decide on how many quotes are needed and what they include.

**WHY PULL-QUOTES SHOULD EXIST ON THE WEB**

Jeremy Keith hasn’t seen the value of block or pull-quotes on the web, yet.79

> There you are reading an article when suddenly it’s interrupted by a big piece of text that’s repeating something you just read in the previous paragraph. Or it’s interrupted by a big piece of text that’s spoiling a sentence that you are about to read in subsequent paragraphs. [...] I’m genuinely curious to hear the design justification for pull quotes on the web (particularly on mobile)

Jeremy’s right, of course. There’s nothing worse than being interrupted by anything, quotes included, when all you want to do is read. What is the purpose of pull-quotes, except to annoy people like Jeremy when your design should be engaging them? What’s the fix? Good judgement. And clever art direction.

There’s rarely any value to merely repeating in a quote something that’s said somewhere else in your content. If your aim is to draw someone’s attention, treat quotes as you would headlines and write something meaningful. If you’ve no choice other than to repeat content from elsewhere, hide it from mobile readers altogether.

Making scanning stories easier, and providing places for people to pause while reading them is a worthy goal, but do this intentionally and in moderation. On small screens, reduce the size of quotes so they don’t occupy space unnecessarily, and always consider how people like Jeremy will feel when they see them.

79 https://smashed.by/pullquotes
Choosing the right numbers

Every typeface has its own distinctive set of numerals, so when your content includes plenty of numbers, pay close attention to their design. When you’re aiming for a specific look, you can choose digits and numerals from one typeface, and letters from another. Even numerals from a different typeface classification are perfectly acceptable, so don’t be afraid to use numbers from a sans serif alongside serif letterforms.

Left: These sans serif numerals from Co Text by Dalton Maag contrast beautifully with serif letterforms set in Bressay to create a modern look. Right: Serif numbers from Dedica add a classical feel alongside this Venn sans serif typeface.

Dates and times, or prices, aren’t just data as they can be powerful visual elements and provide the opportunity to enhance someone’s understanding of a story.

When you need a large number, resist the urge to alter its proportions. Avoid stretching the width of numerals and instead increase their overall size to fill the space.
### Directing type

- **Left:** Emphasise important dates by blowing them up to fill a large screen. Exaggerate them further by allowing them to bleed off the edges. **Right:** Numbers needn’t make any less of an impact on small screens.

<table>
<thead>
<tr>
<th>Year</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1924</td>
<td>Croydon became Imperial Airways operating base with flights to India, Africa, the Middle East, Asia, Africa and Australia.</td>
</tr>
<tr>
<td>1959</td>
<td>The last flight took off from Croydon on 3 May 1950 for its last scheduled flight to Australia.</td>
</tr>
</tbody>
</table>

**Croydon Airport**

In December 1919, Croydon Aerodrome was established as a site for the storage of aircraft and crews during the First World War. In January 1914, the first two daily flights took off from the aerodrome as part of the Defence of Middlesex Aerodrome opened. In 1917, the airfield became the home of the R.A.F. South-Eastern Command, with various units stationed there until the end of the war. After the war, the airfield continued to be used for training and civil aviation activities. In 1924, Croydon became the new Imperial Airways operating base with flights to India, Africa, the Middle East, Asia, Africa and Australia. The Croydon Aerodrome Extension Act, which allowed for the extension of the aerodrome, was passed in 1928, allowing for the expansion of the airfield. The airport continued to serve as a major hub for Imperial Airways, with flights to various destinations across the globe. However, with the advent of new aircraft and the expansion of other airports, Croydon began to lose its prominence. In 1950, the last scheduled flight took off from Croydon for its last scheduled flight to Australia, marking the end of an era for the airport.
Never be afraid to treat numbers the same way you would images. Bold numeral designs can be high-impact, especially when you make them signature features of your design.

Left: I transform these numbers into the centrepiece of this design by rotating them and using blend modes to emphasise where they overlap this image. Right: Rotations may not always be appropriate for small screens. Instead, these overlapping numbers form a bridge between the picture above and running text below.

Left: Break numbers apart and turn their components into eye-catching visual elements. Right: Dates needn’t just look like data. Use rotate and scale to transform them into something more.
Directing pictures

Images and how we display them have an enormous impact on how people perceive our designs, whether that be on a commercial or editorial website, or inside a product.

Positioning images

When you imagine a Bootstrap or similar framework-based website, what are the first things which come to mind?

- Centred headlines
- Four columns with icons
- A full-width Jumbotron

In my own, admittedly anecdotal, experience, people have become as oblivious to a Jumbotron as they are to online advertising.

80 https://smashed.by/jumbotron
FILLING A PAGE

The web’s now so packed with Jumbotron-style images that clients have come to expect them. There’s nothing wrong with photographs which fill a screen—in fact they can have a dramatic effect. However, drama needs contrasting with calmness if it is to be effective. That’s because when every website shouts “Look at me!” people eventually stop listening.

Amy Johnson was the first woman to fly solo from the UK to Australia. Her astonishing story deserves telling in a way which is as exciting as her long journey.

When you use images to add drama, make sure their effect is dramatic on all screen sizes. Don’t waste a great picture by squeezing it on a small screen. Merely shrinking an image to fit a phone decreases its visual weight and, therefore, the impact it makes. That’s why you should always consider altering an image’s aspect ratio, and crop it carefully to get the most from it.

Left: Shrunken but otherwise unchanged, this image has lost its intensity. Centre: This aspect ratio is a better fit, but a poor crop reduces the impact this image makes on a small screen. Right: Changing the aspect ratio preserves the influence I want this image to have over my design.
Directing pictures

Left: Shrinking an image to fit a medium-size screen changes the balance of this composition and reduces the impact this photograph has. Right: Better to use `<picture>` and deliver an alternative image with a crop that better suits this screen.

Cropping photographs to make the most of them has become crucial, as it makes a massive difference to someone’s perception of a design across all screen sizes. Oddly, however, cropping rarely gets the attention it deserves.

Full-width photographs aren’t your only option when you want to make a big impression. Images which occupy the entire viewport height can be just as full-on. Page-filling photos are a common sight in print, but I don’t see them in use as frequently on the web. I predict this will change because we now have the grid layout properties and viewport-based units to make implementing them easier.
INCORPORATING GUTTERS AND MARGINS

You should always use a grid to determine image sizes; however, they needn’t be confined by its columns or rows. To lead the eye from one area of a design to another, incorporate gutter heights and widths into the dimensions of your images.

Leaving empty margins between a grid and the viewport is common. This reinforces page structure, but it can also make a design feel predictable. Images which bleed across one or more margins allow the eye to flow off the page and this hints there’s more to see outside the visible area, adding a sense of mystery to a design.

Left: The margin around this large image acts as a boundary which stops the eye moving beyond it. Right: Allowing that same photograph to bleed off the top and right edges of this page implies there’s more to see outside the viewport.

You can achieve a similar result on medium and small screens, albeit in different ways. For a medium-sized screen, I expand the image size horizontally by one gutter width. To emphasise those bleeds on small screens, I keep the left margin free from content.

Amy Johnson died after bailing out of her plane in 1941, and her body was never recovered. It’s important this design reflects her tragic death.
Left: I preserve its aspect ratio while allowing this image to bleed across gutters and margins. Right: Tight cropping helps this photograph maintain its intensity on small screens.

For extra impact, allow an image to spread across the entire page by fading it into the background. When there are no boundaries, the eye is free to roam anywhere and in any direction. This creates an incredible feeling of space, but be careful—with so much freedom, it’s essential you add structure; for example, with solid blocks of justified running text.
Left: The width of this headline emphasises the two columns of running text below. It’s also an ideal counterweight to the large photograph on the right. Right: In this medium-size design, I transform the image caption into a standfirst. The horizontal rules between each line match the width of the headline and running text below.

USING IMAGES AS SIGNPOSTS

Photographs act as signposts, pointing the way to your content. It’s therefore essential you consider an image’s subject before deciding where to place it.

Left: A photograph can suggest where someone should start reading. In this portrait, the eyeline leads straight to a headline. Right: I feel uncomfortable when looking at this design because Amy is looking in the opposite direction and away from my content.
DIRECTING THE EYE

When using powerful images as focal points, you often need to find ways to balance their visual weight so they don’t overwhelm your content. Instead of increasing the size or weight of the running text in my next design, I surround its copy with white space which helps to balance the weight of my image.

Left: Running text combined with a significant amount of (black) white space perfectly balances the image in this composition. Right: Even though this photograph occupies the same space as before, by cropping it tightly around Amy’s face means it dominates this design.

Directing how someone’s eyes move is by no means an exact science. You can’t force people to follow any path set for them, especially with such a diverse range of screen sizes. However, you can influence people through where you position your images.

By bleeding off the top of the page, this tall, tightly cropped image directs the eye down through my content. To form a Z-pattern with the image caption, I align my headline to the right, pulling it closer to the photograph.
We are all drawn to faces, so when your images contain one, pay attention to which way eyes are looking and use this eyeline as part of your composition.

In this design — reminiscent of the Ulm School for Design journal from the 1950s and ’60s — a shallow, horizontal image draws the eye from the page’s left edge, giving the impression the story began out of sight. The placement of this photograph also brings together the three columns of text below. This technique is one commonly found in newspaper design.

While it’s common for newspaper images to span the same number of columns as the story they illustrate, they needn’t necessarily be restricted by those columns. Photographs like this one act like magnets for the eye, so be careful when placing important content above them because people may not notice it’s there.

https://smashed.by/ulm
Small areas offer plenty of ways to get creative with screen-filling images. Instead of scrolling a picture with the page, consider laying it over your content and add a way to dismiss it with a swipe.
ARRANGING MULTIPLE IMAGES

I rarely see designs which contain just one image, so learning how to arrange several pictures on a page will help ensure your compositions stay balanced. Organising same-size photos at regular intervals will create a rhythm which plays throughout your design. It’s this cadence which makes a design flow and will carry people through it.

People quickly tune into a rhythm, and a series of same-size images will encourage a rhythmic eye movement. For my design for this timeline, I use a modular grid which helps create repeating patterns in my content:

```
1915
In December 1915, Reddingham Aerodrome was established - one of a number of small fields around London. But were created for protection against Zeppelin airship raids during the First World War.

1924
Following the Imperial Airways de Havilland DH.44 crash of December 1924, Britain’s first major civil aviation accident, conditions at Croydon were under criticism from the public inquiry.

1928
The terminal building and control tower were completed and the wooden air traffic control and customs building demolished. The new buildings began operations on 17 January and were opened on 2 May 1928.

1930
Amy Johnson took off from Croydon on 5 May 1930 for her momentous flying trip to Australia. She returned from Australia to be greeted by crowds of thousands.

1940
On 15 August, Croydon Airport was attacked in the first major air raid on the British Isles. The armament was destroyed, the civilian airport terminal building was badly damaged by bombs for the first time.
```

Consistency brings individual pieces of content together and makes it clear they’re part of a greater whole. The visual weight of these dates balances the photographs below them.

Images which are significantly larger or smaller interrupt this rhythm, and when someone notices an element is off-beat, they’re unconsciously drawn towards it. Put this knowledge to good use with differently sized images which emphasise the most critical content.
Left: Both the size and alignment of these images add a natural rhythm to this timeline, which is interrupted only when content spans across two grid modules. Right: To maintain a balance of visual weight on small screens, consider using photographs wider than the viewport. Then allow people to scroll horizontally to see what's hidden.

**AVOIDING MONOTONY**

When you maintain the same rhythm across several pages, there’s a definite risk your designs could quickly become monotonous. That doesn’t mean you should abandon consistency altogether, because there’s plenty you can do to add variety, even when sticking to the same grid.

Comic books use differently sized grid modules to pace the action in their stories. Repeating smaller modules speeds up the movement, while larger ones slow it down. People naturally spend longer looking at larger spacial zones and we can use the same technique at particular moments in someone’s journey to slow them down and make them take notice.
From her audacious Australia flight to wartime heroism, Amy Johnson’s story can be told in several parts. Find inspiration in how comic books tell incredible stories like Amy’s.

You could be forgiven for thinking that filling a screen with just one photograph means throwing out the grid, but it needn’t mean that at all. In fact, modular grids can provide dimensions for large images and their captions.

Left: When you want people to slow down and take notice of what you’re saying, use a large image. Right: Step up the pace by arranging smaller photographs in a 1|2|1|2|1|2 pattern.

On the first page, I use a modular grid to give me the size and position of the caption which cuts into the bottom of that full-page photograph. Look closely, and you might notice my caption is a precise size-match for the small pictures in the gallery on the following page.
LAYERING IMAGES

Before CSS Grid, layering design elements was a relatively complex technical challenge, especially when a design needed to fit several screen sizes. Over the past few years, I’ve seen very few designers use layering to any great extent, which is disappointing because overlapping designs feel more alive and spontaneous.

Layering means you can shift the position of images away from horizontal and vertical grid lines without losing the structure which comes from using a grid. Images may not be precisely aligned, but that doesn’t mean the grid can’t play a role in determining their dimensions and how much they overlap.

The dimensions of these four photographs might look arbitrary, but I determine them all using a modular grid, meaning they maintain a relationship with one another and the column of running text nearby. Sadly, by arranging these photographs so neatly, my design feels predictable overall.

The position and size of these four images are based on a modular grid, but somehow, even with their unusual arrangement, this design feels static.
The apparent randomness of this group of photographs contrasts with the formal column of justified running text alongside it.

Overlapping and rotating gives the impression these photographs have fallen onto the page, making my design feel instantly more spontaneous.

There was nothing impulsive about Amy Johnson's flight to Australia, and a seemingly random scattering of photographs won't be appropriate for every part of her story. This needn't mean overlapping images fly out of the window, as there are plenty of different ways to create the illusions of depth and movement.

My next design looks highly structured, primarily because I carefully position every element according to a grid. The headline, image caption, and four columns of running text all occupy modules, and this gives my composition a feeling of formality. To reinforce the impression that this vertical photograph is floating above the page plane, I include two gutter widths into its dimensions making it marginally wider than the columns of copy below.
In the not-too-distant past, blurring photographs was something only done in an image editor such as Adobe Photoshop. Luckily, we can increase the illusion of depth today by applying blur using CSS.

**IRREGULAR SHAPES**

One aspect of print design I always wished was more easily achievable on the web was wrapping text around irregularly shaped images. For years I dreamt up elaborate hacks using presentational HTML in a series of attempts at emulating print layouts. Thankfully, I don’t need to struggle with hacks any more, because CSS clip-path and Shapes make those design conventions more accessible for the web.

One of the complaints levelled at website design over the years has been that it’s boxy and rectangular. This was true, to an extent. The fact that every HTML element draws a box—known as the box model to those of us in the trade—may be partly to blame for this. However, boxy layouts needn’t now be the norm.
You don’t always need fancy shapes because even text wrapped around a simple circle can make a design appear more polished.

While straight-edged images and text columns communicate formality, organically-shaped photos and copy set in fluid columns can feel friendly and informal. Being friendly is essential when you want to make people feel at ease with your design. Organic shapes add movement, which draws people to them. This helps to connect an audience with your story.
You can also use shapes to make tighter connections between your visual and written content. In the previous designs, text in the main story wraps around the form of an aeroplane propeller, first on the right, then the left.

Above: A simple polygon clips this photograph to form an arrow which points to my content.
Below: Clipping paths can turn images into unusual shapes like the coastline of Australia.
Photographs are inherently rectangular, but that needn’t be how you see them. Clipping paths define a visible area, and you can use them to great effect by using clipped photographs to direct the eye, or by turning them into unusual shapes.

You can also sculpt spectacular shapes from solid blocks of text in the style of Alexey Brodovitch’s work in Harper’s Bazaar in the 1930s.

Left: These beautiful numerals are almost too lovely to hide. They’re also perfect for carving into columns. Right: When I remove the fill from these numbers, I’m left with two unusually shaped columns of running text. Below: The right-hand column of running text echoes the outline of Amy Johnson’s portrait opposite.
Sometimes all you need to change the way people think about your design is an unusual treatment of familiar elements. There’s no reason why captions need to stay underneath images as I can also place them above, left, or to the right of a picture.

Left: To make captions more interesting, consider reducing their width. This caption is one-third as wide as the photograph above it. Right: By moving the caption above the image, there’s a danger they may feel disconnected, but the border to the left of this caption binds the pair together. Below: Aligning two elements to the baseline has been challenging to implement in the past, but is now incredibly easy to develop.
Every HTML element is a chance to tell a story, and because a `<figure>` contains images and text, it provides the perfect opportunity to let your imagination fly. Use borders, transforms, and different text styles to turn something as familiar as a `<figcaption>` into a signature piece in your design.

Left: Make a feature of extended captions by applying borders left and right. Right: Deconstruct your content and rotate each line to turn a caption into a strong visual element. Below: By placing this caption in a box, my design immediately feels more like a quality print publication than a website.
When images fill a page, you might think that grids become irrelevant, but they can still help determine the size and position of captions. For this design, I combine two grid modules into one spacial zone and place this reversed caption inside it.

This solid background colour makes a strong statement, but if you’re aiming for something more subtle, experiment by changing its transparency.

Pad each line to create a ragged edge and add a background colour that’s sampled from the photograph to tie the image and caption together visually.
Be bold when you want to make a strong statement. Look for inspiration in other media which are great at telling stories. My final design was inspired by magazine covers, and I split apart highlights from my story and spread them across the photograph. 

While the arrangement of these highlights may look random at first, the size and position of each of them were inspired by the grid.
The pay-off

People who work on the web come from many different backgrounds and not everyone who now designs products or websites went to graphic design school or was taught why to choose one grid type over another. The same is true of how to use symmetry and asymmetry, contrast, and scale. These are not new principles as they have guided art direction and design for decades and matter just as much on the web as they do in the glossy pages of a magazine.

Grids too are fundamental to someone’s understanding of a story, and you can use them for more than merely aligning content to the edges of columns. Typography can be expressive as well as readable. Images can have an enormous impact on how people perceive our designs.

You needn’t have been to art school to learn and apply the principles I taught you. Just like art direction itself, they’re something which everyone—no matter what your background and current area of expertise—can use every day to improve the effectiveness of a product or website’s design.
When they’re carefully chosen, illustrations can portray a brand’s unique personality, a distinctive tone to design, and be highly effective at communicating brand messaging. To get the most from your illustration investment, they should be an integral part of any design and not an afterthought. This is something which the owners of Animal Search understood very well. Animal Search illustrations stuck to a simple, but stylish colour palette which they carried through every aspect of printed marketing materials and website. These illustrations weren’t merely placed randomly, but act as focal points which lead someone’s eye across their pages from one important content area to another.

Their page layouts and typography are equally stylish. However, what matters most is that the design shows a clear vision for the Animal Search brand and that it’s been executed throughout every aspect of their design.

Illustrations by Vic Bell
Equfund

The UK is facing a homelessness crisis (amongst other crises) and—as a Community Benefit Society—Equfund uses investment to provide affordable and emergency housing. I worked with Equfund, refreshing their visual identity, redesigning their website, and developing a consistent brand experience for customers across different channels and touch-points.

To visualise the impact Equfund has on people and communities, illustrator Vic Bell created a series of playful illustrations. These match perfectly their colour palette, including the brand’s distinctive gold accent.

Vic Bell’s playful illustrations suggest the communities which Equfund are helping to preserve. They’re also the perfect compliment to Equfund’s new typography. Equfund used the full illustrations for impact on key pages and smaller versions to bring content and navigation to life.

equfund.com
Developing for Art Direction

If you want to communicate something, you’d better make sure that your design piece is well-dressed and that its teeth are fixed. At the same time, I still believe that if it is only stylistically great and it has nothing to say, it still is not going to make a lasting impression on anybody.

— Stefan Sagmeister
What you’ll need

I am about to teach you some of the latest HTML and CSS, and I’ll assume you’re already familiar with writing well-structured HTML markup and CSS.

Do you need to be an expert? No. However, if you’re new to CSS, I hope you’ll be inspired to learn more. Is this an HTML or CSS reference? Will I explain every aspect of every property? No again. But what I am going to demonstrate will help you implement your art-directed designs.

So you can see these examples displayed across screens of all sizes and types, you’ll need several current web browsers and their developer tools installed, as well as a phone, tablet, or both. I recommend you have the most up-to-date versions of the following browsers installed:

**APPLE SAFARI**

Safari is the default browser on the Mac and the only rendering engine on Apple’s iOS. On the Mac, make sure you go to Preferences in Safari, click on the Advanced tab and check Show Develop menu in menu bar.

**FIREFOX DEVELOPER EDITION**

The Developer Edition of Firefox includes powerful DevTools tools including features specifically for designing and developing with CSS Grid. These tools allow you to visualise grids, display area names, and preview transformations on the grid.
GOOGLE CHROME CANARY

Whereas Safari uses the WebKit rendering engine, Google’s Chrome web browser uses Blink, its fork of WebKit. Chrome has extensions that help us design websites using a browser and test them during development.

MICROSOFT EDGE

While Edge’s logo may be reminiscent of Internet Explorer, Microsoft’s latest browser carries none of its baggage and, unlike other browsers, Edge only runs on the Windows 10 operating system for PCs, smartphones and tablets, and the Xbox console. In December 2018, Microsoft announced a change from their own EdgeHTML rendering engine to Chromium’s Blink, the same engine as Chrome and Opera.

OPERA

Older releases of the Opera browser used Opera’s own Presto rendering engine, but current versions use the same Chromium Blink engine as Chrome.

You won’t need any special software, so feel free to use your favourite text editor.
Developing layouts with CSS Grid

I find it hard to contain my excitement about CSS Grid (Grid). Yes, I know I should get out more, but I think that Grid, plus thoughtful, art-directed content offers us the best chance yet of making websites which are better at communicating with our audiences.

The best part is that unlike previous methods, Grid is by far the easiest layout method to learn. There are 18 properties in Grid and here’s a confession; I won’t teach you all of them. Instead, I’ll explain properties and techniques which are most appropriate for art direction.

Get started using Grid

When starting to implement a design using Grid, you’ll need to:

1. Choose one or more elements to become grid containers.
2. Define how many columns and rows your design needs.
3. Place child elements into position on the grid.

Grid requires a slightly different mental model from Bootstrap and the many other similar layout frameworks developers have used over the past few years. Instead of thinking of a grid as a whole page made up of columns, you can apply Grid to almost every element which can contain descendants. Generally, that is the `<body>`, articles, sections, or divisions, but it could just as easily be a `<blockquote>`, or list, even a `<figure>`.

For block-level elements, use `display: grid;`. When you want inline or flow-level elements to become grid containers, use `display: inline-grid;` instead.
A horizontal axis cuts this layout in two, just as the River Thames divides north and south London.

If you open a page with those properties attached, you’ll see absolutely no change at all, because Grid needs you to define columns and rows before it does anything. There are several ways to define those columns and rows, but one of the simplest is to spell them out using `grid-template-columns` and `grid-template-rows` properties.
Setting columns

Two rival gangs fought for control of London’s underworld in the 1960s: the Krays in the north, the Richardsons south of the river.

Make it obvious how these two gangs faced off against each other, by mirroring the photographs of their members.

To accomplish this design, I need grids with five asymmetrical columns—four for content, and one empty. However, to explain how to make columns, I first make all five 150px wide:

```css
body {
  display: grid;
  grid-template-columns: 150px 150px 150px 150px 150px;
}
```

Typing out those widths more than a few times soon gets repetitive, so when several consecutive values are the same, use the `repeat()` function instead. This simpler rule builds the same five columns:

```css
body {
  display: grid;
  /* Five columns */
  grid-template-columns: repeat(5, 150px);
}
```

Now when you open that page, a browser automatically arranges every direct descendant of `<body>` into five columns. If there are more than five descendants, a browser creates as many rows as it needs, each with five columns of 150px.
Sizing columns

There are occasions when absolute units like pixels, or units relative to type sizes such as \texttt{ch}, \texttt{em}, or \texttt{rem} are the right choice. More often than not, though, column sizes should reflect space available in a container and columns’ relationships with one another. Before you reach for a calculator, there’s a new flexible length unit which makes it redundant. It’s called the \texttt{fr} unit:

```css
body {
  display: grid;
  /* Five columns */
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
}
```

As you might’ve already guessed, that rule creates five columns each as wide as one fraction of available space, making this design fabulously flexible. The result (see next page) isn't quite what I'm looking for, but by changing the values of those \texttt{fr} units, I can adjust the proportions of my columns. The first two columns remain one fraction wide; column 3 and column 4 are four and two fractions respectively:

```css
body {
  /* Five columns: ¼ ¼ ⅛ ⅛ ¼ */
  grid-template-columns: 2fr 2fr 1fr 1fr 2fr;
}
```

Looked at another way:

- column 1 and column 2 both occupy one-quarter (¼).
- column 3 and column 4 fill one-eighth (⅛) each.
- column 5 occupies one-quarter (¼).
fr units are a brilliant addition to CSS, and you can combine them with ch, em, px, rem, or other units to make layouts even more interesting. In this design, column 1 is the perfect candidate for 100 fixed pixels. These 100px are subtracted from the available width before a browser calculates any flexible columns:
Adding gutters

Gutters are the spaces between columns and rows, and they have an enormous impact on layout clarity. Make them too small, you run the risk of making it challenging to understand a layout; too large and you might lose visual connections between elements. The `column-gap` and `row-gap` properties create these gutters between columns and rows. You can use any CSS length unit, and it should be obvious you can't use negative values. I prefer either viewport width or height units because they keep any gaps in proportion to the page:

```css
body {
  /* Five columns: 100px, then ¼ ⅛ ⅛ ¼ of remaining space */
  grid-template-columns: 100px 2fr 1fr 1fr 2fr;
}
```
Developing layouts with CSS Grid

body {
  column-gap: 2vw;
  row-gap: 2vh;
}

Otherwise, use `gap` to combine these two properties into one handy shorthand:

body {
  gap: 2vw 2vh;
  /* When column and row gaps are the same */
  gap: 2vw;
}

**Setting fields**

In Grid, fields are called rows, and by default—when your writing mode is `horizontal-tb`—they run left-to-right and perpendicular to columns. When your writing mode is vertical, rows run vertically.

You may set as few or as many rows as needed and their heights can be intrinsic—defined by the content inside them—or you can give them an explicit height using absolute or relative units. This `<body>` has two rows, both automatically as tall as their content:

body {
  grid-template-rows: auto auto;
}

When several consecutive row height values are the same, use the `repeat()` function again:

body {
  /* Four rows */
  grid-template-rows: repeat(4, 200px);
}
Compound grids

In 1960s London, the Krays were a bigger firm than their adversaries south of the river. I add more visual weight in the north — increasing the headline and image sizes — and emphasise the asymmetry to portray this imbalance.

This asymmetry requires a compound grid: two stacked grids with different numbers, arrangement, and proportions of columns. These grids look different but feel connected because I base them on the same eight-column foundation. In the north, I use four columns for the Krays, while there are six columns down in the south:

Left: The design I’m aiming to develop. Right: With lines from two grids overlaid.
These grids also contain a different number of explicit rows—four at the top, two at the bottom—which I use to arrange my elements.

**Positioning on the grid**

Before I teach you how to arrange elements on a grid, I want to explain the change in mental model Grid requires. All previous layout systems encouraged you to think about numbers of columns. For example, in Bootstrap's twelve-column grid, the column on the far-left is column 1, and on the opposite side of the grid is column 12.
Rather than focus on column numbers, CSS Grid takes a different approach. It wants you to think about the lines around and between the column and row tracks. Line numbers start with 1—from the first edge of the first column or row—and then increase in sequence.

You can position elements by starting and ending on any of these grid lines. For example in this three-column layout, the vertical headline starts on line 1, `<main>` starts on line 2, and the image and `<aside>` start on line 3. Because no element spans multiple columns, a browser assumes the end line, so there's no need to specify it:

```css
body {
/* Three columns: ⅛ ⅜ ½ */
grid-template-columns: 1fr 3fr 4fr; }

h1 {
  grid-column-start: 1; }
main {
  grid-column-start: 2; }
img, aside {
  grid-column-start: 3; }
```
This is an article about disappearing, so make the headline vanish too. Make the suitcase large enough to see stickers from the list of potential destinations.

When you want elements to span multiple columns, you can do one of two things: either add an end line number, or tell a browser how many columns you need elements to span:

```css
div {
  grid-column-start: 1;
  grid-column-end: 5;
  /* Or use this shorthand */
  grid-column: span 4;
}
```

You can take the same approach with rows too, which is especially important when you're developing some of the more interesting layouts that Grid makes possible:

```css
div {
  grid-row-start: 1;
  grid-row-end: 4;
  /* Or use this shorthand */
  grid-row: span 3;
}
```

If you find writing `grid-column-start`, and `grid-column-end` tiresome, use the `grid-column: 1/4;` shorthand. If `grid-column-start`, `grid-column-end`, `grid-row-start`, and `grid-row-end` are too tedious, you can write `grid-area: 1/5/1/4;` instead. For this design, I need three asymmetrical columns:

1. column 1: vertical headline
2. column 2: main content
3. column 3: image and aside
I also require three asymmetrical rows; one for white space above the running text, the other two for arranging my content:

```css
body {
/* Three columns: ⅛ ⅜ ½ */
grid-template-columns: 1fr 3fr 4fr;
}
```

You needn’t restrict elements to any single column or row and—unlike old layout methods—an element can span across more than one column or row. I want to limit this vertical headline to the first column, but spread it across all three rows:

```css
h1 {
/* No need to specify the end-line */
grid-column: 1;
/* Shorthand for grid-row-start, and grid-row-end */
grid-row: 1 / 4;
}
```

I use the same principles to arrange other elements on this grid:

```css
main {
grid-column: 2;
grid-row: 2 / 4;
}

img {
grid-column: 3;
grid-row: 1 / 3;
}

aside {
grid-column: 3;
grid-row: 3;
}
```
To emphasise my content hierarchy, I need to reduce the weight of supporting information: I want to split content in the `<aside>` element into two columns. At present, there’s no way for elements to inherit columns from outside their parent (subgrids). However, it is possible to make any element into a grid container, even when it’s part of another grid.

This `<aside>` is already part of the page grid, but by applying `display: grid;` I also turn it into a grid container with its own set of columns, gaps, and rows if they’re required:

```css
aside {
  display: grid;
  grid-template-columns: 1fr 1fr;
  column-gap: 2vw;
}
```
This `<aside>` element is now a grid container with its own columns and rows. This reduces the visual weight of supporting information.

A good cover can be essential to your new life on the run. Cover identities can take months of planning and practice to pull off.

Pick a name that won’t nip you up, decide on a state of mind, which country you’re from and bring your travel-ology—detailed you will need to remember when you’re crossing a border. Choose a cover you can easily maintain if you’re questioned. You should be able to consume as you cover without any difficulty.

Don’t burden your trip with your own profession, where you live your favourite things should all come without hesitation. You should be able to project yourself on the road, and engineers — if you’re now a business person, you should be able to discuss balance sheets and ROE.

Don’t be afraid to draw on your previous experience, as long as it works with your cover. The most convincing stories are the truth. Telling the truth whenever you can goes a long way towards making your cover believable.

Before you leave for an airport, get away disguised as a bus-ness person. Think about how some entrepreneurs were underdressed with boarding a plane, or going through security. They’d be the type who carry light and start taking off their belt and shoes before they hold up the line. Look around. Other frequent travelers, easy how they behave. Watching them can teach you a lot.

### Track names

Grid line numbers are easy to stipulate, but keeping track of them can sometimes be tricky. When content areas are more predictable, or you use the same lines repeatedly, it can sometimes be simpler to refer to line names rather than numbers. You can give any line any name and add that before its position in `grid-template-columns` or `grid-template-rows`. Make sure you remember to enclose any `name` in square brackets:

```css
aside {
  display: grid;
  grid-template-columns: [box] 1fr [countries] 1fr;
}
```
Minimum and maximum

Even without using media queries, keeping elements flexible is the best way to handle multiple screen sizes. However, there are times when you need to impose limits on how big or small an element should be. In the past, we’ve had to make do with \texttt{min-width}, \texttt{max-width}, \texttt{min-height}, and \texttt{max-height}, but now there’s something better. It’s the \texttt{minmax} value, and for the time being it applies only to Grid.

With \texttt{minmax}, you can stipulate the smallest size an element can shrink to, and the largest size it can become when it grows:

```css
aside {
  display: grid;
  grid-template-columns: [box] 1fr [countries] minmax(11rem, 1fr);
}
```

\texttt{minmax} accepts any values. For example:

1. \texttt{ch}, \texttt{em}, and \texttt{rem}: ideal for blocks of running text
2. \texttt{fr}, \texttt{vh}, \texttt{vw}, \texttt{vmin}, and \texttt{vmax}: for regular grid columns and rows
3. \texttt{px}: perfect for elements containing advertising

You can use \texttt{minmax} to position lines for both columns and rows, and even combine them with the \texttt{repeat()} function for multiple lines:

```css
div {
  grid-template-columns: repeat(8, minmax(1fr, 11rem));
}
```

I lost count of how many conversations I’ve had with print art directors and designers who still think that the web puts limits on the designs they want to make. That might have been true ten years ago, but there are now incredible CSS layout capabilities and fabulous browser support for them.
Aligning grid tracks

CSS Box Alignment works hand-in-hand with Grid. It enables you to position elements at the `start`, `end`, or `centre` of any grid module, or allow them to `stretch` across modules both horizontally and vertically. There are several similarly named box alignment properties which each have a different effect on the position of elements on a grid. There’s no better way to learn the effects these properties have than seeing them in action.
### Properties for grid containers

<table>
<thead>
<tr>
<th>start</th>
<th>center</th>
<th>end</th>
<th>stretch</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="start" /></td>
<td><img src="image2" alt="center" /></td>
<td><img src="image3" alt="end" /></td>
<td><img src="image4" alt="stretch" /></td>
</tr>
</tbody>
</table>

**justify-items** arranges grid items along the inline (row) axis.

Values for **align-items** are the same as **justify-items** but arrange grid items along the block (column) axis.

**place-items** sets both **align-items** and **justify-items** in one handy declaration. The first value sets **align-items**, the second **justify-items**. Just as elsewhere in CSS, one value sets both.
Properties for grid items

<table>
<thead>
<tr>
<th></th>
<th>start</th>
<th>center</th>
<th>end</th>
<th>stretch</th>
</tr>
</thead>
<tbody>
<tr>
<td>justify-self</td>
<td><img src="justify-self-start.png" alt="Image" /></td>
<td><img src="justify-self-center.png" alt="Image" /></td>
<td><img src="justify-self-end.png" alt="Image" /></td>
<td><img src="justify-self-stretch.png" alt="Image" /></td>
</tr>
<tr>
<td>align-self</td>
<td><img src="align-self-start.png" alt="Image" /></td>
<td><img src="align-self-center.png" alt="Image" /></td>
<td><img src="align-self-end.png" alt="Image" /></td>
<td><img src="align-self-stretch.png" alt="Image" /></td>
</tr>
<tr>
<td>place-self</td>
<td><img src="place-self-start.png" alt="Image" /></td>
<td><img src="place-self-center.png" alt="Image" /></td>
<td><img src="place-self-end.png" alt="Image" /></td>
<td><img src="place-self-stretch.png" alt="Image" /></td>
</tr>
</tbody>
</table>

**justify-self** overrides **justify-items**. Arranges content along the inline (row) axis.

**align-self** overrides **align-items**. Arranges content along the block (column) axis.

**place-self** overrides **place-items**. Arranges content along the block (column) axis.
Properties for grids

**justify-content**

Sets the arrangement of a group of grid items within a grid container; for example when the combined height or width of items is less than the container.

*Places the grid along the inline (column) axis.*

<table>
<thead>
<tr>
<th>start</th>
<th>center</th>
<th>end</th>
<th>space-around</th>
<th>space-between</th>
<th>space-evenly</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="start" /></td>
<td><img src="image2" alt="center" /></td>
<td><img src="image3" alt="end" /></td>
<td><img src="image4" alt="space-around" /></td>
<td><img src="image5" alt="space-between" /></td>
<td><img src="image6" alt="space-evenly" /></td>
</tr>
</tbody>
</table>

**align-content**

*Places the grid along the block (row) axis.*

<table>
<thead>
<tr>
<th>start</th>
<th>center</th>
<th>end</th>
<th>space-around</th>
<th>space-between</th>
<th>space-evenly</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="start" /></td>
<td><img src="image2" alt="center" /></td>
<td><img src="image3" alt="end" /></td>
<td><img src="image4" alt="space-around" /></td>
<td><img src="image5" alt="space-between" /></td>
<td><img src="image6" alt="space-evenly" /></td>
</tr>
</tbody>
</table>

- **start** At starting edge of grid module
- **center** In the centre of a module
- **end** At the end edge of a module
- **stretch** Fills height or width (default)
- **space-around** Equal space with half-space at both ends
- **space-between** Equal space with no space at each end
- **space-evenly** Including both ends
Spacial zones

Grid template areas are a way to arrange elements that even a big, dumb mug like me can understand. They also don't get enough attention. In CSS Grid you can define a grid module by giving it a name, then place an element into either a single module or spacial zone, using the `grid-template-areas` property. Does that sound complicated? It isn't.

A grid template area is one of the easiest ways to use Grid. For this design, I have five elements to place:

- `<img>` Large image
- `<header>` Containing a headline and standfirst
- `<main>` Running text
- `<aside>` Supporting information
- `<div>` Containing advertising
Developing layouts with CSS Grid

My HTML looks like this:

```html
<body>
  <img>
  <header>...</header>
  <main>...<main>
  <aside>...</aside>
  <div class="ad">...</div>
</body>
```

I wrote that markup in the order that makes most sense, just as I would when constructing a narrative. It reads like a dream on small screens and even without styles. I give each element a **grid-area** name that I can use to place them on my grid:

```css
img { grid-area: img; }
header { grid-area: header; }
main { grid-area: main; }
aside { grid-area: aside; }
.ad { grid-area: ad; }
```

Your grid area names don't necessarily need to reflect element types. In fact, you can use any values, even single letters like a, b, c, d, or e. I need three columns and rows for this design. The outer two columns each occupy one-quarter of the page width; the running text occupies the remaining half. Their contents define the height of all rows:

```css
body {
  display: grid;
  /* Three columns: ¼ ½ ¼ */
  grid-template-columns: 1fr 2fr 1fr;
  grid-template-rows: auto auto auto;
}
```

Here's where the magic happens. I draw the grid in CSS using the **grid-template-areas** property, where each full stop (.) represents one module:
Now it’s time to position elements on that grid using the \texttt{grid-area} names I created earlier. I place each element’s name into a module on the grid, and if I repeat that value across multiple adjacent modules—either across columns or rows—that element expands across them to create a spacial zone. Leaving a full stop (.) leaves a space empty:

```css
body {
  grid-template-areas:
  " . . . "
  " . . . "
  " . . . "
  };
```

So far, I’ve shown how to use Grid for large screens. However, I usually start with a small screen and then work up, using breakpoints to add or change styles. With Grid, making a layout adapt to various screen sizes is as simple as positioning elements into different grid template areas. There are two ways I can do this: first, by changing the grid itself, second, by positioning elements into different grid template areas on the same grid.

```css
@media screen and (min-width : 48em) {
  body {
    display: grid;
    /* Two columns, the first with minmax */
    grid-template-columns: minmax(300px,1fr) 1fr; }
  };

@media screen and (min-width : 64em) {
  body {
    /* Three columns */
    grid-template-columns: 1fr 2fr 1fr; }
  }
```
Overlapping grid items

In magazine design, elements often overlap to create designs which feel like they’re full of energy. On the web, the only tool we’ve had to accomplish a similar result has been absolute positioning. However, when you position an element absolutely, it is removed from the flow of the document, so other elements have no idea of its dimensions or position. CSS Grid makes it easier than ever to develop flexible designs where elements overlap.
CSS Grid Builder

Grid template areas make developing art-directed layouts so easy that even a flatfoot like me can do it, but if you’re the type that likes tools to do the dirty work, CSS Grid Builder from CoffeeCup Software might be just the thing for you. You might have used WYSIWYG editors before, so you’ll remember how lousy the code they spat out was. Let me stop you there. CSS Grid Builder outputs clean CSS and accessible markup.

Browsers’ developer tools are getting better at inspecting grids, but CSS Grid Builder helps you build them. Obviously. At its core, CSS Grid Builder is a Chromium-based browser wrapped in a user interface, and it runs on macOS and Windows. That means if the browser can render it, the UI tools can write it, with one or two notable exceptions including CSS Shapes.

In fact, CSS Grid Builder builds more than grids, and you can use it to create styles for backgrounds—including gradients, which is very handy—borders, and typography. It even handles Flexbox and multi-column layouts. If you’re one of those people who’s worried about the shrinking percentage of people using incapable browsers, CSS Grid Builder also offers settings for fallbacks. Then copy and paste the CSS styles to somewhere else in your project or export the whole kit and caboodle.
For larger screens, I develop this energy-filled grid with ten columns and two automatically sized rows. There are several interesting features of this grid:

1. Column 2 uses \textit{em} to bind width to its text size
2. Column 3 uses \textit{rem} to calculate its width from the root element
3. I named two columns: \texttt{[content]} and \texttt{[case]}
4. I set the vertical alignment to \texttt{start} instead of the default \texttt{stretch}

```css
@media screen and (min-width : 64em) {
  body {
    display: grid;
    grid-template-columns: 1fr 30px 3rem
      \[content\] 1fr 1fr 1fr 1fr
      \[case\] 1fr 1fr 1fr;
    grid-template-rows: auto auto;
    align-items: start; }
}
```

The HTML for this design is surprisingly compact. It looks like this:
I position those eight elements on my grid using a combination of line names and numbers, just as I did in previous examples:

```css
/* The first row */
small {
  grid-column: 2;
  grid-row: 1;
}

h1 {
  grid-column-start: content;
  grid-column-end: 9;
  grid-row: 1;
}

/* The second row */
p {
  grid-column: content / 7;
  grid-row: 2;
}

[src*="stamp-1"],[src*="stamp-2"],[src*="stamp-3"],[src*="stamp-4"] {
  grid-row: 2;
}

[src*="stamp-1"] { grid-column: 4; }
[src*="stamp-2"] { grid-column: 7; }
[src*="stamp-3"] { grid-column: 8; }
[src*="stamp-4"] { grid-column: 9; }

/* Elements spanning both rows */
picture {
  grid-column-start: case;
  grid-column-end: -1;
  grid-row: 1 / span 2;
}
I do two things to emphasise the overlapping quality of this design; first, the headline is slightly transparent and uses two text shadows to improve legibility when it overlaps the suitcase image:

```css
h1 {
  color: rgba(255,255,255,.85);
  text-shadow: 0 0 10px #ba0e37, 0 0 40px #ba0e37;
}
```

Second, by using `z-index`, I override the position in the stacking order of several elements:

```css
picture {
  /* One step beneath other elements in the stacking order */
  z-index: -1;
}

[src*="stamp-1"] {
  /* Image below text in the stacking order */
  z-index: -1;
}

[src*="stamp-3"] {
  /* Image is underneath everything, at the bottom of the stacking order */
  z-index: -3;
}
```

So many designs on the web include a full-width banner image—you only need to look for Bootstrap layouts or Squarespace templates to see that—so this design makes a refreshing change. It could easily be appropriate as a replacement to those generic banners, or might be a great way to set the scene for a more in-depth article.

Getting away might involve leading the authorities in the wrong directions, so angle every element off the straight and narrow.

Most articles which have been designed with readability in mind include solid blocks of running text. To provide relief from those structures and inject even more energy, I want to rotate every element in this design
away from the horizontal or vertical. Transforms are the ideal tools for this, although you may need to experiment with the number of degrees, translation amounts, and even transform origins to get the precise positioning you're looking for:

```css
@media screen and (min-width : 64em) {
  small {
    transform: rotate(160deg) translate(30px, 0px);
  }

  h1 {
    transform: rotate(20deg) translate(60px, 100px);
  }

  p {
    transform: rotate(-10deg) translateY(80px);
  }

  [src*="stamp-1"] {
    transform: rotate(30deg) translate(-20px, 75px);
  }

  ...
}
```

Whether you make websites for businesses who want to sell more, charities who need to raise more money through donations to good causes, or news outlets that want to tell stories more effectively, Grid plus thoughtful art direction makes all that possible. ■
Developing components with Flexbox

In atomic web design, navigation links, images and captions, and search inputs and buttons are called *molecules* and *organisms*. While Grid is ideal for implementing art-directed layouts, Flexible Box Layout (Flexbox) is often better suited to developing these molecules and organisms.

Unlike Grid, there's no explicit column and row terminology in Flexbox. Instead, there's a visual model with horizontal and vertical axes. When you make an element flex, its descendants flow along one main axis, another axis that's perpendicular, and sometimes both. These two axes make Flexbox the ideal tool for flexible components in an art-directed layout.

In normal flow, elements display one after another, depending on a document's writing mode. If you're new to these concepts, Rachel Andrew wrote an excellent primer.⁸²

Think of flex as an imaginary line drawn across a box. This line is the main axis in Flexbox, and you spread flex items along it. Just like justifying text left, right, or centre, you can justify items along the main axis. When you change the direction of that line, those items appear to run in the opposite direction, even if you don't change the order in your markup. You can switch from horizontal to vertical just by choosing a different `flex-direction`, and even change the `order` in which individual items appear.

⁸² https://smashed.by/flexbox-use-cases
Flexing a container

Using Flexbox, all I need to make this timeline of Mini Cooper models (below) is an equally compact ordered list:

```html
<ol id="timeline">
  <li><h3>1969</h3></li>
  <li><h3>1976</h3></li>
  <li><h3>1984</h3></li>
  <li>…</li>
</ol>
```

Make the difference between these two content areas obvious by using a different grid for each. Draw attention to the most important information in the timeline—in this case the dates—by making them red to match the Mini.

Vanish for good

A good cover can be essential to your new life on the run.

Pick a name that won’t easily trip you up, declare a date of birth, which country you hail from, and why your parents are traveling—all details you remember whenever you’re crossing borders. Choose a cover: you can really mean it if you’ve researched your own career without difficulty. Detail your own life with your own profession, where else you, your favorite niche should be stamped on your history. You should be able to discuss your own career. The better you know for an answer, the easier your professional will be.

1969
1. Cooper Mk I (P138)
2. Cooper Mk II
3. MkIII

1976
1. Mini Moke (introduced in 1971, offered until 1984, microcar / citycar with foldable side windows)
2. Mini Moke / 1000

1984
1. Mini Moke / 1270 (introduced in 1984, offered until 1995, microcar / citycar with soft top folding roof body style)

1989
1. Cooper Mk III / 10th Anniversary (introduced in 1989, microcar / citycar with foldable side windows)

1990
1. Cooper Mk IV / VDP/4
2. Cooper Mk V
3. Mini V

1991
1. Cabriolet / 1000 (introduced in 1991, microcar / citycar with convertible body style)

1992
1. Mini Moke / 1270

1993
1. Cabriolet / 1275 (introduced in 1993, microcar / citycar with convertible body style)

The visual order of this ordered list matches the source order, earliest date first.
There's one list item per year containing a headline and another, nested ordered list, or short paragraph:

```html
<ol id="timeline">
  <li>
    <h3>1969</h3>
    <ol>
      <li>Clubman Estate</li>
      <li>...</li>
    </ol>
  </li>
  <li>...</li>
</ol>
```

Each list item in the timeline naturally displays vertically, but I can change that by turning it into a flex container:

```
#timeline {
  display: flex;
}
```

While `flex` turns that list into a flex container, it doesn't remove any block properties, so the list still fills all available space inside its parent element. If I prefer an element not fill up available space, or I'd like to turn an inline element into a flex container, I can use `inline-flex` instead:

```
span {
  display: inline-flex;
}
```

With `display: flex;` applied, the timeline's list items are arranged along the main axis automatically. As the authors of Flexbox included some smart default behaviours, that could very well be all I need.
Flex direction

You’ll often need to change the visual direction of flex items from horizontal to vertical, mainly when people use screens that have portrait orientations. Flexbox makes this straightforward. When you don’t specify a `flex-direction`, the default direction is a row. Unless you’ve set the `dir` attribute to `rtl` (for right-to-left languages), a `flex` starts on the left and ends on the right.

In the layout I’m developing, the date order appears oldest first, but I can change that easily by reversing the `flex-direction`, altering the visual order without modifying my markup:

It’s important you remember accessibility when changing `flex-direction` as `row-reverse` disconnects the order of your content in HTML from how it’s displayed visually, and this could easily confuse some people.

```
#timeline {
  display: flex;
  flex-direction: row-reverse;
}
```
There are plenty of occasions when I'd want to specify `flex-direction` as a column. Each of the timeline's list items contains a headline date and other content. If headlines come before the content, any differences in length won't matter. If I want my content to appear after, those headlines won't line up.

By making each list item a flex container, changing the `flex-direction`, and then applying the `auto` value to the top of my headlines, those dates line up neatly again.
Vanish for good

A good cover can be essential to your new life on the run.

Creative captions

Figure elements include one or more captions and images:

By aligning the dates to the bottom of each list item, I improve how the timeline looks and make scanning them easier.

Captions commonly appear under the images they describe, following the usual order in markup.

There's no reason captions need to stay underneath images. I can place them above, or to the left or right of an image too. In fact, they needn't be next to an image at all. I can put them anywhere on a page. Sometimes a caption above an image looks more exciting, and I use Flexbox and then \texttt{flex-direction} to move them:
Developing components with Flexbox

Left: A standard caption underneath an image. Right: Even when captions stay where expected, limiting their maximum width can make them look more interesting.

```
figure {
  display: flex;
  flex-direction: column-reverse;
}
```

Limiting a caption’s maximum width and placing it above an image immediately gives a design an editorial feel.
When I place captions on either the left or right of an image, I immediately give designs the look of a magazine. To position captions and images horizontally, I change the `flex-direction` again. For a caption on the right:

```css
figure {
  display: flex;
  flex-direction: row; }
```

And on the left:

```css
figure {
  display: flex;
  flex-direction: row-reverse; }
```

Use **flex-direction** to place captions on either side of an image. Remember to keep both in proportion and adjust the size of caption text to match the measure.
Wrapping flex items

In old-fashioned float-based layouts, when the width of several floated elements combined is broader than their parent element, those which won’t fit drop onto a new line. Flexbox behaves differently, and a flex container automatically increases in width to allow for the widths of every flex item. The authors of Flexbox made a smart choice when they made `flex-wrap: nowrap;` the default behaviour.

Change the `flex-wrap` value to `wrap` and browsers respect any widths applied to flex items and allow them to flow onto new lines.

When I use `flex-wrap: nowrap;` it overrides any specific width given to flex items.

There’s usually no need to set `flex-wrap` to `nowrap` as this is the default.
So far, the items in this timeline are in chronological order: 1969–1989. Whether I use \texttt{wrap} or \texttt{nowrap} values, they appear in the same order as their markup. I can reverse the order those items are displayed in:

```css
#timeline {
  display: flex;
  flex-wrap: wrap-reverse;
}
```

To save a few characters, combine \texttt{flex-direction} and \texttt{flex-wrap} into the \texttt{flex-flow} shorthand property:

```css
#timeline {
  display: flex;
  flex-flow: row nowrap;
}
```

These properties make art directing layouts which would've been difficult or even impossible much more straightforward. However, there's much, much more Flexbox can do.

Reversing the chronological order while wrapping items in this timeline.
Sizing flex items

Before Flexbox, my process developing fluid grids started with a calculator:

\[ \text{target} \div \text{context} = \text{result} \]

\[ 300 \div 960 = 0.3125 \]

\[ \text{width: 31.25\%} \]

Margins between elements affected calculations too, making developing responsive layouts more of a mathematical than creative process. Flexbox changed that, and I haven’t used a calculator since.

When a flex container is broader than the combined widths of items inside it, a browser leaves any remaining space empty. More often than not, I want those items to grow and fill all available space. Flexbox contains properties that provide options for controlling how elements grow or shrink to match the width of their container.

Browsers resize images to fit into a flex container—but remember performance. No one likes to wait while large images download.
**flex-grow** determines how elements expand to fill space, in proportion to their siblings. Using **flex-grow** is easy; understanding how browsers calculate widths can be tricky.

To explain: I have three elements, each 300px wide. The default value of **flex-wrap** is **nowrap**, so when a flex container is narrower than 900px (all their widths combined), browsers ignore those widths and resize elements to fit.

Above those combined widths, browsers leave white space at the end of the flow.

*Leaving empty space at the end of a line is normal block behaviour.*

Giving those elements all a **flex-grow** factor of 1 makes them expand to fill all the available space in a container, no matter how wide it is:

```javascript
li {
    flex-grow: 1;
}
```

*These images all expand by the same amount to fill any available space.*
Those items expand beyond their specified 300px width until all the space in their flex container is filled; their widths increase by equal amounts. But what if I want the available space in that container distributed differently? To do this, I give one of those items a higher `flex-grow` factor:

```
li:nth-of-type(2) {
  flex-grow: 2;
}
```

Now, the second element expands by twice the amount of available space as its siblings.

While `flex-grow` controls the ratio by which elements expand to occupy available space in a container, `flex-shrink` does the opposite. This property specifies the ratio that elements reduce in size when a container is narrower than their combined widths. When I set the value of `flex-shrink` to 1, every element shrinks by the same amount:

```
li {
  flex-grow: 1;
  flex-shrink: 1;
}
```

To shave a few bytes from my style sheet, I can combine those two properties into the `flex` shorthand:
Both `flex-grow` and `flex-shrink` allow the widths of elements to be completely fluid, with no restriction on how wide or narrow they can be. There are occasions when a flex item should start at a certain size before it grows or shrinks. Enter the last flex property—`flex-basis`—and perhaps the most difficult to understand.

In Flexbox, `flex-basis` provides a starting width for an element before it flexes. When neither `flex-grow` nor `flex-shrink` are applied, with a `flex-basis` of 300px, this couple of elements are equal in size:

```css
li {
  flex-basis: 300px;
}
```

Both `flex` items are the same size before they either contract or expand.

In horizontal layouts, `flex-basis` acts the same way as width, leaving white space at the end of the flow:

```css
li:nth-of-type(1) {
  flex-grow: 1;
  flex-shrink: 1;
}
```
Ordering flexible boxes

Designers have wanted the ability to rearrange the visual order of elements without modifying underlying HTML source order for years. Developers came up with complicated hacks like extreme negative margins, but now Flexbox does that for them, no hacking required.

The flex-direction property regulates the overall flow direction, but order enables precise control over the position of any flex item in that flow. There are no classes or IDs on any of these divisions, but each has an implied number, which comes from the order they appear in the document source. The first division has an ordinal value of 1, the second 2, and so on.

There are plenty of occasions I may want to reorder elements, especially when developing art-directed layouts for different screen sizes. I can use the order property and—pay attention—this doesn’t include a flex-prefix. I don’t need classes or IDs in my markup either, because pseudo-class selectors are for occasions exactly like this:
Again, it’s important you remember accessibility when changing the order as this disconnects your content order in HTML from how it’s displayed visually, and this could confuse some people.

These destinations follow the same order as the document source: China, Russia, and finally Spain.

Changing the visual order this way doesn’t affect the DOM, so screen readers still read the source order. Remember accessibility and don’t do anything that might hinder it.
The initial order value for all items in an ordinal group is zero (0), so I don’t need to set that on every item. If I only change one item to -1, it appears before all the others. When there’s only a small number of items, I can afford to give each one its order value, so I can arrange them in an order that’s very different from the source:

```css
div:nth-of-type(1) {
  order: 3;
}

div:nth-of-type(2) {
  order: 1;
}

div:nth-of-type(3) {
  order: 2;
}
```

Flex axes

Unlike floats, which are one-dimensional, flex containers have two axes, perpendicular to each other. They’re called the main axis and cross axis.
When the main axis is a row, the cross axis is a column and crosses it at 90 degrees.

If the main axis is a column—for example, when I set the `flex-direction` to `column` or `column-reverse`—the cross-axis is a row. These axes make Flexbox remarkably useful for developing designs where elements flow in several directions, and for justifying and aligning items in different ways.

Just like when I justify text, I can justify flex items along the main axis using the `justify-content` property and values: `flex-start`, `center`, and `flex-end`. Three additional values control the distribution of any remaining space outside those flex items. They are `space-between`, `space-around`, and `space-evenly`.

Whereas `justify-content` arranges flex items along the main axis, the `align-items` property arranges them along the cross axis. I can align flex items to either the `flex-start`, `center`, or `flex-end` of the cross axis. If I specify no alignment at all, a value of `stretch` is the default, making it easy to develop columns whose backgrounds fill the full height of a flex container.
Wheel Man: Flexbox action

It’s time to bring CSS Grid and Flexible Box Layout together: Grid for the page, Flexbox for its components. Here’s the design I’m developing.

Left: I rotated this box of running text just enough to make it distinctive, but without making it difficult to read. Right: I rearranged this layout to make it more appropriate for a medium-size screen.

The running text is separate from, but still related to, the suitcases and information about the three countries, so connect them visually by making them overlap.

For medium-size screens, I’ve chosen a six-column grid and eight columns for larger screens. For the page layout, I need just two elements:

```
<main>…</main>
<aside>…</aside>
```

I wait until there’s enough space before applying Grid. This layout is not complicated, so using named grid lines makes the most sense.
@media screen and (min-width : 64em) {
  body {
    display: grid;
    grid-template-columns:
      [main-start] 1fr 1fr 1fr
      aside-start] 1fr 1fr 1fr 1fr
      [aside-end] 1fr; }
}

Now I bind my elements to those columns using my named grid lines:

@media screen and (min-width : 64em) {
  main {
    grid-column-start: main-start;
    grid-column-end: aside-start; }

  aside {
    grid-column-start: aside-start;
    grid-column-end: aside-end; }
}

Left: An eight-column grid for larger screens. Right: A six-column grid for medium-size screens where the \(<aside>\) element flows underneath \(<main>\).
Any Grid area can itself become a grid, nesting one inside another. I take advantage of this to position the `<main>` headline and a division full of running text:

```html
<main>
  <h1>…</h1>
  <div>…</div>
</main>

@media screen and (min-width: 48em) {
  main {
    display: grid;
    grid-template-columns: [headline-start] 1fr 1fr
      [division-start] 1fr 1fr 1fr [division-end] 1fr;
  }

  h1 {
    grid-column-start: headline-start;
    grid-column-end: division-start;
  }

  div {
    grid-column-start: division-start;
    grid-column-end: division-end;
  }
}
```

Left: Headline and division full of running text inside the `<main>` element. Right: When there's space enough to display them, I turn my `<aside>` into a flex container to create columns for the three suitcases.
It’s important to remember that elements can be both grid items and flex containers. I use Flexbox to arrange those suitcases and the content which goes with them. Inside my `<aside>` are three figures, images, and captions:

```html
<figure>
  <img src="">
  <figcaption>
    <h3>…</h3>
    <p>…</p>
  </figcaption>
</figure>
```

When there’s space enough to display the three suitcases side by side, I turn that `<aside>` into a flex container:

```css
@media screen and (min-width : 48em) {
  aside {
    display: flex; }
}
```

Each `<figure>` element occupies the same proportion of available space in that `<aside>`, with a margin to the right of all but the final `<figure>`:

```css
@media screen and (min-width : 48em) {
  figure {
    flex: 1 1 auto; }

  figure:not(:last-of-type) {
    margin-right: 2vw; }
}
```

This is a `:not()` CSS pseudo-class selector, otherwise known as a negation pseudo-class. It matches any `<figure>` which is not the last of its type within its container.
On medium-size screens, those figure captions flow underneath their images naturally, without me doing anything at all. However, I want them to sit alongside the images to make my larger screen design more interesting.

It’s also important to remember that elements can be both flex items and flex containers. I apply `display: flex;` to each `<figure>` to switch the orientation of those images and captions:

```css
@media screen and (min-width : 64em) {
  figure {
    display: flex; }

  img {
    flex: 3; }

  figcaption {
    flex: 2; }
}
```

By default, images that are also flex items retain their implicit height, even when their width changes. There aren’t many occasions I can think of where I want this to happen. To prevent images from being distorted, apply the `align-self: center;` property to images.

Ultimately, to make my design more interesting, I use a transform to rotate the entire `<main>` element by 10 degrees anti-clockwise:

```css
@media screen and (min-width : 64em) {
  main {
    transform: rotate(-10deg); }
}
```

One final touch. I want each caption to align to a different place: the caption in the first `<figure>` to the end of the vertical axis, the second in the
centre, and the final caption to the start. I use the Flexbox `align-self` property with three different values to accomplish this:

```css
@media screen and (min-width : 64em) {
  figure:nth-of-type(1) figcaption {
    align-self: flex-end;
  }

  figure:nth-of-type(2) figcaption {
    align-self: center;
  }

  figure:nth-of-type(3) figcaption {
    align-self: flex-start;
  }
}
```

This property allows the alignment of any element to override any values set previously using `align-items`.

For years, I was frustrated that I needed to use extra presentational and semantically unnecessary markup, and convoluted CSS to develop the type of complex layouts I needed for my art-directed designs. I’m so glad those days are gone and that Flexbox packs properties which make striking designs possible at every screen size.
Developing typography

It’s been possible to create columns of text without resorting to presentational markup for years. In fact, I’ve written about the possibilities of CSS Multi-column Layout in every book since 2005. Yet I still see very few websites taking advantage of it, even today.

There are plenty of reasons to use Multi-column Layout, not least the fact that using columns reduces the measure, which makes reading long lines of copy more comfortable.

Multi-column layout

Consider this design when it's viewed on a medium-sized screen. In portrait orientation, the running text has an optimal measure of 75 characters per line. However, turning the screen to landscape pushes the measure beyond what’s comfortable at this text size.

Readability on small screens is one of the most important design considerations today.
Art direction at Airbnb

To give people an experience of a brand that’s consistent, engaging and delightful to use across multiple platforms is especially challenging when a company needs to balance inspiration with the utility of doing something as simple as booking a place to stay. But it’s something art directors and designers at Airbnb do incredibly well. When they were designing their Plus service, Airbnb designers thought carefully about which parts of the design should use elements their customers were already familiar with, and what should be novel and new. What’s especially impressive about Airbnb Plus is how, even within a template-driven, dynamic system, designers can achieve pages which appear to have been individually art-directed.
For their revamped amenities interface, Airbnb considered whether to use icons or photography for items as diverse as internet routers and irons. The carefully matched photographs feel more authentic and reassuring than iconography or illustrations, something that’s important when someone is finding a place to stay on a service like Airbnb.
Setting type in columns is one of the most effective ways to improve readability. So why do we see it so rarely on the web?

Column widths and counts

To improve someone’s experience of this running text, I can introduce columns: first by using the `column-width` property. When I want columns to be specific widths, I set the measure using any CSS absolute length unit: millimetres, centimetres, inches, picas, pixels, and points; or relative units: `ch`, `em`, `ex`, `rem`, `vh` and `vw`, `vmax` and `vmin`. A browser automatically calculates how many columns of that width it can fit into the available space:

```css
div {
  column-width: 32rem;
}
```

Browsers start with one column on small screens and add more columns as the space allows.
Using text-based units means that `column-width` binds the number of columns to the size of my text, which is ideal for readability. However, I can think of plenty of situations where I might want to specify a precise number of columns instead. The `column-count` property is what I need here to make two columns:

```css
div {
  column-count: 2;
}
```

The widths of these two columns will change according to space available, but the number of columns will stay the same. I can vary the number of columns on different screen widths by using media queries:

```css
@media screen and (min-width : 48em) {
div {
  column-count: 2;
}
}
```

A browser will adjust the width of these columns automatically, but pay attention to the leading and measure of your type to ensure it stays readable.
GUTTERS

Gutters are vertical spaces between columns, and their width can have an enormous impact on the clarity of your layouts. Too small and it might make text difficult to read; too large and you might lose the connection between columns. I could specify this `column-gap` using px, em, or rem, but I prefer to use viewport units as they keep gaps in proportion to the page:

@media screen and (min-width : 48em) {
  div {
    column-gap: 2vw;
  }
}
RULES

Horizontal rules are important enough to warrant an `<hr>` element. Vertical rules can be equally important. Although there isn’t a `<vr>` element, there is a `column-rule` property in Multi-column Layout. Here, I specify the width of a rule using pixels. I often increase rule widths across media query breakpoints:

```css
div {
    column-rule-width: 1px;
}

@media screen and (min-width : 48em) {
    div {
        column-rule-width: 2px;
    }
}

@media screen and (min-width : 64em) {
    div {
        column-rule-width: 4px;
    }
}
```

Of course, I can specify the colour of rules too:

```css
div {
    column-rule-color: #f5f5f5;
}
```

Finally, I define a style for those rules. `dashed`, `dotted`, and `solid` are staples styles for rule design. You’ll be excited to find that `groove`, `ridge`, `inset`, and `outset` are all valid rule styles. I know I was.

```css
div {
    column-rule-style: solid;
}
```

SPANNING COLUMNS

When people read text in columns, their eyes move down one column to the bottom, then back up to start the next. Whereas in a magazine or
newspaper, people are familiar with reading text in columns, they’re less accustomed to doing that on the web.

Unlike a printed page, on the web you rarely know the height of someone’s window on your content. Tall columns can quickly grow beyond the viewport, which makes reading them more difficult, so you need to think carefully about the structure of your content before setting long passages of copy in columns.

Flowlines are useful for ensuring that blocks of running text are more likely to fit within one screen. You can use horizontal rules, images, or text elements such as significant headlines to define these sections and help people understand the purpose of each one. Apply the column-span property to some carefully chosen elements:

```css
h1, h2, figure {
  column-span: all;
}
```
This property makes an element span any number of columns, and while `column-span` allows several values (`all`, `inherit`, `initial`, and `unset`), only `all` is of any practical use.

For years, I was disappointed that more developers didn't share my appreciation of Multi-column Layout. However, with support in browsers now universal, and Grid making people think differently about layout, I hope more developers will rediscover them.

**Arranging type with writing modes**

A browser lays out elements by starting in the top-left of the viewport and arranging them horizontally until it reaches the top-right. Then it starts a new line and continues down the page. This left-to-right, top-to-bottom arrangement is called its block flow direction. If you were to specify this default writing mode, it would be:

```
div {
  writing-mode: horizontal-tb;
}
```

In this mode, elements flow horizontally left-to-right, and vertically top-to-bottom. There are two more modes:

- **vertical-rl**: Content flows vertically top-to-bottom, horizontally right-to-left. Browsers place new vertical lines on the left.

- **vertical-lr**: Content flows vertically top-to-bottom, horizontally left-to-right. Browsers place new vertical lines on the right.
**Wheel Man: Vertical headlines**

Vertical text elements are a common sight in magazines, but I rarely see them on the web, except in Chinese, Japanese, Korean, and other scripts, which use vertical writing modes. Nonetheless, vertical text can make a big impression online, even when used in small ways.

Use layout to help tell the story of what comes between Ronnie and Reggie. It’s OK to make readers feel uncomfortable.

I want to return to the story of Ronnie and Reggie Kray. This time, a bar of vertical text separates them in a way that sees design imitating life.
Developing typography

To develop this vertical headline, I need do nothing more than switch its default writing mode to vertical (right–left):

```css
h1 {
  writing-mode: vertical-rl;
}
```

My headline fits in the centre of this page and the reading direction starts at the top. To make it read bottom-up, I can spin it around by 180 degrees and justify it right, so it starts at the top of the column:

```css
h1 {
  transform: rotate(180deg);
  text-align: right;
}
```

CSS TEXT-ORIENTATION

Using `transform:rotate();` and `text-align:right;` to change an element's reading direction feels like a hack, and it is. You won't need that hack for long, though, because a new set of writing mode values are coming.

Left: `sideways-rl`. Content flows vertically top-to-bottom and text is written sideways, towards the right. Right: `sideways-lr`. Content flows vertically top-to-bottom and text is written sideways, towards the left. (Shown in Firefox.)
**Vertical text elements**

Vertical headlines might be too much of a stretch for some, but there are plenty of other ways to add character to a design using vertical text. Photo credits are just one, and you won’t need any special markup to make them.

This Krays design includes one large image of the twins. It would lose its impact if squeezed unaltered onto smaller screens, so I use `<picture>` to deliver two images with different crops:

```html
<picture>
  <source srcset="img-large.jpg" media="(min-width: 48em)">
  <img src="img-small.jpg" alt="">
</picture>
```

It’s important that people realise which photograph a caption refers to, and the `<figure>` and `<figcaption>` elements bind them together both semantically and visually. My `<figure>` markup looks like this:

```html
<figure>
  <picture>...</picture>
  <figcaption>...</figcaption>
</figure>
```

Vertical text is an excellent way to add character while making the most of the space you have available.
When horizontal space allows, I turn that `<figure>` into a positioning context for any absolutely positioned descendants, then use text-based padding to clear an area on the right. This creates space which fits my `<figcaption>` photo credit perfectly:

```css
@media (min-width: 48em) {
  figure {
    position: relative;
    padding-right: 2em;
  }
}
```

Using CSS absolute positioning, changing its writing mode, and rotating the `<figcaption>` by 180 degrees moves it into that space:

```css
@media (min-width: 48em) {
  figcaption {
    position: absolute;
    bottom: 0;
    right: 0;
    text-align: right;
    transform: rotate(180deg);
    writing-mode: vertical-lr;
  }
}
```

**VERTICAL NAVIGATION**

This article about the Krays is part of a series about London's gangland in the 1960s, so linking to a list of related articles would help people navigate around the Wheel Man website. To develop that, all I need is a single link:

```html
<a href="/" title="Go to series">London in 1967</a>
```

I use the same technique as before, positioning this link off the edge of the page to hide its left border:
Occasionally I want the glyphs in vertical text elements to appear upright instead of sideways. In this past, this meant convoluted markup and complicated CSS. Fortunately, the `text-orientation` property has made this much more straightforward. All I need for this navigation is to add `text-orientation: upright;` to the previous style.
VERTICAL TEXT IN A BRANDED HEADLINE

With their connections to celebrities and politicians, the Krays built a well-known brand as well as a criminal empire. I want the headline in this article about Reggie to be branded too, and developing it using Grid and writing mode is easy. My markup looks like this:

```html
<body>
  <header>…</header>
  <main>
    <h1>…</h1>
    <article>…</article>
  </main>
</body>
```

My design includes this K branded headline. Unlike an image, this element has no impact on performance.
My design depends on nesting three grids using `<body>`, `<main>`, and the deconstructed `<h1>`. When space becomes available, I first create two equal-width columns on the `<body>`:

```css
@media (min-width: 64em) {
  body {
    display: grid;
    grid-template-columns: 1fr 1fr;
  }
}
```

To develop the asymmetric proportions of `<main>`, I add two more columns; one is half the width of the other (1fr) and acts as a wide margin on the left of my running text (2fr):

```css
main {
  display: grid;
  grid-template-columns: 1fr 2fr;
}
```

The `<article>` inside `<main>` starts at column 2, whereas the headline above it starts at column 1 and spans the entire width of the grid container:

```css
article {
  grid-column: 2;
}

h1 {
  grid-column: 1 / -1;
}
```

This ability to apply a grid to almost every element is one of CSS Grid’s greatest strengths. To develop my deconstructed headline, I first break it into three components:
Next, I turn this headline into a grid container which contains three columns. The outer columns each occupy one-quarter of the container width (1fr). The central column occupies half (2fr):

```html
<h1>
  <span>Dial</span>
  <span>K</span>
  <span>for Murder</span>
</h1>
```

I didn't need to add classes to those spans, because using `nth-of-type()` pseudo-selectors means I can select each `<span>` and position them in my headline grid in this order:

```css
span:nth-of-type(1) {
  grid-column: 1;
}
span:nth-of-type(2) {
  grid-column: 2;
}
span:nth-of-type(3) {
  grid-column: 2 / -1;
}
```

Why stop when there's so much more I can do to brand this headline? I can colour that characteristic K red and add padding inside a thick border:

```css
span:nth-of-type(2) {
  padding: 12px 6px;
  color: #ba0d37;
  border: 18px solid #ba0d37;
}
```

Some people might think that styling elements to this degree is excessive, but I would disagree. Every text element provides an opportunity to tell a story, and when there's a chance you can improve storytelling through design, you should take it.
Decorative typography

You’ll find distinctive drop and initial caps in books from centuries ago, where they were used to decorate manuscripts and define sections in dense passages of text. On the web, you might use their visual weight to draw the reader’s eye to the start of a new section, but they can also be used to weave a brand’s personality into a design.

DECONSTRUCTED TYPE

Nicknamed “Brown Bread Fred”—“brown bread” is Cockney rhyming slang for dead—Freddie Foreman was a British gangster, twice convicted of armed robbery and murder, and an associate of Ronnie and Reggie Kray. In the 1960s, Freddie was one of the most feared gangsters in London.

You need a design which leaves its mark on the reader when telling a story like Freddie’s.

I use Grid and transforms to develop this biography page. (Photograph courtesy of Salon.)
To develop this design, I use a headline, `<figure>`, and a deconstructed `<figcaption>`, which I positioned to suggest someone’s brains being blown out. There’s nothing criminal about my markup; it looks like this:

```html
<body>
  <h1>Dial <span>K</span> for Murder</h1>
  <figure>
    <img src="img.jpg" alt="">
    <figcaption>
      <span>Foreman confessed to</span>
      <span>the murder of</span>
      <span>Frank “Mad Axeman” Mitchell</span>
      <span>who he shot 12 times</span>
    </figcaption>
  </figure>
</body>
```

Right out of the traps, I create eight equal-width columns on the `<body>` and align grid items vertically in the centre:

With CSS properties like these available now, there’s no excuse for developing tame designs.
Placing the headline and `<figure>` on the same row causes them to overlap. It also hints at the headline flying into Freddie's right ear:

```html
h1, figure {
  grid-row: 1;
}
h1 {
  grid-column: 1 / 5;
}figure {
  grid-column: 4 / -1;
}
```

Adding a second eight-column grid to the `<figure>`, I position Freddie's portrait `<img>` and the `<figcaption>` alongside:

You can use decorative typography to weave a brand's personality into a design.
I overlap that `<img>` and `<figcaption>` by placing them on the same grid row to suggest text is exploding from Freddie's left ear:

```css
figure {
  display: grid;
  grid-template-columns: repeat(8, 1fr);
  align-items: center;
}

img, figcaption {
  grid-row: 1;
}

img {
  grid-column: 1 / 7;
  object-fit: cover;
}

figcaption {
  grid-column: 6 / -1;
}

span {
  display: inline-block;
  margin-bottom: 6px;
  padding: .15em .25em .15em;
  background-color: #ba0d37;
  color: #fff;
}
```

To increase the movement in my `<figcaption>`, I rotate each `<span>` by a different number of degrees:

```css
span:nth-of-type(1) {
  transform: rotate(-5deg);
}

span:nth-of-type(3) {
  transform: rotate(5deg);
}

span:nth-of-type(4) {
  transform: rotate(10deg);
}
```
Data attribute effects

On Easter Monday 1983, a gang broke into a Security Express depot in Shoreditch, east London and escaped with £6m — around £30m today. At the time, this was the biggest cash haul in British history. For his part in the robbery, Freddie Foreman received a nine-year prison sentence.

A tinted photograph is a major part of the design of this story about that robbery.

To give this image the impact it deserves, I want it to fill half the width of the page and the full viewport height. Developing a plan to achieve this is no bother at all. My markup contains just three elements:
How to turn that lean markup into a grid should be second nature to you by now:

```html
body {
  display: grid;
  grid-template-columns: 1fr 1fr;
}

h1, header, main { grid-row: 1; }

h1 { grid-column: 1 / -1; }

header { grid-column: 1; }

main { grid-column: 2; }
```

When I place the headline on the same row as `<header>` and `<main>`, it overlaps them both. However, there's a flaw in this plan—when the headline's text is dark, there's not enough contrast between it and the image behind. Accessibility matters, so this is unacceptable. If I make the text light, I have the same problem against the white background.

*Left: Making sure there's plenty of contrast between background and foreground colours is one of the easiest things you can do to improve accessibility. Right: Whichever way you look at it, there's nothing clever about this lack of contrast.*
I need a solution, and luckily, Mandy Michael’s Text Effects\(^3\) provided one. It involves a small amount of presentational HTML, but not so much the markup police will chase me. All I need is to add a \texttt{data-} attribute to that headline:

\begin{verbatim}
<h1 data-title="Tooling up with Freddie Foreman">
  Tooling up with Freddie Foreman
</h1>
\end{verbatim}

Did you notice the headline text is repeated in that attribute? Mandy made me do it. Honest.

I need the value of that attribute so I can position it over the headline:

\begin{verbatim}
h1 {
  position: relative;
  text-align: center;
  color: #000;
}

h1:before {
  content: attr(data-title);
  position: absolute;
  overflow: hidden;
  color: #fff;
}
\end{verbatim}

Finally, the \texttt{clip-path} property clips this generated text halfway along its length. This lines up the change in colour with the centre of my layout, between the \texttt{<header>} and \texttt{<main>}:

\begin{verbatim}
h1:before {
  -webkit-clip-path: polygon(0 0, 50% 0, 50% 100%, 0% 100%);
  clip-path: polygon(0 0, 50% 0, 50% 100%, 0% 100%);
}
\end{verbatim}

\(^3\) https://smashed.by/texteffects
Text shadow effects

The shadows cast by gangsters in 1960s London haven’t faded in almost sixty years. In another article design about the 1983 Security Express robbery, shadows make the headline stand out. You can apply shadows to any text element, but sometimes a little help is needed to pull off more complex designs. Fortunately, data attributes and pseudo-elements are just the thing.

You can also display content from attribute values, including alt, title, and in this case data-. Data attributes keep information hidden from everything including screen readers, but they remain accessible to CSS and addressable by scripts.

Pseudo-elements enable you to display entities, images, or text between an element’s content box and padding box, so there’s no need to hard-code them into your HTML.
To create this three-dimensional effect using multiple shadows, I need another `data-` attribute for this new headline:

```html
<h1 data-title="Tooling up with Freddie Foreman"> Tooling up with Freddie Foreman </h1>
```

Unlucky for some, I apply thirteen text shadows to that headline, each with gradually larger horizontal and vertical offsets, which combine to create the effect of a single, longer shadow:

```css
h1 {
  text-shadow:
  -1px 0 0 #e2e2e2, 0 -1px 0 #e2e2e2,
  -2px -1px 0 #e2e2e2, -1px -2px 0 #e2e2e2,
  -3px -2px 0 #e2e2e2, -2px -3px 0 #e2e2e2,
  -4px -3px 0 #e2e2e2, -3px -4px 0 #e2e2e2,
  -5px -4px 0 #e2e2e2, -4px -5px 0 #e2e2e2,
  -6px -5px 0 #e2e2e2, -5px -6px 0 #e2e2e2,
  -7px -6px 0 #e2e2e2;
}
```

In the absence of a `text-stroke` property, I add a thin outline around the `:before` pseudo-element using four more thin text shadows. A browser generates the content of this pseudo-element from the data attribute:

```css
h1:before {
  content: attr(data-title);
  position: absolute;
  text-shadow:
  -1px -1px 0 #999, 1px -1px 0 #999,
  -1px 1px 0 #999, 2px 1px 0 #999;
}
```

I use the `:after` pseudo-element to add what appears to be the headline colour, before moving it right and up to enhance the three-dimensional effect:
Young designers and people who want to break into the industry often ask me, “Which aspect of design should I learn first?” My answer’s always the same. “Learn typography.” Some people think that type is just content—that’s why we have terms like content strategy or content creation—but type is about more than imparting information; it’s about connecting people to a story. And that’s why how your type looks is just as important as what it says. ●
Chapter 11

People pay attention to powerfully art-directed stories. They get noticed, and “An Unbelievable Story of Rape,” art-directed for the web by ProPublica’s design director David Sleight, won a Pulitzer Prize for Explanatory Reporting. This story is even more notable because it was written by two journalists. One reported from the woman’s perspective, the other from that of the police. In itself, this isn’t unusual, but instead of merging two voices into one, the editor kept them separate. As the art director, David recognised the two timelines, and alternating voices made the story more powerful, so he emphasised the distinctions through his design choices. The woman’s story uses specially commissioned illustrations, whereas the police story includes photography. Her story’s told in the left half of the page, theirs on the right, with the introduction and epilogue relevant to both placed in the centre. “An Unbelievable Story of Rape” is a compelling story, told more emotively because of its art direction.

https://propublica.org/article/false-rape-accusations-an-unbelievable-story
"This was a case where art direction literally set the editorial structure for the piece. It was done through collaboration with reporters and across two different newsrooms sharing a repo in GitHub."

— David Sleight
Developing with images

It shouldn’t have escaped your attention that many of my Wheel Man pages include full-height photographs like the portrait of Amy Johnson in this next design. A few years ago, it was challenging to develop designs like this because—while the dimensions of an image have always been predictable—the size and shape of a browser window or screen are not.

Fitting viewports

In the past, I might’ve used a percentage width of 50%. However, percentages are relative to the size of a parent container and not the dimensions of the viewport, which is what I need for this design.
When I use the term viewport, I’m referring to either the edges of a browser window or the sides of windowless, full-screen devices, such as phones or tablets.

VIEWPORT UNITS

As you might expect, viewport-based values are relative to either the viewport height, width, or whichever is larger (vmax) or smaller (vmin).

When I swap a percentage value for a viewport width (vw) unit, I get closer to the result I want:

```css
img {
  width: 50vw;
}
```

The image now occupies precisely half of the viewport width, no matter how large that becomes. However, there are problems. At narrower widths, my picture is too short to fill the page height and at wider widths it becomes too tall.
I could specify both `height` and `width` in viewport units, but this changes the picture's aspect ratio. That may be suitable for an abstract image, but not for something as recognisable as a portrait:

```css
img {
  width: 50vw;
  height: 100vh;
}
```

Changing the aspect ratio of recognisable subjects, including faces, is rarely a good thing to do.

Fortunately, there are two extra properties which help to solve this problem by changing the way a browser resizes images, videos, and other embedded media. These are the `object-fit` and `object-position` properties.

**FITTING OBJECTS**

If you've used the `background-size` property, you'll probably already be familiar with two of its most useful, if somewhat confusingly named, values:
contain Scales an image so that both its height and width stay contained inside an element. Neither are clipped and the original aspect ratio is preserved.

cover While maintaining the aspect ratio, both height and width change so that an image covers the entire box.

You can use both these values with object-fit to control how embedded objects scale to match the height and width of a box:

Clockwise from top-left: contain, cover, scale-down, and fill.
Setting a value of `fill` will alter the aspect ratio so that an object fills an entire box. `fill` is also a browser’s default behaviour, so there’s usually no reason to specify it. A value of `none` ignores the height and width of a box altogether, and so preserves an image’s original size. `scale-down` works in a similar way to `contain` and renders an image at the smallest possible size.

**POSITIONING OBJECTS**

`object-fit` lets a browser take care of cropping of images within a box. By default, objects scale from the centre (50% 50%) both horizontally and vertically. However, there’s one final `object-` property, `object-position`, which enables precise control over which parts of an object remain visible and which areas are cropped.

`object-position` does nothing on its own, but when combined with `object-fit` it defines the position of an object inside its box. Like many other CSS properties, including `background-position`, `object-position` uses two coordinates, where x is the horizontal distance from the top-left, and y the vertical distance. You can control an object’s position using any CSS length unit:

```css
img {
  width: 50vw;
  height: 100vh;
  object-fit: contain;
}
```

One of the most profound challenges of responsive design, and a fundamental difference from a tightly controlled fixed-width layout, is adjusting to less control over how images are displayed. Allowing an image to
scale to 100% has become one of the most common techniques in responsive design. But to get the most from an image, there are times when you need to stay in charge. By combining viewport units with \texttt{object-fit} and \texttt{object-position}, you can retain control over images without sacrificing responsiveness.

Layering images

In Chapter 9, I taught you how Grid makes developing layouts which bring art direction to life on the web more accessible than before. I’m
looking forward to seeing more artistic compositions in general; however, I’m particularly excited about how Grid enables designs which so far have been either difficult or even impossible.

It’s common in print to see design elements that overlap to create the illusion of depth. On the web, achieving this effect has been tricky, so design elements mostly stay separate.

For years we’ve been able to use CSS absolute positioning, margins, and transforms to create overlapping effects. The problem with these properties is they don’t affect the document flow, and this has made using them to develop responsive designs more difficult. Grid solves that problem, and developing art-directed layouts has never been easier, even with the day-to-day considerations of content management systems and dynamic content.
Because at present Grid does not inherit to form subgrids, to achieve this design, I need two grids: one for the `<body>`, and another for arranging images within a `<figure>`. My markup contains not even a scrap of presentational HTML and it looks like this:

```html
<body>
  <h1>Amy Johnson</h1>
  <main>…</main>
  <figure>
    <img src="img-1.jpg" alt="" />
    <img src="img-2.jpg" alt="" />
    <img src="img-3.jpg" alt="" />
    <img src="img-4.jpg" alt="" />
    <figcaption>…</figcaption>
  </figure>
</body>
```

The grids on `<body>` and `<figure>` each have different sets of columns and rows. Within the eight-column `<body>` grid, my `<main>` content occupies one-quarter (2fr) of the width, while the `<figure>` occupies the remaining three (6fr). These Grid properties should be very familiar by now:

```css
body {
  display: grid;
  grid-template-columns: 6fr 2fr;
  grid-template-rows: 1fr auto;
  column-gap: 2vw; }

main {
  grid-column: 6 / -1;
  grid-row: 2; }
```

I position my `<figure>` on the `<body>` grid and then turn it into a grid container to hold images and a `<figcaption>`:
Finally, I place those images and `<figcaption>` into columns and rows on my `<figure>` grid:

```html
figure {
  grid-column: 1 / 6;
  grid-row: 2;
  display: grid;
  grid-template-columns: repeat(6, 1fr);
  grid-template-rows: repeat(4, auto);
  grid-gap: 2vw;
}
```

```html
img:nth-of-type(1) {
  grid-column: 1 / 3;
  grid-row: 1 / 4;
}

img:nth-of-type(2) {
  grid-column: 3 / 6;
  grid-row: 1 / 3;
}

img:nth-of-type(3) {
  grid-column: 2 / 3;
  grid-row: 4 / 5;
}

img:nth-of-type(4) {
  grid-column: 3 / -1;
  grid-row: 3 / 5;
}

figcaption) {
  grid-column: 1;
  grid-row: 4 / 5;
}
```

Somehow, even with its unusual composition, this design feels static. I want to make it feel spontaneous and look as though the photographs fell onto the page. I also need those seemingly unstructured images to contrast with the structured look of the justified text alongside it. I could create the scattered look by using transforms to pull these images away from their grid lines:
Although simple to develop, this layout would’ve been tricky to make responsive using absolute positioning.

```css
img:nth-of-type(1) {
  transform: translateX(3rem);
}

img:nth-of-type(2) {
  transform: translate(-2rem, 3rem);
}
```

Somehow, moving these images with transform feels no better than using absolute positioning. Instead, I define a new grid, a different grid, one which contains the most complex set of columns and rows I’ve shown you so far.

```css
figure {
  display: grid;
  grid-template-columns: repeat(5, 1fr) 5fr 1fr 1fr;
  grid-template-rows: 75px 75px 150px 75px 75px 150px;
  grid-gap: 0;
  align-items: start;
}
```
You may be wondering why sometimes I choose to define my grid columns and rows using `grid-column` and `grid-row`, whereas other times I use `grid-area`. Grid template areas are ideal for developing layouts where no elements overlap. But elements placed in this way cannot occupy the same grid-area. On the other hand, elements placed using `grid-column` and `grid-row` can occupy the same module.

I position images and the `<figcaption>` to columns and rows on this new `<figure>` grid. Because some of these images occupy the same rows, they will overlap. `z-index` enables these pictures to stack visually, regardless of their place in the document flow:

```css
img:nth-of-type(1) {
  grid-column: 2 / 6;
  grid-row: 1 / 6;
  z-index: -1;
}

img:nth-of-type(2) {
  grid-column: 5 / 7;
  grid-row: 2 / 4;
  z-index: 1;
}
```
I add `rotate()` transforms to three of those images and now this scattered look is complete:

```css
img:nth-of-type(1) {
  transform: rotate(-5deg); }
img:nth-of-type(2) {
  transform: rotate(5deg); }
img:nth-of-type(4) {
  transform: rotate(10deg); }
```

Rotating these photographs so that they appear to have fallen onto the page is a playful touch and one that contrasts nicely with the structured column of running text.
STACKING IMAGES

To contrast with the previous design’s spontaneity, my next design looks highly structured. The headline, image caption, and four columns of running text all occupy grid modules and give my composition a feeling of formality. To reinforce the impression, a tall photograph floats above the page plane, and it casts a `box-shadow` over the columns below. To create even more depth I also blur the image in the background.

This design combines several techniques including the layering capabilities of Grid.

The markup for this design is compact and meaningful as it consists only of:

```html
<body>
  <h1>Amy Johnson</h1>
  <main>...</main>
  <aside>...</aside>
  <img src="img-h.jpg" alt="">
  <img src="img-v.jpg" alt="">
</body>
```
To develop this layout, I need an eight-column grid with six evenly spaced rows.

```css
body {
  display: grid;
  grid-template-columns: repeat(8, 1fr);
  grid-template-rows: repeat(6, auto);
  grid-gap: 2vw;
}
```

The process of arranging items on this grid should be familiar by now. First, I place the large horizontal photograph on the second row and allow it to span the entire page width:

```css
[src*="img-h"] {
  grid-column: 1 / -1;
  grid-row: 2 / 4;
  z-index: -1;
}
```

Now, the vertical portrait photograph rests over the top, spanning five rows. By bleeding off the top of the page, this tall image directs the eye down towards my `<main>` content. Giving this image a wide border and casting a semi-transparent shadow over my content helps create the illusion of depth:

```css
[src*="img-v"] {
  grid-column: 4 / 8;
  grid-row: 1 / 6;
  border: 10px solid #fff;
  box-shadow: 0 60px 30px 0 rgba(0,0,0,.25);
}
```

Finally, I position the headline, `<aside>`, and `<main> content on the grid and use a multi-column layout property to spread my running text across four columns:
In the not-too-distant past, blurring a photograph was something only done in an image editor such as Adobe Photoshop. Luckily today, we can use a CSS `filter` property to apply a tiny amount of blur to this horizontal image to increase the illusion of depth:

```css
[src*="img-h"] {
  filter: blur(2px);
}
```

Left: Without blurring or a shadow, this design falls flat. Right: By introducing a subtle blur effect and shadow, the design feels more dynamic.
I’m ridiculously excited about Grid. Yes, I know I probably should get a hobby, but I predict that its arrival means a turning point for art direction on the web. Our opportunity for doing creative work and for telling better stories online has never been greater. I cannot be more thrilled about that.

**CSS Shapes**

For years it’s been possible to use CSS borders, shadows, and transforms[^4] to turn HTML elements into all manner of fake shapes, from parallelograms to Pac-Man[^5].

```css
.pacman {
  width: 0;
  height: 0;
  border-right: 60px solid transparent;
  border-top: 60px solid #ba0d37;
  border-left: 60px solid #ba0d37;
  border-bottom: 60px solid #ba0d37;
  border-top-left-radius: 60px;
  border-top-right-radius: 60px;
  border-bottom-left-radius: 60px;
  border-bottom-right-radius: 60px;
}
```

While not coming close to the worst CSS I’ve ever seen, there’s something very unpleasant about using such convoluted styles to make shapes like this. However, the big problem with fake shapes is that other elements don’t see them as shapes. So a `<div>` might look like Pac-Man to you, but to everything around it, it’s still a plain old box.

This portrait of Amy Johnson might look like a circle, but to the document flow it stays a rectangle. Even when you float it, content flows only

[^4]: https://smashed.by/cssshapes
[^5]: https://smashed.by/enjoycss
around its canvas (the CSS reference box) because a browser pays no attention to what’s in the picture.

When CSS Shapes (Shapes) was introduced, everything changed, and now you can wrap content around a clipping path or an image alpha-channel. You can size shapes using flexible units and adjust them via media queries, which makes them brilliantly suited to responsive design.

Reference box is the technical term for the content, padding, border, and margin boxes which comprise the CSS Box Model. You can use them to control the size and position of a circle or any other basic shape. margin-box is the default, so there’s usually no need to specify that. When you’re looking for a different effect, change to border-box, content-box, or padding-box.

SHAPE-OUTSIDE

When you need content to flow around a shape, use the shape-outside property. You must float an element left or right for shape-outside to have any effect, but when it’s applied, content flows around any shape.
Developing with images

You may want content to flow around a shape which doesn’t precisely match the contents an image. By using a position value or a `closest-side` or `farthest-side` keyword, you can change its position within an element and subsequently how content flows around it. Position places a circle horizontally and vertically within the element (x for the horizontal and y for the vertical, in that order) using CSS length units:

```css
div {
    shape-outside: circle(8rem at 6rem 6rem);
    /* radius 8rem. 6rem from the left, 6rem from the top */
}
```

`closest-side` Length from the centre of a shape to the side of its reference box which is closest.
**farthest-side** Length from the centre of a shape to the side of its reference box which is farthest away.

```html
div {
  shape-outside: circle(farthest-side at 20% 40%);
  /* radius 50%, 20% from the left, 40% from the top*/
}
```

*Left: Circle with a position value of closest-side applied. Right: The same circle with a position value of farthest-side applied.*

As you might imagine, circles aren’t the only shapes available. You can flow text around an ellipse, an inset (rectangle), and a polygon.

**inset()** is a rectangle. Now, you could be wondering why this shape is needed at all, considering boxes are already rectangular. However, with **inset()** you can define a shape which has a size and position which are different from the reference box. Unsurprisingly, these values are called insets, and you use four of them: one for each side of a shape:

```html
div {
  shape-outside: inset(40px 20px 20px 40px round 20px);
  /* 40px from top, 20px from right and bottom, 40px from left*/
}
```

Insets follow the same round-the-clock, top-right-bottom-left pattern as **margin** and **padding**. Unlike regular reference rectangles, insets can be rounded. The **round** keyword in the declaration signals rounded
corners, and that final value is their corner radii. When you need rounded corners on both an element and a shape, you'll still need to apply `border-radius` to that element:

```css
div {
    shape-outside: inset(40px 20px 20px 40px round 2rem);
    border:.5rem solid #ba0d37;
    border-radius: 2rem;
}
```

`ellipse()` is a closed curved shape. Its values are very similar to `circle()` except that because it can vary in shape, you must specify two radii (x for the horizontal and y for the vertical, in that order):

```css
div {
    shape-outside: ellipse(); /* default values, or */
    shape-outside: ellipse(80px 40px);
    /* ellipse radius 80px horizontally, 40px vertically */
}
```

`polygon()` defines more complex shapes and consists of a series of comma-separated coordinate pairs. One pair specifies the position of each point in a shape (x for the horizontal and y for the vertical, in that order). This polygon has points which form a parallelogram:

```css
div {
    float: right;
    shape-outside: polygon(25% 0%, 100% 0%, 75% 100%, 0% 100%): }
```

The Firefox Shape Path Editor and Chrome's CSS Shapes Editor are both useful tools for making new polygons and inspecting the points in existing ones. I love Olivier Forget's incredible Shape-Outside Playground, which lets you try different images without committing too much time to code.

86 https://smashed.by/shapeoutside
When you flow content around shapes, be careful not to allow any lines of text to become too narrow and fit only one or two words. Also, make sure faces look towards your content or you might make people feel uncomfortable.

**Polygon shapes in action**

Polygons add movement to a design and this draws people in. They help to connect an audience with your story and make tighter connections between your visual and written content. It often needs surprisingly little markup to develop original layouts.
Developing with images

My HTML for this design consists of only two `<article>` elements and one image, and it looks like this:

```html
<body>
  <article>
    <img src="shape.png" alt="">
    <h1>Croydon Airport</h1>
    ...
  </article>
  <article>
    <h1>Amy Johnson</h1>
    ...
  </article>
</body>
```

Maintaining a consistent grid across a product or website is one way to link screens together, even if they look very different on the surface, and I use the same eight-column grid as in previous examples. That helps this page feel connected to other locations on my Wheel Man website.

I arrange the two `<article>` elements into asymmetrical columns. My most important story occupies a wide column, and the other occupies a much narrower column:

This is the design I’m developing: running text in the main story flows around the shape of an aeroplane propeller.
When you need to put some distance between a shape and content around it, the `shape-margin` property draws a further shape outside the contours of the first one. Codrops has a detailed explanation the technical aspects of `shape-margin`.87

When I apply `shape-outside` and `shape-margin` to this propeller image, running text from my most important story flows around its polygon points. This content reflows to match the width of any screen:

```
.layout-1 [src*="shape"] {  
  float: left;  
  shape-outside: polygon(0px 0px, 651px 0px, ...);  
  shape-margin: 2vw; }  
```

Left: Content for this important story about Croydon Airport flows around the propeller on the left. Right: When the main story flips to one about Amy Johnson, so does the layout.

87 https://smashed.by/shapemargin
INVISIBLE SHAPES

In the examples I’ve shown so far, I applied `shape-outside` to an HTML element in my document, but there will be times when you’ll need to create shapes without additional HTML. Fortunately, `shape-outside` can be also applied to `::before` and `::after` pseudo-elements:

```css
div::before {
  content: "";
  float: right;
  width: 250px;
  height: 250px;
  shape-outside: circle(50%);
}
```

POLYGON SHAPES SCULPT COLUMNS

You can create strong, structural shapes with nothing more than type. Combining `polygon()` and pseudo-elements, you can sculpt shapes from solid blocks of running text, in the style of Alexey Brodovitch and his influential work for Harper’s Bazaar.

```css
body {
  display: grid;
  grid-template-columns: 1fr 1fr;
  grid-gap: 2vw;
  max-width: 48em;
}
```

These beautiful numerals are almost too lovely to hide. They’re also perfect for carving into those columns.
When I use invisible pseudo-elements with no background or borders to develop polygon shapes, the result is two unusually shaped columns.

I formed these columns from two `<article>` elements, but this time I need a gutter between them and a maximum width, which helps maintain a comfortable measure:

Because there are two `<article>` elements and I also specified two columns for my grid, there's no need to be specific about the position of those articles. I can let a browser place them for me, and all that remains for me is to apply `shape-outside` to a generated pseudo-element in each column:

```
article:nth-of-type(1) p:nth-of-type(1)::before {
  content: "";
  float: left;
  width: 160px;
  height: 320px;
  shape-outside: polygon(0px 0px, 90px 0px, [...]); }

article:nth-of-type(2) p:nth-of-type(2)::before {
  content: "";
  float: right;
  width: 160px;
  height: 320px;
  shape-outside: polygon(20px 220px, 120px 0px, [...]); }
```
If you’ve designed using a tool like Sketch, as I as often do, you may know it lets you copy SVG code for a polygon or other shape. If you’d prefer to avoid the hard work of converting SVG polygons for use with `shape-outside`, developer Bear Travis has written a handy tool which does that for you.

### DEVELOP SHAPES FROM IMAGES

For me, one of the most incredible aspects of modern-day CSS is that I can create a shape from the alpha channel of a partially transparent image with no need to draw a polygon path. I only need to create an image and then a browser will take care of the rest.

The two shapes in this story about Amy Johnson and her famous flight between England and Australia were derived from images.

[https://smashed.by/pathtopoly](https://smashed.by/pathtopoly)
I think this is one of the most exciting additions to CSS and it makes developing art direction for the web more straightforward, especially if you work with a content management system and dynamically generated content.

To develop shapes from images, they must have an alpha channel which is either entirely or partially transparent. Next, I have two graphics: one, the shape of the United Kingdom; the second, an Australia-shaped portrait of Amy Johnson. When I exported these images from Sketch, I made the areas outside each country fully transparent.

```html
<main>
  <img src="uk.png" alt="">
  <img src="australia.png" alt="">
</main>
```

I needn’t draw a polygon to enable content to flow around the coastlines of these countries; instead, I need only specify the URL of an image file as the `shape-outside` value:

```css
[src*="uk"] {
  float: left;
  shape-outside: url(uk.png);
}

[src*="australia"] {
  float: right;
  shape-outside: url(australia.png);
}
```

I also add a small amount of `shape-margin` to push my running text away from these shapes:

```css
[src*="uk"],[src*="australia"] {
  shape-margin: 2vw;
}
```
Watch out for CORS (cross-origin resource sharing) when using images to develop your shapes. You must host images on the same domain as your product or website. If you use a CDN, make sure it sends the correct headers to enable shapes. It’s also worth noting that the only way to test shapes locally is to use a web server. The `file://` protocol simply won’t work.\(^9\)

**SETTING A THRESHOLD**

Occasionally the parts of an image you want to form a shape aren’t entirely transparent. If this is the case, the default `shape-image-threshold` value (0) won’t develop a shape. To fix that, adjust the threshold value to match the transparency level of your image. For example, if a background is 25% transparent, change the threshold value to at least \(0.25\):

```css
img {
  shape-image-threshold: 0.25;
}
```

However, I’ve found a little trial and error is often required when developing a shape using `shape-image-threshold`.

**DEVELOP SHAPES FROM MASK IMAGES**

In my previous design, the visible areas of those two country images also acted as shapes and content flowed around them. However, there may be times when you’ll need content to flow around a shape which is different from what’s visible.

For that, all you need do is specify a different image URL for `shape-outside` to develop a shape:

```html
<main>
  <img src="amy.jpg" alt="Amy Johnson">
</main>
```

---

89 [https://smashed.by/cors](https://smashed.by/cors)
Content now flows around an invisible shape where the mask is transparent and entirely different from the visible shapes in this portrait of Amy Johnson. That effect was impossible to achieve on the web before Shapes, and it opens up so many creative possibilities that I find containing my excitement about Shapes almost impossible too.

So far I’ve taught you how to use Grid to develop layouts which help you communicate messages and tell stories better. I’ve explained how Flexbox works hand-in-hand with Grid when developing smaller components, and how Multi-column Layout brings an editorial feel to a design. I’ve shown new ways to use writing modes to create vertical text, and how to use Grid and Flexbox to create effects with type. Lastly, I’ve explained how Shapes helps you achieve results that were once impossible on the web. Before I make my getaway, there’s one more CSS property I want to share with you.
Clipping paths

I am so excited by new CSS properties that help conquer our addiction to rectangular boxes. *clip-path* helps you quickly develop irregular shapes from a single image.

Clockwise from top-left: Circle, diamond, hexagon, star, pentagon, and heptagon.

*Bennett Feely’s CSS clip-path maker*[^1] is a great way to experiment using *clip-path*.

*clip-path* creates a clipping region in which parts of a picture remain visible and others disappear as if they never existed. This opens up enormous possibilities for developing original designs.

[^1]: https://smashed.by/clippy
BASIC CLIPPING REGION SHAPES

As with Shapes, you can create clipping regions from a selection of basic shapes including `circle()`, and `inset()`. The box sizing and position values used for Shapes also apply to clip paths. To define a clipping region, use the `clip-path` property:

```
img {
  clip-path: circle(50%);
/* or */
  clip-path: inset(40px 20px 20px 40px round 2rem);
}
```

*Left:* `circle()` with a radius of 50%. *Right:* `inset()` with 2 rem radius rounded corners.

CLIPPING POLYGONS

Clipping images into unusual shapes using a `polygon()` clip path is a great way to bring brand personality into a design. Like polygons in Shapes, clipping region polygons contain a series of comma-separated coordinate pairs where one pair specifies the position of a path point.
There’s very little I can say about the markup for this design, except I haven’t used even a scrap of presentational HTML:

```html
<body>
  <header>
    <img src="amy.jpg" alt=""/>
  </header>
  <main>…</main>
</body>
```

To emphasise the connection between my `<main>` content and a full-height photograph, I clipped it to form an arrow using this simple `polygon()`:

```css
header img {
  clip-path: polygon(15% 0, 100% 0, 100% 100%, 15% 100%, 0% 50%); }
```
The same clipping region will be visible when I replace this photograph, making this technique perfect for websites with dynamic content and a CMS.

CLIPPING FROM SVG PATHS

For the final design in this book, I want to combine a portrait photograph of Amy Johnson with the shape of Australia. In the past, this would’ve meant making the image in Sketch and exporting it as a PNG.

A portrait photograph of Amy Johnson clipped by the shape of Australia.

Changing that photograph would’ve meant making a new image every time, which could be inconvenient at best, impossible at worst for sites using a CMS. Clipping regions offer an elegant solution to this problem.
For complex polygons, I’ve found the easiest way to determine coordinate pairs for a clip path is to create an image using Sketch and export it as an SVG. It’s not possible to copy SVG path coordinates directly into a style sheet, as they need to be converted into a different format. Bear Travis’ Path to Polygon Converter[^1] is the best tool I’ve found to handle these conversions:

```
[src*="amy"] {
  clip-path: polygon(905.0% 399.8%, 904.8% 399.3%, ...); }
```

When `polygon()` contains more than a handful of coordinate pairs, a simpler alternative is specifying the URL of an SVG file and the ID of its path as `clip-path` values:

```
[src*="amy"] { clip-path: url(mask.svg#path-australia); }
```

It’s worth noting that when you define a `clip-path` using an image, it will not animate or transition when a state changes. One of the best features of `clip-path` is that you can change, transition, and even animate them between states. As one final touch to this design, I made a clipping region with the shape of the United Kingdom. I want that clipping region to transition smoothly between the shapes of the UK and Australia while keeping the same portrait photograph behind. First, I need coordinate pairs for the shape of the UK:

```
[src*="amy"] {
  clip-path: polygon(0.956% 30.674%, 0.658% 30.434%, ...); }
```

I will be replacing this set of coordinates when a state changes, so add a smooth transition between the two states:

```
[src*="amy"] { transition: clip-path 1s ease; }
```

[^1]: https://smashed.by/converter
Next, I change those coordinate pairs to a set which forms the shape of Australia:

```css
[src*="amy"]:hover {
  clip-path: polygon(905.0% 399.8%, 904.8% 399.3%, 904.5% 397.4%);
}
```

Now, when someone interacts with this design, the `polygon()` clipping region will transition smoothly between the shape of the UK and Australia without affecting the image it clips.
The pay-off

What good is art direction for the web if we don't have the layout or other technologies we need to get the results we're aiming for?

Guess what? We do.

CSS Grid and Flexbox are the best layout systems for the web so far, and they're perfect for developing art direction at every screen size. We have access to a massive library of typefaces and newer technologies like variable fonts to set type. Properties like clip-path and shape-outside give us countless opportunities to use art direction to capture someone's attention and then keep them engaged.

I hope by now you're as excited about all this as I am. Whether you make websites for a business, charity, government, or news outlet and you want to tell stories more effectively, you now have everything you need to make great art direction for the web.
Time to get away

OK. Now, like my Wheel Man drivers, it’s time for me to make a speedy getaway. When I set out to write this book, I wanted to teach people about the importance of art direction for the web. My aim was to explain how art direction can help people tell stories by using design, so that products and websites will connect with audiences and keep them engaged.

I wanted to tell how art direction can improve someone’s experience and maintain brand values and design principles by connecting touch points throughout marketing and products. I wanted every kind of business to understand the benefits art direction can bring them on the web. I also wanted web designers and developers to learn how to make art direction work for digital products and websites, and for experienced art directors to know that their skills can be as valuable for the web as they are for print.

Whether you design websites which run on platforms like Shopify or Squarespace, or you handcraft every aspect of a digital experience, art direction will make your sites more effective at communicating, persuading, and selling. If you develop products, art direction will make them more compelling and more enjoyable to use.

I hope this book has inspired you to think about art direction and how it can make your work, and the web, much better for everyone. Is art direction relevant for the web? Absolutely. Do we have the technologies and tools to deliver art direction for any product or website? There’s no doubt about that.

I’ve played my part, the rest is up to you.
Further reading

Alexey Brodovitch
Kerry William Purcell
amzn.to/2FGp7ol

American Modernism: Graphic Design, 1920 to 1960
R Roger Remington
amzn.to/2U1SQMl

The Anatomy of Type: A Graphic Guide to 100 Typefaces
Stephen Coles
amzn.to/2CYaaOw

Art Direction Explained, At Last!
Steven Heller and Veronique Vienne
amzn.to/2FRwIR0

The Art Directors' Handbook of Professional Magazine Design
Horst Moser
amzn.to/2xb704E

Combining Typefaces
Tim Brown
tinyurl.com/combining-typefaces-book

CSS Secrets: Better Solutions to Everyday Web Design Problems
Lea Verou
amzn.to/2NHPyy8

Designing for Emotion
Aarron Walter
amzn.to/2pF4gsy

Design Systems
Alla Kholmatova
amzn.to/2q4c2Mo

Flexible Typesetting
Tim Brown
tinyurl.com/typesetting-book

The Graphic Language of Neville Brody
Jon Wozencraft
amzn.to/2DnoZoO

The Graphic Language of Neville Brody 2
Jon Wozencraft
amzn.to/2xQfgqQ

Harper's Bazaar: 150 Years: Greatest Moments
Glenda Bailey
amzn.to/2GJvfgt

Hegarty on Advertising: Turning Intelligence into Magic
John Hegarty
amzn.to/2E6OWeG
Layout Essentials: 100 Design Principles for Using Grids
Beth Tondreau
amzn.to/2HqPaNu

Making and Breaking the Grid
Timothy Samara
amzn.to/2q5Dk4z

The Modern Magazine: Visual Journalism in the Digital Era
Jeremy Leslie
amzn.to/2FClCfq

On Advertising
David Ogilvy
amzn.to/2Jlt8Oa

O Design de Bea Feitler (Portuguese Brazilian)
Bruno Feitler
amzn.to/2Uds4Re

Ruth Ansel
Hall of Femmes
tinyurl.com/ansel-book

Sagmeister: Made You Look
Stefan Sagmeister
amzn.to/2xcdBeU

The Story of The Face: The Magazine that Changed Culture
Paul Gorman
amzn.to/2QsPlbj

Super Graphic: A Visual Guide to the Comic Book Universe
Tim Leong
amzn.to/2QvXAZq

Thoughts on Design
Paul Rand and Michael Bierut
amzn.to/2p8tZJD

Ugly Is Only Skin-Deep: The Story of the Ads that Changed the World
Dominik Imseng
amzn.to/2pPxjJI

Web Typography
Richard Rutter
book.webtypography.net