

GD3

Name: Leo Roberts
Project title:
Project number: 4
Tutor: Jason Holroyd
Date:

Project description

Identify a topic and design a website around it. Fundamentally demonstrate how UX and UI impact the user and how they are integral to modern digital design. Consider target audience if there is a call to action, if so, what is it? How will content differ when made responsive and compatible for a world of evermore new devices and dimensions.

Project aims

- Improve skills required to gather and record relevant research.
- Develop conceptual thinking and unique ways of presenting ideas.
- Improve my skills in developing final solutions.
- Use different technologies to prototype, test, and craft a response to the brief.
- Improve knowledge of programming languages such as, HTML, CSS, and JavaScript.
- Plan, wireframe and prototype an effective website that is responsive across multiple different devices.
- Improve my ability to design for websites and applications.
- Produce a website that demonstrates UX and UI design literacy.
- Improve understanding of programmes, such as: Figma, Adobe XD, Web-flow.

Proposed plan of work

- Explore a range of responses to the brief
- Choose a path and develop. Demonstrate how UX and UI are integral to modern digital design.
- Carry out varied research on the topic to inform my outcome.
- Plan, wireframe, and design a responsive website.
- Mock-up up website digitally (using applications such as: Figma, Adobe XD, Web flow etc.)
- Present outcome coherently

Coding Practise: Tribute Page

Practising HTML and CSS.

Website has no design merit at all, nothing's aligned properly but I did manage to get things on the page and make an unordered list, using the `` element.

Top left : HTML

Top right: CSS

Bottom: Final Outcome

I might come back to this at a later date to improve the CSS and add more content.

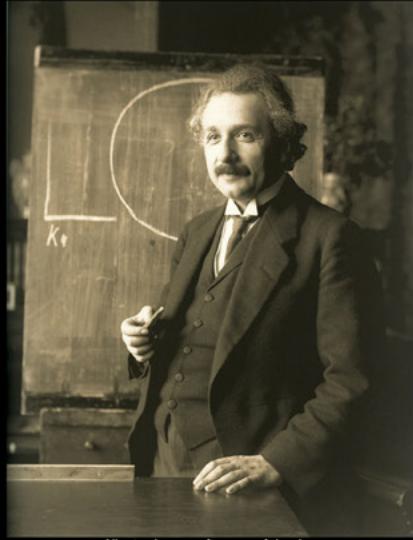
```
1 <style>
2 body {
3     background-color: black;
4     color: white;
5     font-family: helvetica;
6 }
7 </style>
8 <div id="main">
9     <h1 id="title">
10        Albert Einstein
11    </h1>
12    <div id="img-div">
13        <a id="tribute-link" target="_blank"
14            href="https://en.wikipedia.org/wiki/Albert_Einstein">
15            <a/>
16            <img id="image">
17                src="https://upload.wikimedia.org/wikipedia/commons/thumb/3/3e/Einstein_1921_by_F_Schmutzler_-_restoration.jpg/800px-Einstein_1921_by_F_Schmutzler_-_restoration.jpg"
18                alt="einstein with chalk" height="28%" width="28%">
19        <div id="img-caption">Albert enjoys an afternoon of drawing on his chalkboard.
20    </div>
21    <div id="tribute-info">
22        Albert Einstein was a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is best known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics
23    </div>
24 </div>
25
26 <p id="tribute-info" class="tree">
27 <b>Alberts skills:</b>
28 <ul>
29     <li> Smartness </li>
30     <li> Cool </li>
31     <li> wealth </li>
32 </ul>
33 </p>
34 </div>
35
36 </style>
```

```
1 #tribute-info {
2     width: 300px;
3     margin: 0 auto;
4 }
5 #img-caption {
6     width: 300px;
7     margin: 0 auto;
8 }
9
10 #image {
11     max-width: 100%;
12     height: auto;
13     display: block;
14     margin: 0 auto;
15 }
16 .tree {
17     margin: 0 auto;
18 }
```

Albert Einstein

Albert Einstein was a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is best known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics

- Smartness
- Cool
- wealth



Albert enjoys an afternoon of drawing on his chalkboard.

Learning HTML+CSS

Using freecodecamp to learn the basics of HTML + CSS.

FreeCodeCamp gives u a lot of challenges and tutorials that walk you through different elements of coding. it's very easy and doesn't challenge me much but it does lay things out very clearly. I think it's a good starting point before moving onto personal projects .

Search 8,000+ tutorials

Responsive Web Design Basic HTML and HTML5

Use HTML5 to Require a Field ✓

You can require specific form fields so that your user will not be able to submit your form until he or she has filled them out.

For example, if you wanted to make a text input field required, you can just add the attribute `required` within your `input` element, like this: `<input type="text" required>`

Make your text `input` a required field, so that your user can't submit the form without completing this field.

Then try to submit the form without inputting any text. See how your HTML5 form notifies you that the field is required?

Run the Tests (Ctrl + Enter)

Reset All Code

Get Help -

Your text `input` element should have the `required` attribute.

```
freeCodeCamp(A)
1 <h2>CatPhotoApp</h2>
2 <main>
3   <p>Click here to view more <a href="#">cat
photos</a>.</p>
4
5   <a href="#"></a>
6
7   <p>Things cats love:</p>
8   <ul>
9     <li>cat nip</li>
10    <li>laser pointers</li>
11    <li>lasagna</li>
12  </ul>
13  <p>Top 3 things cats hate:</p>
14  <ol>
15    <li>flea treatment</li>
16    <li>thunder</li>
17    <li>other cats</li>
18  </ol>
19  <form action="https://www.freecatphotoapp.com/
submit-cat-photo">
20    <input type="text" placeholder="cat photo
URL">
21    <button type="submit">Submit</button>
22  </form>

/**
* Your test output will go here
*/
```

Menu

CatPhotoApp

Click here to view more [cat photos](#).



Things cats love:

- cat nip
- laser pointers
- lasagna

Top 3 things cats hate:

1. flea treatment
2. thunder
3. other cats

cat photo URL Submit

Example of lesson.

Coding Practise: Survey Form

Practising HTML

No design on this website.
Was playing with `<labels>`,
`<input>` fields, and radio
buttons.

Right: HTML
Left: Final outcome

Need to add CSS.

```
<h1 id="title"> Flamethrower Application </h1>
<p id="description">
  This is a survey form to see if you are fit to handle a flame thrower </p>
<form id="survey-form">
  <label for="name" id="name-label"> Name </label>
  <input id="name" type="text" name="name" placeholder="type here" required>
  <br>
  <label for="email" id="email-label"> Email </label>
  <input type="email"
        name="email"
        id="email"
        class="form-control"
        placeholder="Enter your email"
        required>
  <br>
  <label for="number" id="number-label"> number </label>
  <input type="number"
        name="number"
        min = 10
        max = 100
        id="number"
        class="number"
        placeholder="Enter your number"
        required>
  <br>
  <label for="dropdown" id="type-label"> Type </label>
  <select id="dropdown">
    <option value="big flame"> Big flame </option>
    <option value="small flame"> Small flame </option>
    <option value="lighter"> Lighter </option>
    <option value="flasher"> Flasher </option>
    <option value="killer"> Killer </option>
  </select>
  <br>
  <label for="model_color" id="model_color"> <b> Model Colour </b> </label>
  <br>
  <br>
  <input id="model_color" type="radio" id="black" name="color" value="black">
  <label for="model_color"> Black </label>
  <br>
  <input id="model_color" type="radio" id="red" name="color" value="red">
  <label for="model_color"> Red </label>
  <br>
  <input id="model_color" type="radio" id="white" name="color" value="white">
  <label for="model_color"> White </label>
  <br>
  <br>
  <label for="Features"> <b> Features: </b> </label>
  <br>
  <br>
  <input id="checkbox" type="checkbox" name="features" value="silencer">
  <label for="checkbox"> Silencer (£500) </label>
  <br>
  <br>
  <input id="checkbox" type="checkbox" name="features" value="flashlight">
  <label for="checkbox"> flashlight (£700) </label>
  <br>
  <br>
  <input id="checkbox" type="checkbox" name="features" value="fuel_tank">
  <label for="checkbox"> Extra Fuel Tank (£2500) </label>
  <br>
  <br>
  <input id="checkbox" type="checkbox" name="features" value="grenade_launcher">
  <label for="checkbox"> Grenade Launcher (£5000) </label>
  <br>
  <br>
  <label id="textarea"> Additional Comments </label>
  <br>
  <br>
  <textarea id="instructions" name="instructions" rows="5" cols="40">
  </textarea>
  <br>
  <br>
  <input id="submit" type="submit" value="placeorder">
</form>
```

Flamethrower Application

This is a survey form to see if you are fit to handle a flame thrower

Name

Email

number

Type

Model Colour

Black

Red

White

Features:

Silencer (£500)

flashlight (£700)

Extra Fuel Tank (£2500)

Grenade Launcher (£5000)

Additional Comments

Coding Practice: Product Page

Practising HTML and CSS

Everything is really jumbled together, and I was struggling to fix this. In this project I tried to add `<video>` onto the page but it wasn't successful. I tried to add a toolbar but that didn't work either.

Right: CSS

Left: HTML

Bottom: Outcome

I want to re-visit this project once my skills have improved, I found some of it quite confusing.

```
1<h1> Electric Trousers </h1>
2<header id="header" id="img">
3
4
5<nav id="nav-bar" >
6  <h1> Electric Trousers &nbsp; &nbsp; &nbsp;
7  <a class="nav-link" href="#Collection>Collection </a> &nbsp; &nbsp; &nbsp;
8  <a class="nav-link" href="#Shop>shop </a> &nbsp; &nbsp; &nbsp;
9  <a class="nav-link" href="#My-Story>My story </a> &nbsp; &nbsp; &nbsp;
10 </h1>
11 
19   </img>
20 </header>
21
22
23<body>
24
25  <section id="Collection">
26  </section>
27
28  <section id="Shop">
29  </section>
30
31  <section id="My-Story">
32  </section>
33
34 <div id="overlaylayer" align="left" style="width: 100%; overflow: hidden" id="video">
35
36   <h2> The Future of Garments </h2>
37
38   <video id="video" controls>
39   <source src="https://www.youtube.com/watch?v=Dc3AovUZgvo" type="video/mp4" mute>
40   </video>
41
42 </body>
43
44<footer>
45  <form id="form" action="https://www.freecodecamp.com/email-submit">
46    <input
47      id="text"
48      type="text"
49      name="text"
50      placeholder="Have a Question?"'
51      Class="text"
52      required>
53    </input>
54    <br>
55    <input
```

```
56      id="email"
57      type="email"
58      name="email"
59      placeholder="Enter email"
60      Class="email"
61      required>
62    </input>
63    <br>
64    <input
65      id="submit"
66      type="submit"
67      name="text"
68      value="Ask Now"
69      Class="submit">
70  </input>
71 </form>
72
73
74 </footer>
```

```
1<header {
2  position: fixed;
3  Top: 0px;
4  width: 100%;
5  height: 19px;
6  padding:25px 0px;
7  display: flex;
8  Justify-content: space-between;
9  align-items: center;
10 background-color: black;
11 opacity:0.7;
12 }
13
14 #header-img {
15  margin: auto;
16  width: 10px'
17 }
18
19 @media (max-width: 500x) {
20  body {
21    flex-wrap: wrap;
22  }
23
24 }
```



Have a Question?
Enter email
Ask Now

Coding Practice: Document Page

Practising HTML and CSS

Simple page, practising lists again. I tried using links aswell (the blue text at the top takes you down the page to the corresponding company).

Right: Outcome

Middle: HTML + CSS

Left: HTML

```
1  <nav id="navbar">
2    <header> Formula One Team Rundown </header>
3    <ul>
4      <li> <a class="NavLink" href="#mercedes> Mercedes </a> </li>
5      <li> <a class="NavLink" href="#ferrari> Ferrari </a> </li>
6      <li> <a class="NavLink" href="#mclaren> McLaren </a> </li>
7      <li> <a class="NavLink" href="#alpine> Alpine </a> </li>
8      <li> <a class="NavLink" href="#redbull> RedBull </a> </li>
9      <li> <a class="NavLink" href="#haas> Haas </a> </li>
10     <ul>
11       <br>
12     </ul>
13   <br>
14   <br>
15 <main id="main-doc">
16   <section class="main-section" id="mercedes">
17     <header> Mercedes</header>
18     <p>
19       <code> The fastest car! </code>
20     </p>
21     <p>
22       <ul>
23         <li>Best Car</li>
24         <li>Best driver</li>
25       </ul>
26     </p>
27   <br>
28   <section class="main-section" id="ferrari">
29     <header> Ferrari</header>
30     <p>
31       <code> The classiest team! </code>
32     </p>
33     <p>
34       <ul>
35         <li>Best history</li>
36         <li>Best fans</li>
37       </ul>
38     </p>
39   <br>
40   <section class="main-section" id="mclaren">
41     <header> McLaren</header>
42     <p>
43       <code> The funniest team! </code>
44     </p>
45     <p>
46       <ul>
47         <li>Best humour</li>
48         <li>Best livery</li>
49       </ul>
50     </p>
51   <br>
52   <section class="main-section" id="alpine">
53     <header> Alpine</header>
54     <p>
55       <code> The newest team </code>
56     </p>
```

```
57   <p>
58     <ul>
59       <li>Best defencne in Hungary</li>
60       <li>Best Frenchman</li>
61     </ul>
62   </p>
63   <br>
64   <section class="main-section" id="redbull">
65     <header> Red Bull</header>
66     <p>
67       <code> The darkside </code>
68     </p>
69     <p>
70       <ul>
71         <li>Stroppiest driver</li>
72         <li>Worst livery</li>
73       </ul>
74     </p>
75   <br>
76   <section class="main-section" id="haas">
77     <header> Haas</header>
78     <p>
79       <code> The worst team! </code>
80     </p>
81     <p>
82       <ul>
83         <li>worst Car</li>
84         <li>worst drivers</li>
85       </ul>
86     </p>
87   </main>
88
89
90
91
92
93 #navbar {
94
95   position: auto;
96   height: 100%;
97   min-width: 300px;
98   width: 300px;
99
100  top: 0px;
101  left: 0px;
102
103  color: black;
104  background-color: white;
105  padding: 75px 0px;
106
107 }
```

Formula One Team Rundown

- [Mercedes](#)
- [Ferrari](#)
- [McLaren](#)
- [Alpine](#)
- [RedBull](#)
- [Haas](#)

Mercedes

The fastest car!

- Best Car
- Best driver

Ferrari

The classiest team!

- Best history
- Best fans

McLaren

The funniest team!

- Best humour
- Best livery

Alpine

The newest team

- Best defencne in Hungary
- Best Frenchman

Red Bull

The darkside

- Stroppiest driver
- Worst livery

Haas

The worst team!

- worst Car
- worst drivers

CSS Box Model

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model.

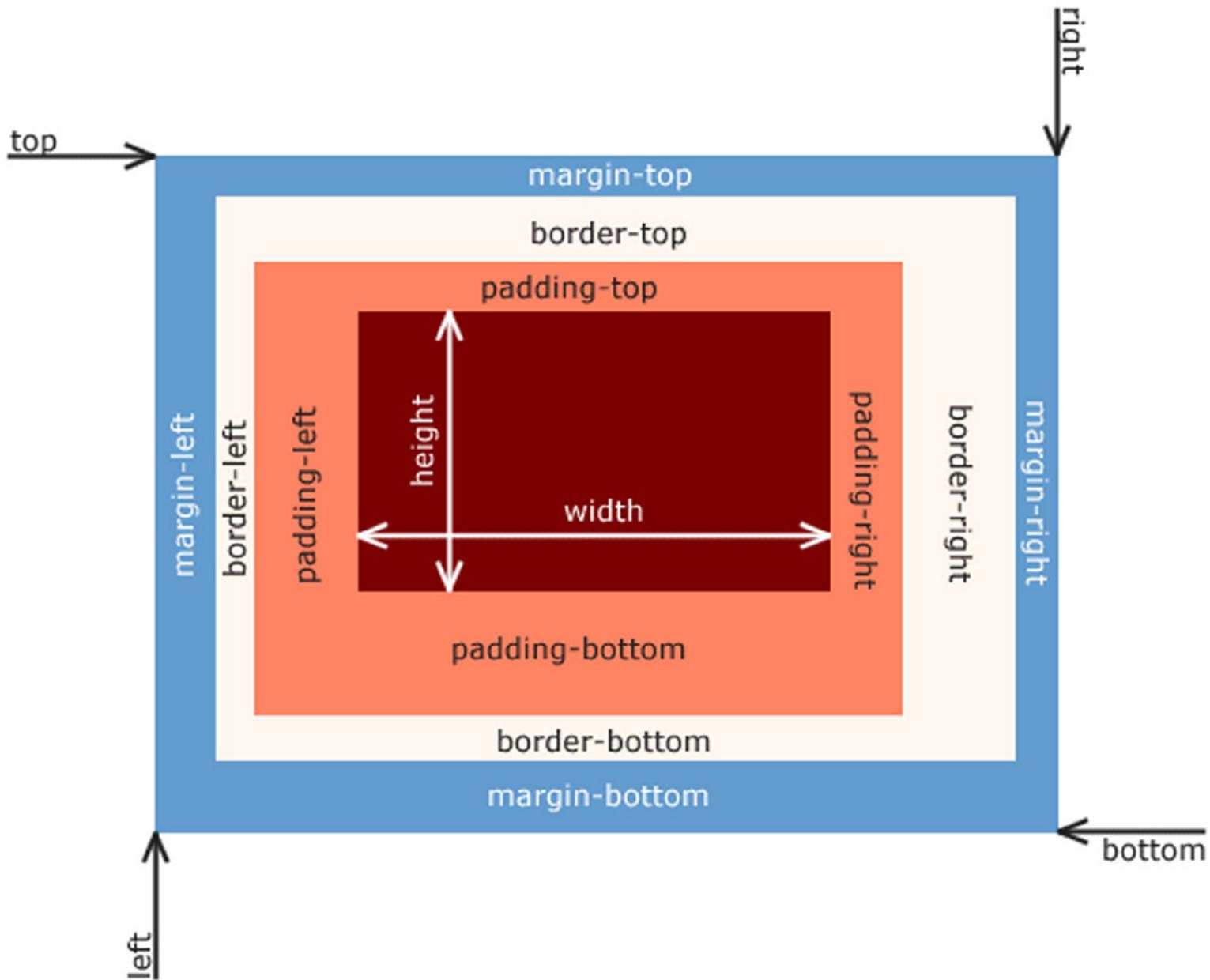
Content - The content of the box, where text and images appear

Padding - Clears an area around the content. The padding is transparent

Border - A border that goes around the padding and content

Margin - Clears an area outside the border. The margin is transparent

https://www.w3-schools.com/css/css_box-model.asp



Brief Change

Not doing a self-initiated brief anymore. I think it suits my personal direction more to focus on just coding and not UX/UI or even web design. I just want to do practise and learn how to code so In the future I can use my design skills to make really nice websites.

I'm going to go back to a previous brief 'Celebrate Crafts'. I think this suits me more. I want coding to be the craft and i'm going to celebrate it by learning it and making little projects that will come together as one final outcome.

GD3

Title: **Celebrate craft**

Assessment weighting: **100%**

Type: **Staff open brief**

Tutors: **JD ES HH KC JH TW NH SMC SM**

JN JR RN

Issue date: **28 September 2021**

Introduction

Craft has seen resurgence in recent years, there has been a backlash against technology and soulless mechanization, as small designer/makers have been championing the art of creating things by hand once again. Designers, such as Anthony Burrill, have eschewed modern technology and 'gone back to basics' to use ancient processes, such as letterpress and screen printing to make their work. Ceramist Denise Roberts digs her own clay from vacant land, constructs sculptural forms and then fires the pieces naturally by building an 'earth kiln' around them and using wood to stoke it over a period of a few days to produce truly original pieces. It is these kinds of passionate people that produce amazing work that has real personality that sets it apart from mass production.

Project aims

To develop appropriate research and fact-finding skills. To develop your ideas generation for concepts and proposals. To improve your skills in interpreting information and developing final solutions from them, which are relevant to the needs of the client and/or user. To further develop and improve personal visual, technological and time management skills.

The Brief

Your task is to explore and celebrate craft and all that could entail. The amount of craft makers to choose from is endless, but some you could consider are brewing, weaving, lacemaking, knitting, wood turning, tailoring or even sculpting, painting and metalwork. Research your chosen skill, discover as much as you can about it. Understand it, then devise a proposition and communicate it in an appropriate and inventive way. Think carefully about what you want to say about the subject matter. Are you an advocate of it? Do you want to tell a personal story? Do you want to show the process? Are you looking at its origins or history? Are you wanting to get rid of its 'hippy' perception and make it more acceptable? For example, you could interview a craft maker and document their process or look at ways of promoting it. Your outcome could take the form of an exhibition, TV title sequence for a proposed documentary series or even an interactive piece. The choice is yours but should be sympathetic to your chosen subject matter. Be clear on what message you are communicating to your audience. What is it you are trying to say and to whom?

Coding Practice: Recipe Page

Practising HTML and CSS

Used Visual Studio Code for this, coded each page separately and created a link between the two using a 'href'.

Top: Two coded pages
Bottom: 2 pages how they appear on the web.

I want to develop this with CSS .

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=<device-width>, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1> Odin Recipes</h1>
  <a href="#">recipes/BigMac.html">Big Mac</a>
</body>
</html>
```

```
<body>
<h1> BIG MAC </h1>
images/bigmac.jpg">
<h5> The Big Mac is a hamburger consisting of two 1.6 oz (45.4 g) beef
```

Odin Recipes

[Big Mac](#)



The Big Mac is a hamburger consisting of two 1.6 oz (45.4 g) beef patties, Iceberg lettuce, American cheese, pickles, chopped onions and special Mac Sauce (a Thousand Island dressing variant, though the recipe remains

Ingredients

- Buns
- Burger Patties
- Tomatoes
- Lettuce
- Seeds
- Ketchup
- Mustard
- Pickle
- Onions
- American Cheese

Steps

1. Add sauce to the buns
2. Put in patty no1
3. Add cheese
4. Add onions, lettuce, tomatoes
5. Add sauce
6. Add bun no2
7. Add patty no2
8. More onions, lettuce and tomatoes
9. More Sauce
10. Add final bun

User Experience (UX)

The user experience (UX or UE) is how a user interacts with and experiences a product, system or service. It includes a person's perceptions of utility, ease of use, and efficiency.

Improving user experience is important to most companies, designers, and creators when creating and refining products because negative user experience can diminish the use of the product.

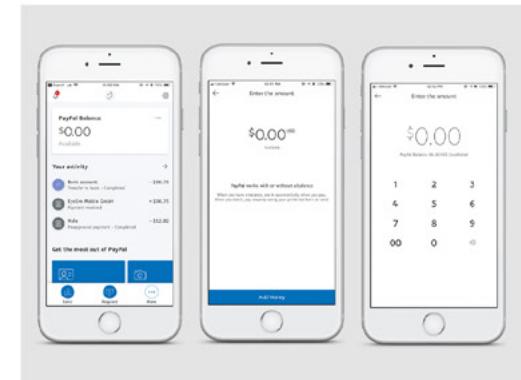
(https://en.wikipedia.org/wiki/User_experience)

The personal, internal experience customers go through when using a product's interface.

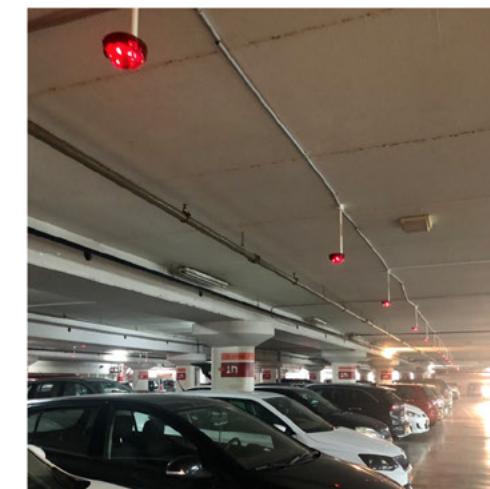
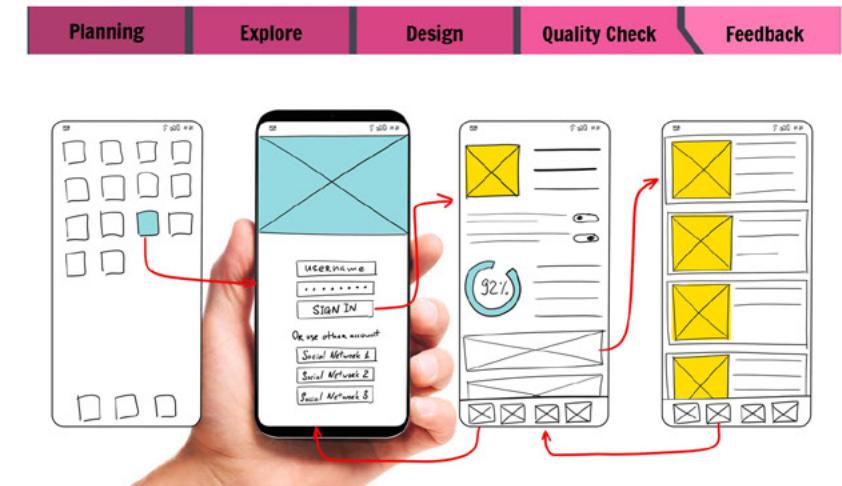
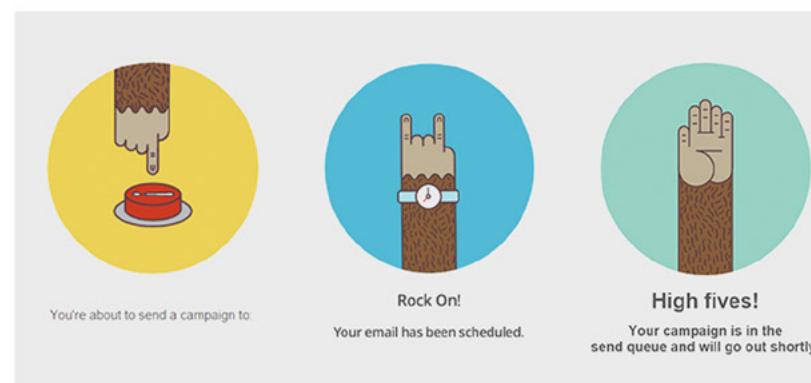
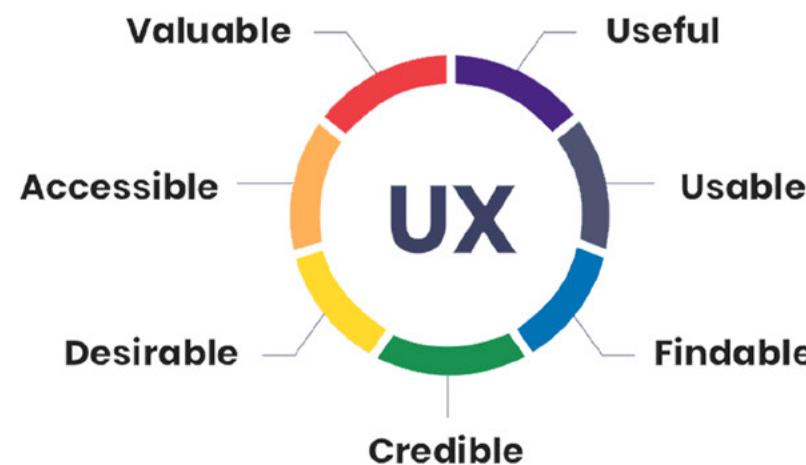
a meaningful user experience allows you to define customer journeys on your product that are most conducive to business success

UX Designer Key Responsibilities

- Content/Strategy: Customer Analysis, Competitor Analysis, Product Structure/Strategy
- Prototyping and **Wireframing**: Prototyping, Testing/Iteration, Development, Planning, **Wireframing**
- Analytics and Execution: Coordination with Developer(s), Coordination with UI Designer(s), Analysis and Iteration, Tracking Goals and Integration



UX design should be:



Light above cars, signalling if the space is full, empty or disabled.

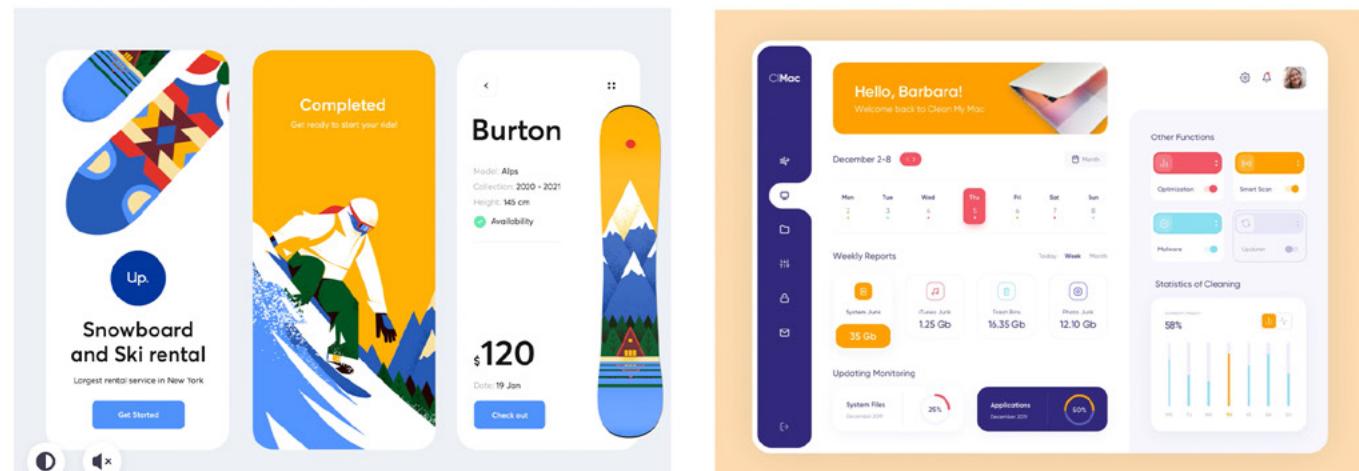
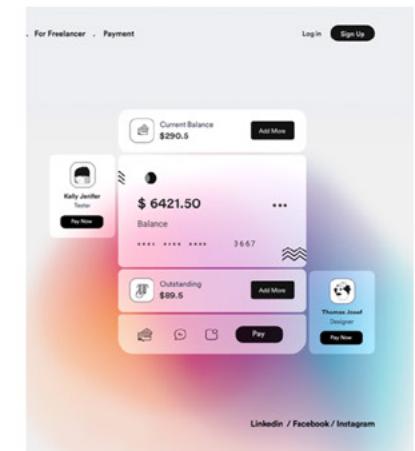
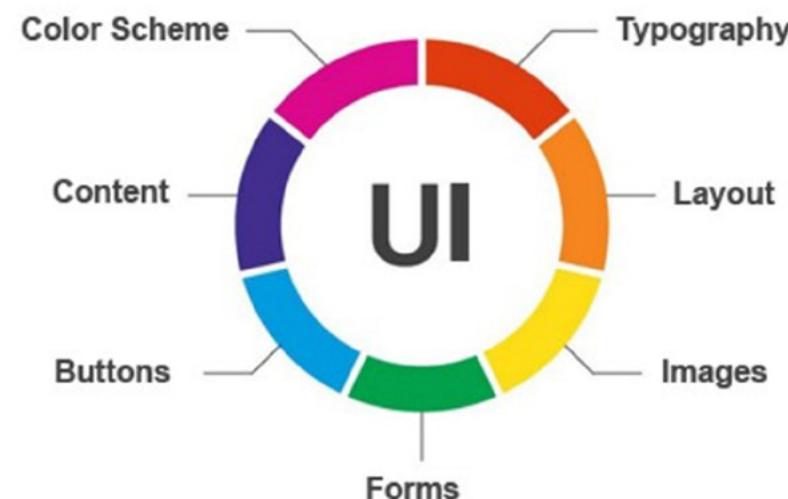
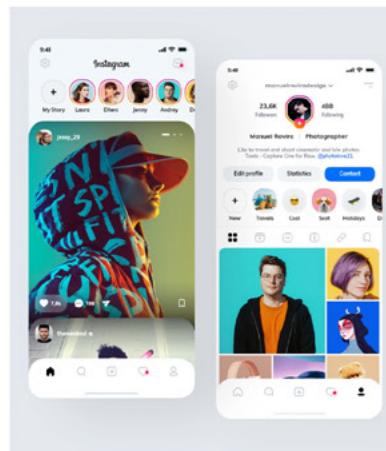
User interface (UI)

User Interface (UI) Design focuses on anticipating what users might need to do and ensuring that the interface has elements that are easy to access, understand, and use to facilitate those actions. UI brings together concepts from interaction design, visual design, and information architecture.

<https://www.usability.gov-/what-and-why/user-interface-design.html>

UI Designer Key Responsibilities

- Look and Feel: Branding and Graphic Development, User Guides/Storyline, Customer Analysis, Design Research
- Responsiveness and Interactivity: Adaptation to All Device Screen Sizes, Interactivity and Animation, Implementation with Developer, UI Prototyping, Implementation with Developer



1. Make Everything the User Needs Readily Accessible
2. Be Consistent
3. Be Clear
4. Give Feedback
5. Use Recognition, Not Recall
6. Choose How People Will Interact First
7. Follow Design Standards
8. Elemental Hierarchy Matters
9. Keep Things Simple
10. Keep Your Users Free & In Control

The Difference Between UX and UI

I found this and think it explains the difference in very basic terms.

+ Other examples

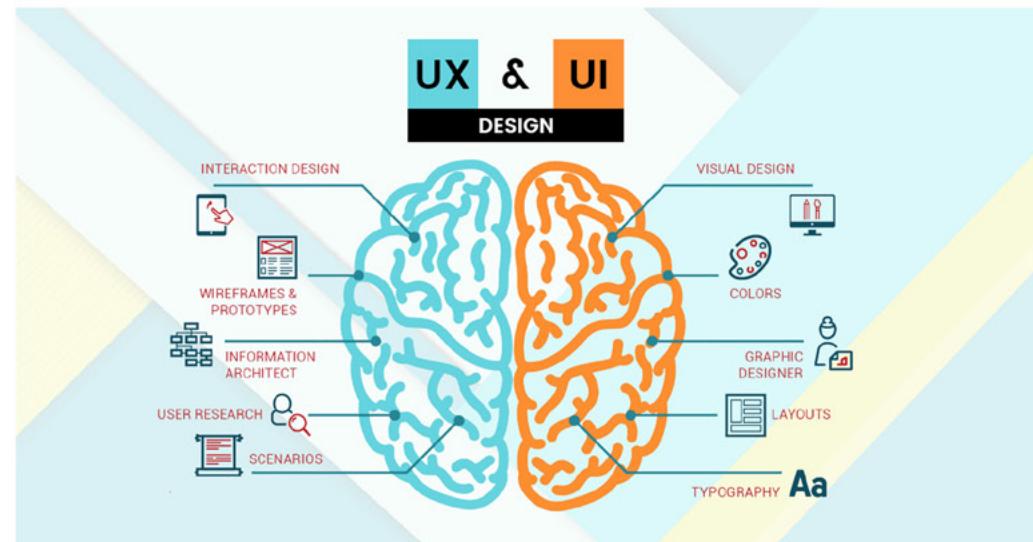
In general they are two separate entities, the UX is the functionality and the UI is the aesthetics.

This is UI:

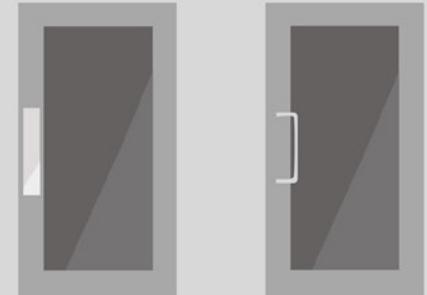
Navigation, subnavigation, menus, drop-downs, buttons, links, windows, rounded corners, shadowing, error messages, alerts, updates, checkboxes, password fields, search fields, text inputs, radio selections, text areas, hover states, selection states, pressed states, tooltips, banner ads, embedded videos, swipe animations, scrolling, clicking, iconography, colors, lists, slideshows, alt text, badges, notifications, gradients, pop-ups, carousels, OK/Cancel, etc. etc. etc.

This is UX:

People, happiness, solving problems, understanding needs, love, efficiency, entertainment, pleasure, delight, smiles, soul, warmth, personality, joy, satisfaction, gratification, elation, exhilaration, bliss, euphoria, convenience, enchantment, magic, productivity, effectiveness, etc. etc. etc.



Push or Pull?



Norman's Doors

UI

VS

UX

Cognitive Biases

Cognitive bias is a limitation in objective thinking that is caused by the tendency for the human brain to perceive information through a filter of personal experience and preferences.

<https://www.techtarget.com/searchenterpriseai/definition/cognitive-bias>

It is important to know different cognitive biases to make an informed decision while designing for UX and also to get different perspective by putting ourselves in the user's shoes.

A few random examples.

45



Representativeness heuristic

What is it?

People tend to judge the probability of an event by finding a 'comparable known' event and assuming that the probabilities will be similar.

When people rely on representativeness to make judgments, they are likely to judge wrongly because the fact that something is more representative does not actually make it more likely.

Example:

In business, if a customer meets a salesman from a certain company that is aggressive, the customer might assume that the company has an aggressive culture.

34



Overconfidence Effect

What is it?

We systematically overestimate our knowledge and our ability to predict.

Overconfidence measures the difference between what people really know and what they think they know.

It turns out the experts suffer even more from the overconfidence effect than laypeople do.

Studies have found that over 90% of US drivers rate themselves above average, 68% of professors consider themselves in the top 25 percent for teaching ability, and 84% of Frenchmen believe they are above-average lovers.

29



Licensing Effect

What is it?

When people allow themselves to indulge after doing something positive first.

Drinking a diet coke with a cheeseburger can lead one to subconsciously discount the negative attributes of the meal's high caloric and cholesterol content.

Going to the gym can lead us to ride the elevator to the second floor.

A study showed that people who took multivitamin pills were more prone to subsequently engage in unhealthy activities.

25



Herd Behavior

What is it?

The tendency for individuals to mimic the actions (rational or irrational) of a larger group. Individually, however, most people would not necessarily make the same choice.

Example:

In the late 1990s investors were investing huge amounts of money into Internet-related companies, even though most of them did not have structured business models. Their driving force was the reassurance they got from seeing so many others do the same.

01



Affect Heuristic

What is it?

A mental shortcut that allows people to make decisions quickly by bringing their emotional response into play. They make decisions according to their gut feeling.

Example:

If someone has harmed you, you quickly arrive at the conclusion that this person is cold and unfriendly. If fact, even if the person didn't harm you on purpose, you may still feel about him or her the same.

24



Hedonic Adaptation

What is it?

People quickly return to their original level of happiness, despite major positive or negative events or life changes.

When good things happen, we feel positive emotions but they don't usually last. The excitement of purchasing a new car or getting a promotion at work is temporary.

One study showed that despite initial euphoria, lottery winners were no happier than non-winners eighteen months later.

04



Bounded Rationality

What is it?

The idea that in decision-making, people are limited by the information they have, the cognitive limitations of their minds, and the finite time. As a result, they seek for a "good enough" decision and tend to make a satisficing (rather than maximizing or optimizing) choice.

Example:

During shopping when people buy something that they find acceptable, although that may not necessarily be their optimal choice.

03



Availability Heuristic

What is it?

A mental shortcut that relies on immediate examples that come to a given person's mind when evaluating a specific topic, concept, method or decision.

People tend to heavily weigh their judgments toward more recent information, making new opinions biased toward the latest news.

Example:

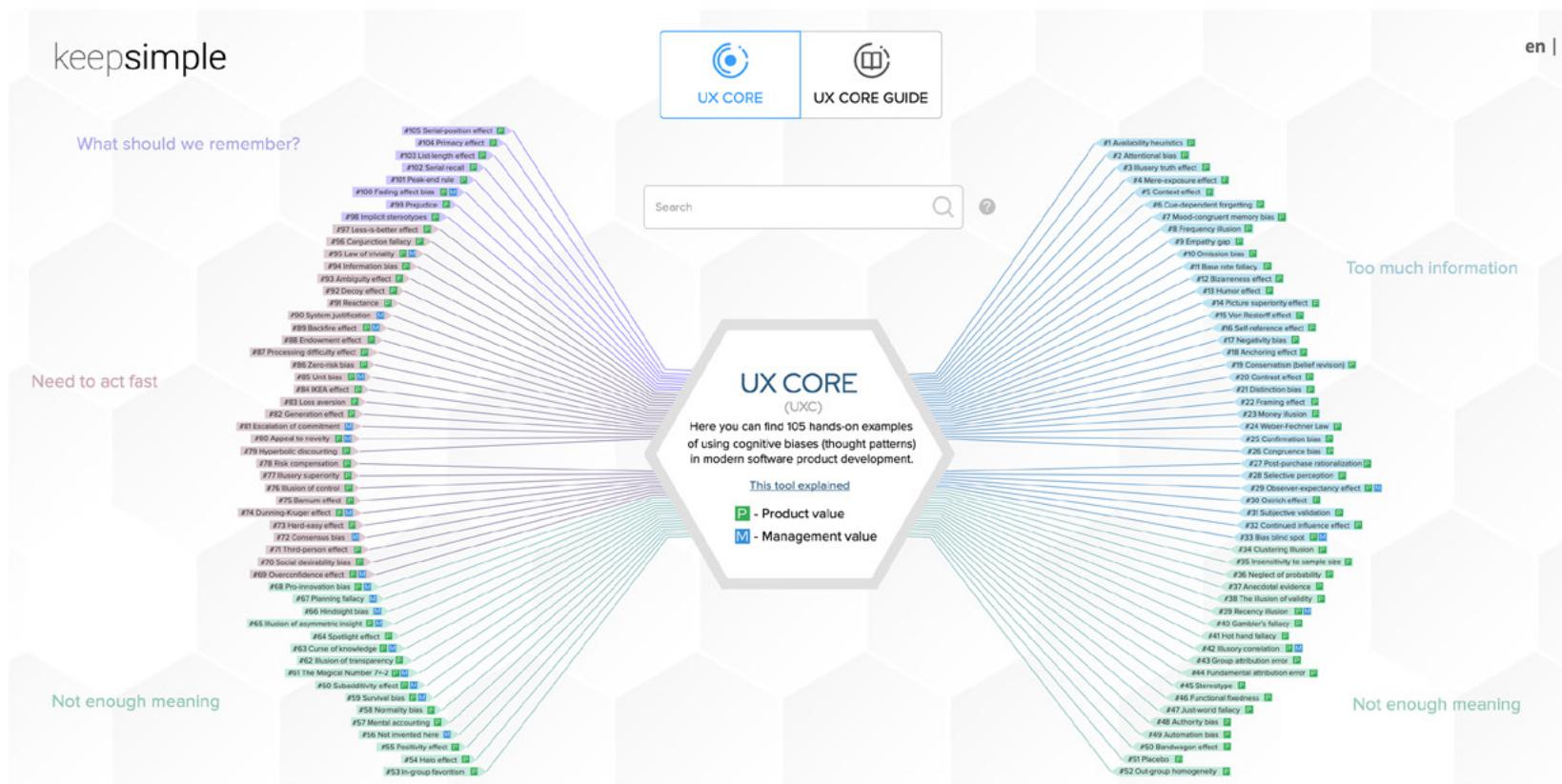
While high-profile police shootings might lead you to think that cops have a more dangerous job, statistics actually show that loggers are more likely to die on the job than cops.

Cognitive Biases

Cognitive bias hub (<https://keepsimple.io/uxcore/10>). I plan to look at this as I go through through the project. Hopefully this will give me a deeper understanding of cognitive biases and how they can be thought about when designing.

Above: Hub with 105 different examples.

Below: Example of one of the biases.



The detailed view of the "Omission bias" page shows the following structure:

- Description:** One of the cognitive biases manifested in people's tendency to underestimate the consequences of inaction compared to action with a similar result. Responsibility for action by people is perceived more than inaction.
- Usage:** Using it in software often goes against moral and ethical standards and continues to be actively used by a wide variety of companies. For example, knowing that inaction is underestimated, we can create an online magazine service with subscriptions in such a way that people would just be lazy to unsubscribe from the service.
- Mentioned in UX Core Guide questions:**
 - PreSignUp: #1. Our product/company has a reputation issue. How did this happen?
 - PostSignUp: #11. Our users accuse us of problems that have arisen because of their carelessness.
- How useful you found this?**: A scale from 1 (Not useful) to 10 (Very useful).
- Contact us**: Links to Podcast about UXCG, Using UXCG in Education, and UX Core Guide.

Gestalt Principles

Gestalt Principles are principles/laws of human perception that describe how humans group similar elements, recognize patterns and simplify complex images when we perceive objects. Designers use the principles to organize content on websites and other interfaces so it is aesthetically pleasing and easy to understand.

<https://www.interaction-design.org/literature/topics/gestalt-principles>

1. The law of proximity

This is a single group.



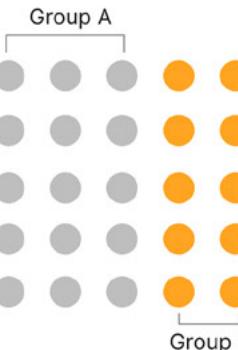
These are considered as separate groups



Name
James Stewart

Email
James@stewart.connect

2. The law of similarity



Email
James@stewart.connect

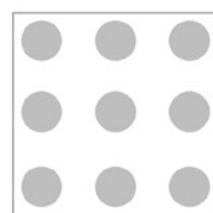
Password
.....

LOGIN

By using our software you agree our [terms](#) and [conditions](#). For any assistance please visit [helpdesk](#).

3. The law of common-region

Group A



Putting a different ending
World famous and oscar winning movie Shape of water loved by everyone, but what if this movie had different ending. Check out!

Learning to board
Skate-boarding a journey from becoming a beginner to the professional. A journey toward freedom of self expression.

Far Most Gone
Telling a journey from becoming a beginner to the professional. A journey toward freedom of self expression.

Hidden chambers of love
Big loved is the best experience you

4. The law of focal point



Name
James Stewart

Email
James@stewart.connect

Address
12 St, Jecon Flastin, LA

Password
.....

SIGN UP

Gestalt Principles

5. The law of continuity



In UX

Home	
Profile	
Shop >	
Contact	
Help	

Men >
Women >
Kids >

6. The law of closure

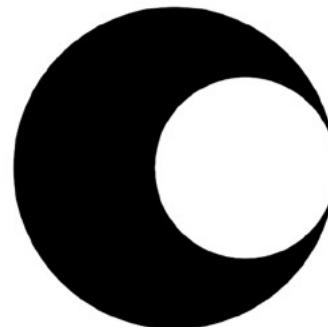


In design



LemonadeFilms

7. The law of figure-ground



In UX

Putting a different ending
World famous and oscar winning movie Shape of the water loved by everyone, but what if this movie had different ending. Check out:

Far Most Gone
Trekking a journey from becoming a beginner to the professional. A journey toward freedom of self expression.

You can focus on this message because of this law.

Hidden chambers of love
Being loved is the best experience you

Freecodecamp

Some of the Lessons I have completed on Freecodecamp.com as part of the Responsive Web Design course.

Each one of these Topics is essentially a chapter, the lesson count for each one is in the bottom right corner.

This course helped me to understand what html and css can do and taught me the basics. It was a great introduction to the topic.

Basic HTML and HTML5

HTML is a markup language that uses a special syntax or notation to describe the structure of a webpage to the browser. HTML elements usually have opening and closing tags that surround and give meaning to content. For example, different elements can describe text as a heading, paragraph, or list item.

In this course, you'll build a cat photo app to learn some of the most common HTML elements – the building blocks of any webpage.

▶ Expand course

⌚ 28/28

Basic CSS

CSS, or Cascading Style Sheets, tell the browser how to display the text and other content that you write in HTML. With CSS, you can control the color, font, size, spacing, and many other aspects of HTML elements.

Now that you've described the structure of your cat photo app, give it some style with CSS.

▶ Expand course

⌚ 44/44

Applied Visual Design

Visual design is a combination of typography, color theory, graphics, animation, page layout, and more to help deliver your unique message.

In this course, you'll learn how to apply these different elements of visual design to your webpages.

▶ Expand course

⌚ 52/52

Applied Accessibility

In web development, accessibility refers to web content and a UI (user interface) that can be understood, navigated, and interacted with by a broad audience. This includes people with visual, auditory, mobility, or cognitive disabilities.

In this course, you'll learn best practices for building webpages that are accessible to everyone.

▶ Expand course

⌚ 22/22

Responsive Web Design Principles

There are many devices that can access the web, and they come in all shapes and sizes. Responsive web design is the practice of designing flexible websites that can respond to different screen sizes, orientations, and resolutions.

In this course, you'll learn how to use CSS to make your webpages look good, no matter what device they're viewed on.

▶ Expand course

⌚ 4/4

CSS Flexbox

Flexbox is a powerful, well-supported layout method that was introduced with the latest version of CSS, CSS3. With flexbox, it's easy to center elements on the page and create dynamic user interfaces that shrink and expand automatically.

In this course, you'll learn the fundamentals of flexbox and dynamic layouts by building a Twitter card.

▶ Expand course

⌚ 17/17

CSS Grid

The CSS grid is a newer standard that makes it easy to build complex responsive layouts. It works by turning an HTML element into a grid, and lets you place child elements anywhere within.

In this course, you'll learn the fundamentals of CSS grid by building different complex layouts, including a blog.

▶ Expand course

⌚ 22/22

Responsive Web Design Projects

Time to put your newly learnt skills to work. By working on these projects, you will get a chance to apply all of the skills, principles, and concepts you have learned so far: HTML, CSS, Visual Design, Accessibility, and more.

Complete the five web programming projects below to earn your Responsive Web Design certification.

Build a Tribute Page

⌚

Build a Survey Form

⌚

Build a Product Landing Page

⌚

Build a Technical Documentation Page

⌚

Build a Personal Portfolio Webpage

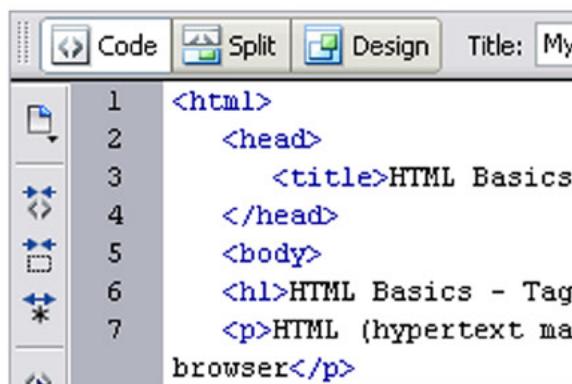
⌚

Tags, Attributes and Setting up Document with Boilerplates

Recording my learning of tags, attributes and how to start an HTML document using boilerplate code that can be repeated over multiple docs.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
</body>
</html>
```

This is an example of HTML5 boilerplate text. Every html doc should start with `<!DOCTYPE html>` and this is usually followed by declaring the language etc. title (`<title>`) and body (`<body>`) tags.



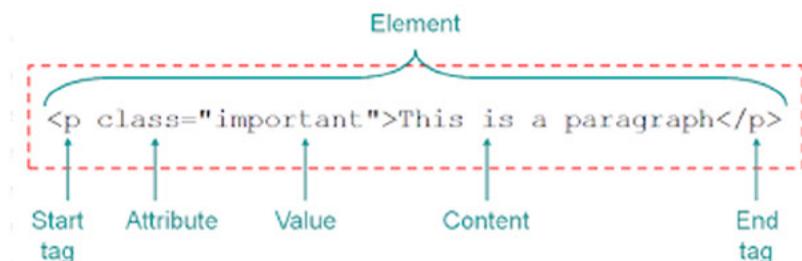
```
Code Split Design Title: My
1 <html>
2   <head>
3     <title>HTML Basics</title>
4   </head>
5   <body>
6     <h1>HTML Basics - Tag</h1>
7     <p>HTML (hypertext markup language)</p>
```

An HTML tag is a piece of markup language used to indicate the beginning and end of an HTML element in an HTML document.

Examples:

`<p> <title> <header> <h1> <a> <body> <div> `

Most of these tags need closing with a / included e.g.
`</body> </title> </p>` etc.



An HTML attribute is a piece of markup language used to adjust the behavior or display of an HTML element. For example, attributes can be used to change the color, size, or functionality of HTML elements.

<https://www.digitalocean.com/community/tutorials/what-is-an-html-attribute>

Examples:

`src="" ; class="" ; id="" ;`

How to animate using CSS

- **@keyframes**
- **animation-name**
- **animation-duration**
- **animation-delay**
- **animation-iteration-count**
- **animation-direction**
- **animation-timing-function**
- **animation-fill-mode**
- **animation**

Looking at animation using css. I want to try and do a mini project using this.

To use CSS animation, you must first specify some keyframes for the animation.

Keyframes hold what styles the element will have at certain times.

https://www.w3schools.com/css/css3_animations.asp

Some examples of how to write animations, and the different things you can do with them.

```
/* The animation code */
@keyframes example {
    0% {background-color:red; left:0px; top:0px;}
    25% {background-color:yellow; left:200px; top:0px;}
    50% {background-color:blue; left:200px; top:200px;}
    75% {background-color:green; left:0px; top:200px;}
    100% {background-color:red; left:0px; top:0px;}
}
```

```
#div1 {animation-timing-function: linear;}
#div2 {animation-timing-function: ease;}
#div3 {animation-timing-function: ease-in;}
#div4 {animation-timing-function: ease-out;}
#div5 {animation-timing-function: ease-in-out;}
```

```
/* The animation code */
@keyframes example {
    from {background-color: red;}
    to {background-color: yellow;}
}

/* The element to apply the animation to */
div {
    width: 100px;
    height: 100px;
    background-color: red;
    animation-name: example;
    animation-duration: 4s;
}

    animation-name: example;
    animation-duration: 4s;
    animation-iteration-count: 3;
```

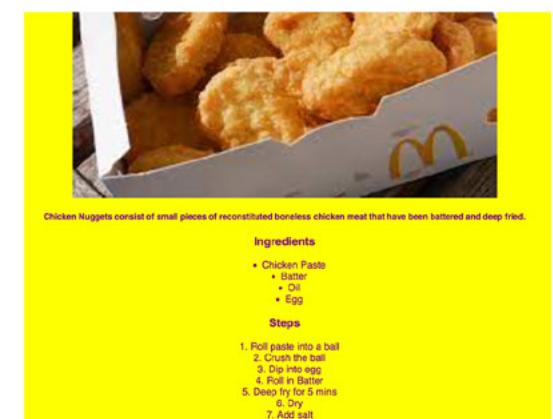
Recipe Website

Made a simple website. Used links from 1 main page to three separate sub-pages, each with a separate recipe. This was the first time I added background colours etc. with CSS. I used it to re-size images, text, and central align items on the page (didn't always work), need to work out what went wrong with burger on first page.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <style>
      .body {
        background-color: #yellow;
      }
      img {
        width: 60%;
        left: 0px;
        right: 0px;
        text-align: center;
      }
    </style>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link rel="stylesheet" href="menu.css">
  </head>
  <body class="body">
    <div class="front">
      <h1> Odin Recipes </h1>
      <br>
      <br>
    </div>
    <div class="seven">
      <a href="recipes/BigMac.html"> Burger </a>
      <br>
      <a href="recipes/nuggets.html">Chicken Nuggets</a>
      <br>
      <br>
      <a href="recipes/fries.html">Fries</a>
      <br>
      <br>
      <br>
      <br>
      <br>
    </div>
    <img src "./images/burger.jpg">
  </body>
</html>
```

```
<head>
  <h1 class="front"> Fries </h1>
  <style>
    .bm {
      font-family: Helvetica, Arial, sans-serif;
      font-size: 40px;
      color: #purple;
      background-color: #yellow;
      text-align: center;
    }
    ul {
      text-align: center;
      list-style-position: inside;
    }
    ol {
      text-align: center;
      list-style-position: inside;
    }
    img {
      width: 60%;
      left: 0px;
      right: 0px;
    }
    h1 {
      color: #red;
    }
  </style>
</head>
<body class="bm">
<img src "../images/fries.jpg">
<h3> French fries, chips, finger chips, French-fried potato </h3>
<h3><strong>Ingredients</strong></h3>
<ul>
  <li>Potatoes</li>
  <li>Salt</li>
  <li>Oil</li>
</ul>
<h3><strong>Steps</strong></h3>
<ol>
  <li> Cut potatoes into long finger shapes</li>
  <li> Deep fry until golden</li>
  <li> Add salt</li>
</ol>
</body>
```

```
<head>
  <h1 class="front"> Burger </h1>
  <style>
    .bm {
      font-family: Helvetica, Arial, sans-serif;
      font-size: 40px;
      color: #red;
      background-color: #yellow;
      text-align: center;
    }
    ul {
      text-align: center;
      list-style-position: inside;
    }
    ol {
      text-align: center;
      list-style-position: inside;
    }
    img {
      width: 60%;
      left: 0px;
      right: 0px;
    }
    h1 {
      color: #red;
    }
  </style>
</head>
<body class="bm">
<img src "../images/bigmac.jpg">
<h3> The Burger consists of two 1.6 oz (45.4 g) beef </h3>
<h3><strong>Ingredients</strong></h3>
<ul>
  <li>Buns</li>
  <li>Burger Patties</li>
  <li>Lettuce</li>
  <li>Seeds</li>
  <li>Mustard</li>
  <li>Pickle</li>
  <li>Onions</li>
  <li>American Cheese</li>
</ul>
<h3><strong>Steps</strong></h3>
<ol>
  <li> Add sauce to the buns</li>
  <li> Put in patty meat</li>
  <li> Add cheese</li>
  <li> Add lettuce, tomatoes</li>
  <li> Add bun top</li>
  <li> Add patty meat</li>
  <li> More onions, lettuce and tomatoes</li>
  <li> More Sauce</li>
  <li> Final bun</li>
</ol>
</body>
```



```
.front {
  font-family: Helvetica, Arial, sans-serif;
  font-size: 70px;
  color: #red;
  background-color: #yellow;
  text-align: center;
}

.seven {
  font-family: Helvetica, Arial, sans-serif;
  font-size: 40px;
  color: #purple;
  background-color: #yellow;
  text-align: center;
}
```

This mini project gave me a deeper insight into how I could bring my designs to life through code + link separate pages together to make a working website.

Good UX

Landing page design
I like.

The landing page features a large, semi-transparent image of a young man wearing glasses and a red Puma hoodie. Overlaid on this image are several interactive elements: a white 'N' logo with a menu icon, a 'MENU' button, and social media links for YouTube, LinkedIn, Instagram, Facebook, and Twitter. A 'LET'S CHAT' button is located in the top right corner. In the center, there's a 'CLIENT' section with the word 'PUMA'. To the right, a large white title reads 'PUMA All-Star Group'. Below the title is a descriptive paragraph: 'To build buzz around their youth soccer program, PUMA sent us to the heart of the Arsenal FC training program in London.' At the bottom right is a 'VIEW CASE STUDY' button. On the far right, there are scroll navigation arrows. At the bottom left, a call-to-action says '↓ Scroll through our recent projects'.

Fellow Nurture

CLIENT

PUMA

PUMA All-Star Group

To build buzz around their youth soccer program, PUMA sent us to the heart of the Arsenal FC training program in London.

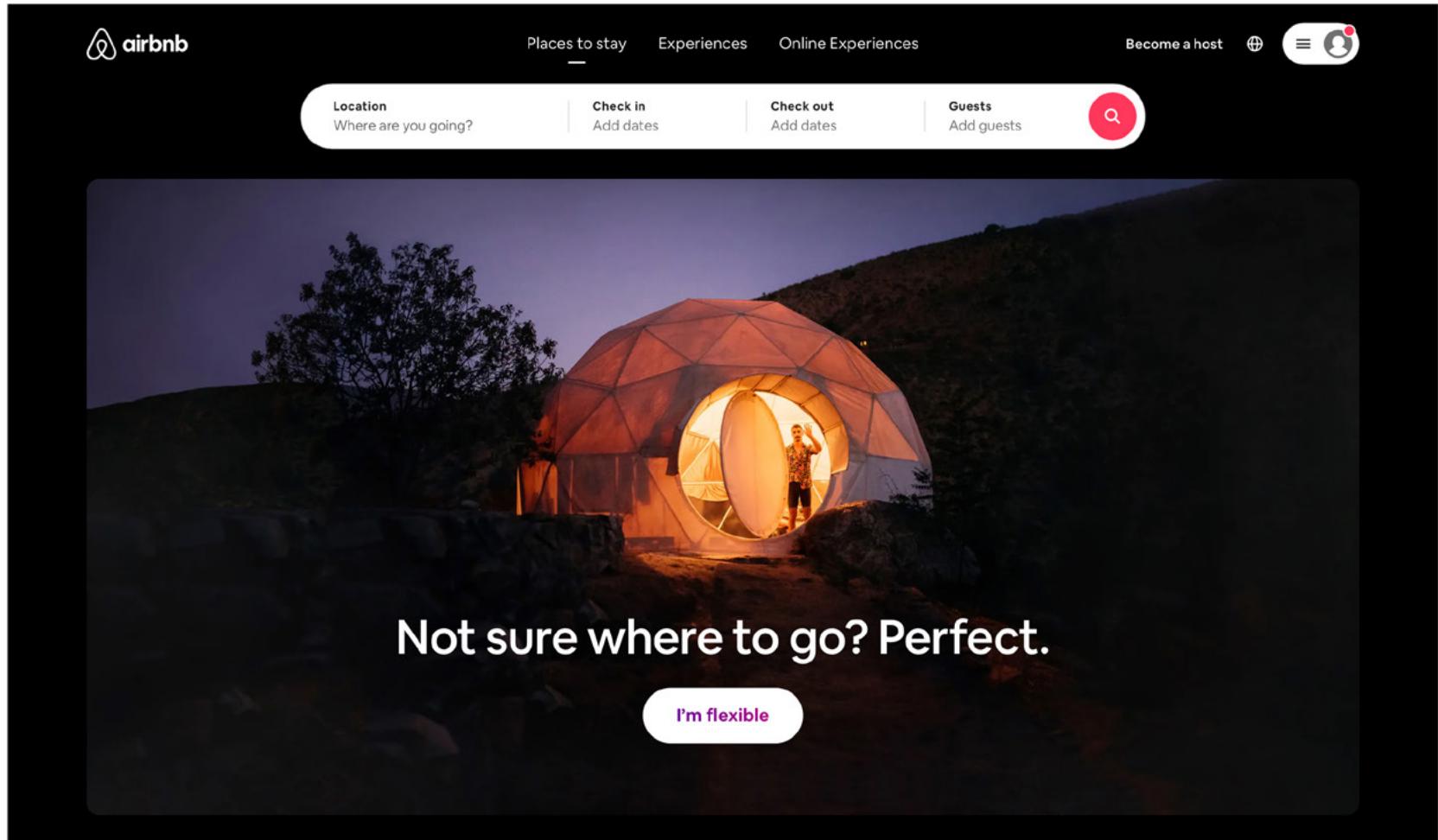
VIEW CASE STUDY

↓ Scroll through our recent projects

Nurture shoots. Everpost edits. Copyright Nurture Digital 2020 hello@nurturedigital.com

Good UX

Landing page design
I like.



Inspiration for your next trip

Brighton

47 miles away

Hastings

54 miles away

Plymouth

192 miles away

Margate

65 miles away

Using Git and Github

Learnt how to use git to stage, commit and push my work onto the website github. This will be helpful to keep track of my code, essentially saving it along the way. I did this using commands on the terminal and then with Visual Studio Code .

```
[Leos-MBP-2:odin-recipes leoroberts21$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    menu.css
    odin-recipes.txt

nothing added to commit but untracked files present (use "git add" to track)
Leos-MBP-2:odin-recipes leoroberts21$ git add odin-recipes.txt
Leos-MBP-2:odin-recipes leoroberts21$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  odin-recipes.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    menu.css

Leos-MBP-2:odin-recipes leoroberts21$ git add menu.css
Leos-MBP-2:odin-recipes leoroberts21$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  menu.css
    new file:  odin-recipes.txt

Leos-MBP-2:odin-recipes leoroberts21$ git commit -m "add odin-recipes.txt"
[main 6b6181d] add odin-recipes.txt
| 2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 menu.css
create mode 100644 odin-recipes.txt
Leos-MBP-2:odin-recipes leoroberts21$ git commit -m "add menu.css"
On branch main
nothing to commit, working tree clean
Leos-MBP-2:odin-recipes leoroberts21$ git log
commit 6b6181d40b9d5873c1f9a296b6f7a6f46311168c (HEAD -> main)
Author: Leo Roberts <leo.roberts21@gmail.com>
Date:  Sun Feb 27 00:37:47 2022 +0000

  add odin-recipes.txt

commit 6b6181d40b9d5873c1f9a296b6f7a6f46311168c (HEAD -> main)
Author: Leo Roberts <leo.roberts21@gmail.com>
Date:  Sun Feb 27 00:37:47 2022 +0000

  add odin-recipes.txt

commit 4bdff9f4edc4a258df04c791539d40f1f59810fe
Author: Leo Roberts <leo.roberts21@gmail.com>
Date:  Wed Feb 23 21:01:25 2022 +0000

  add odin-recipes.txt
```

Github repository

The screenshot shows a GitHub repository named 'odin-recipes' owned by 'Leoroberts21'. The repository has 5 commits, all made 6 days ago. The README.md file describes the project as 'a website about multiple recipes, in an attempt to showcase my HTML skills'. The repository has 0 stars, 1 watching, and 0 forks. It also lists 'About', 'Releases', 'Packages', and 'Languages' sections.

The screenshot shows the GitHub Desktop application interface. It displays a list of staged changes for the 'odin-recipes' repository. There are merge conflicts with 'BigMac.html' and 'index.html'. A 'Publish Branch' button is visible at the bottom right.

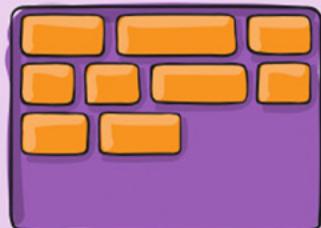
Pushing changes
visual Studio Code.

Understanding Flex-box

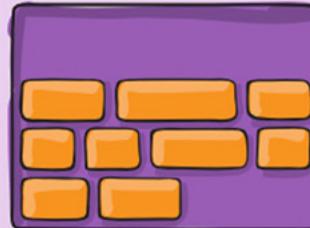
The flex layout allows responsive elements within a container to be automatically arranged depending upon screen size.

align-content

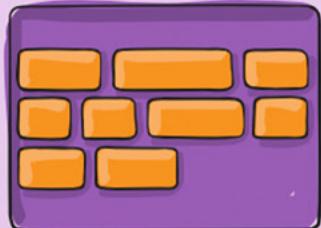
flex-start



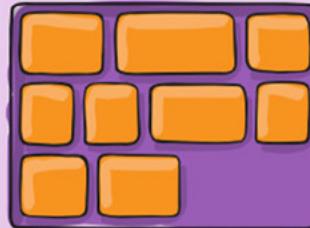
flex-end



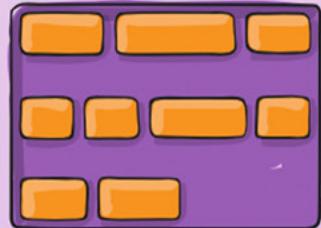
center



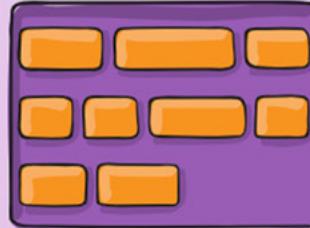
stretch



space-between



space-around

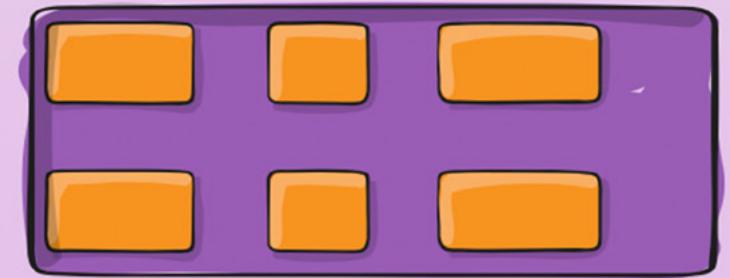


gap, row-gap, column-gap

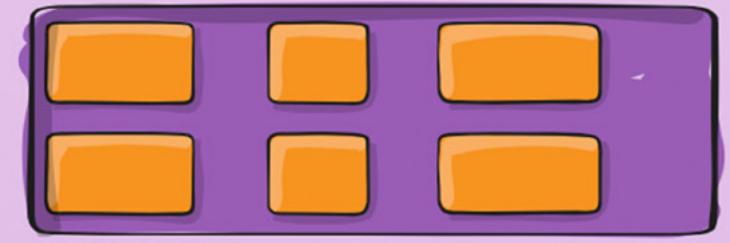
gap: 10px



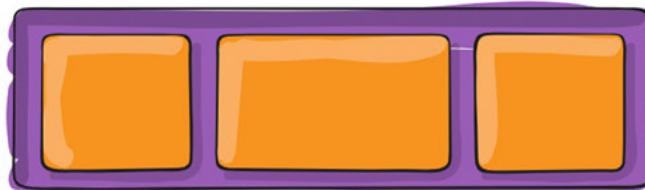
gap: 30px



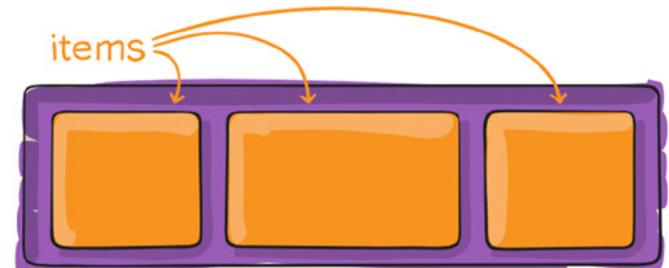
gap: 10px 30px



container ↗



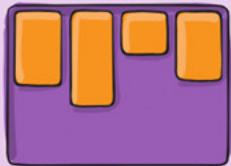
items ↗



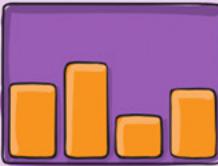
Understanding Flex-box

align-items

flex-start



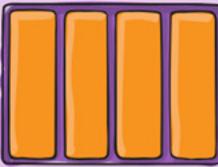
flex-end



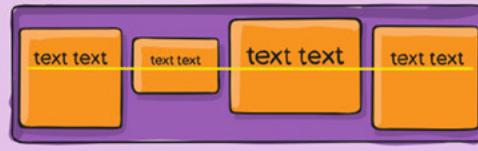
center



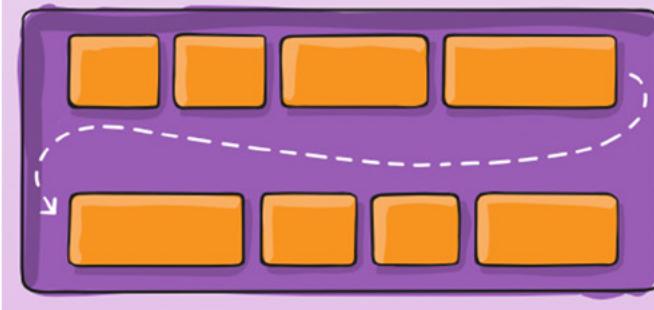
stretch



baseline



flex-wrap

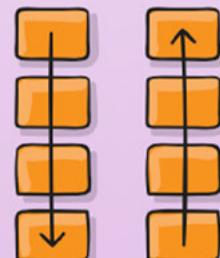


flex-grow

1 1 1

1 2 1

flex-direction



justify-content

flex-start



flex-end



center



space-between



space-around



space-evenly



CSS Flex Box exercises

Two challenges trying to imporve my skills using flexbox in CSS.

Left : Challenge 1

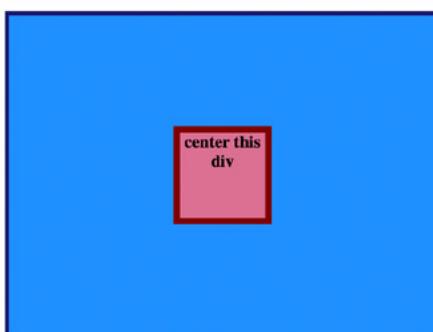
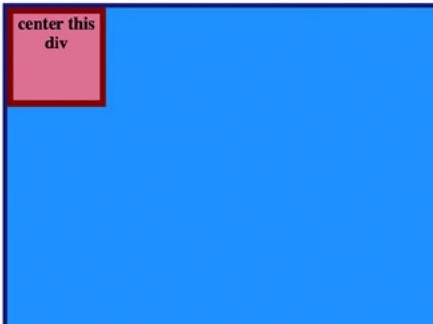
Move the box to the middle of the container.

Right: Challenge 2

Re-arrange the nav bar so it's readable, spaced out etc. and is responsive. This took me a while to figure.

```
<style>
    .container {
        background: #dodgerblue;
        border: 4px solid #midnightblue;
        width: 400px;
        height: 300px;
        display: flex;
        align-items: center;
        justify-content: center;
    }

    .box {
        background: #palevioletred;
        font-weight: bold;
        text-align: center;
        border: 6px solid #maroon;
        width: 80px;
        height: 80px;
    }
</style>
</head>
<body>
    <div class="container">
        <div class="box">center this div</div>
    </div>
</body>
</html>
```



ONE
TWO
THREE

LOGO

FOUR
FIVE
SIX

ONE TWO THREE

LOGO

FOUR FIVE SIX

```
<style>
    .header {
        font-family: monospace;
        background: #papayawhip;
        display: flex;
        justify-content: space-between;
        align-items:center;
        height: 100px;
        padding: 15px;
    }

```

```
.logo {
    font-size: 48px;
    font-weight: 900;
    color: #tomato;
    background: #white;
    padding: 4px 32px;
    display: flex;
    justify-content: center;
    align-items:center;
    height:fit-content;
}
```

```
ul {
    /* this removes the dots on the list items*/
    list-style-type: none;
    display: flex;
    margin: 0;
    padding: 0;
    gap: 8px;
}
```

```
a {
    font-size: 22px;
    background: #white;
    padding: 8px;
    /* this removes the line under the links */
    text-decoration: none;
}
```

```
.flex-c1 {
    display: flex;
    justify-content: space-between;
}

.flex-c2 {
    display:flex;
}
```

```
</style>

</head>
<body>
    <div class="header">
        <div class="left-links">
            <ul class="flex-c1">
                <li><a href="#">ONE</a></li>
                <br>
                <li><a href="#">TWO</a></li>
                <br>
                <li><a href="#">THREE</a></li>
            </ul>
        </div>
        <div class="logo">LOGO</div>
        <div class="right-links">
            <ul class="flex-c2">
                <li><a href="#">FOUR</a></li>
                <br>
                <li><a href="#">FIVE</a></li>
                <br>
                <li><a href="#">SIX</a></li>
            </ul>
        </div>
    </div>
</body>
</html>
```

CSS Flex Box exercises

Two more challenges trying to imporve my skills using flexbox in CSS.

Top: Challenge 3

Arrange the elements(left) so they look like a egible nav-bar.(right)

Right: Challenge 4

Arrange information in the center of the page.

LOGO

link-one
link-two
link-three

1 new notification



LOGO

link-one link-two link-three

```
body {  
  margin: 0;  
  background: #eee;  
  font-family: Besley, serif;  
}
```

```
.header {  
  background: #white;  
  border-bottom: 1px solid #ddd;  
  box-shadow: 0 0 8px rgba(0,0,0,.1);  
  display: flex;  
  padding: 10px;  
  justify-content: space-between;  
}
```

```
.profile-image {  
  background: #rebeccapurple;  
  box-shadow: inset 2px 2px 4px rgba(0,0,0,.5);  
  border-radius: 50%;  
  width: 48px;  
  height: 48px;  
}
```

```
.logo {  
  color: #rebeccapurple;  
  font-size: 32px;  
  font-weight: 900;  
  font-style: italic;  
  margin-right: 10px;  
}
```

```
button {  
  border: none;  
  border-radius: 8px;  
  background: #rebeccapurple;  
  color: #white;  
  padding: 8px 24px;  
}
```

```
a {  
  /* this removes the line under our links */  
  text-decoration: none;  
  color: #rebeccapurple;  
}
```

```
ul {  
  list-style-type: none;  
  display: flex;  
  margin: 0;  
  padding: 0;  
  gap: 16px;  
}
```

```
.left, .right {  
  display: flex;  
  align-items: center;  
  gap: 16px;  
}
```

```
</style>
```

```
<body>  
<div class="header">  
<div class="left">  
<div class="logo">  
  LOGO  
</div>  
<ul class="links">  
  <li><a href="google.com">link-one</a></li>  
  <li><a href="google.com">link-two</a></li>  
  <li><a href="google.com">link-three</a></li>  
</ul>  
</div>  
<div class="right">  
  <button class="notifications">  
    1 new notification  
  </button>  
  <div class="profile-image"></div>  
</div>  
</div>  
</body>  
</html>
```

Information!



This is a type of plant. We love this one.



This is another type of plant. Isn't it nice



We have so many plants. Yay plants.



I'm running out of things to say about plants.

```
<style>  
body {  
  font-family: 'Courier New', Courier, monospace;  
}  
  
img {  
  width: 100px;  
  height: 100px;  
}  
  
.title {  
  font-size: 36px;  
  font-weight: 900;  
}  
  
/* do not edit this footer */  
.footer {  
  position: fixed;  
  bottom: 0;  
  left: 0;  
  right: 0;  
  height: 52px;  
  display: flex;  
  align-items: center;  
  justify-content: center;  
  background: #eee;  
}
```

```
body {  
  text-align: center;  
}
```

```
.title {  
  margin-bottom: 32px;  
}
```

```
.container {  
  display: flex;  
  justify-content: center;  
  gap: 52px;  
}
```

```
.info {  
  max-width: 200px;  
}
```

```
</style>  
</head>  
<body>  
  <div class="title">Information!</div>  
  
  <div class="container">  
    <div class="info">  
        
      <div class="text">This is a type of plant. We love this one.</div>  
    </div>  
    <div class="info">  
        
      <div class="text">This is another type of plant. Isn't it nice</div>  
    </div>  
    <div class="info">  
        
      <div class="text">We have so many plants. Yay plants.</div>  
    </div>  
    <div class="info">  
        
      <div class="text">I'm running out of things to say about plants.</div>  
    </div>  
  </div>  
</body>
```

Information!

barberry

This is a type of plant. We love this one.

chilli

This is another type of plant. Isn't it nice

pepper

We have so many plants. Yay plants.

saffron

I'm running out of things to say about plants.

What I want from this Project

What do I want to do for this project?

- Show coding development.
- HTML
- CSS
- JAVASCRIPT
- Improve skills with Figma.
- Learn Webflow.
- Design a final outcome that documents this progress. (website, zine etc)

Projects I could do to achieve this:

- Recipe website
- Code a Landing page
- Design and mockup a website using Figma.
- Use Webflow to bring website to life.
- Re-create a pre-existing website using code.
- Javascript projects.
- Design final outcome

CSS Flex Box exercises

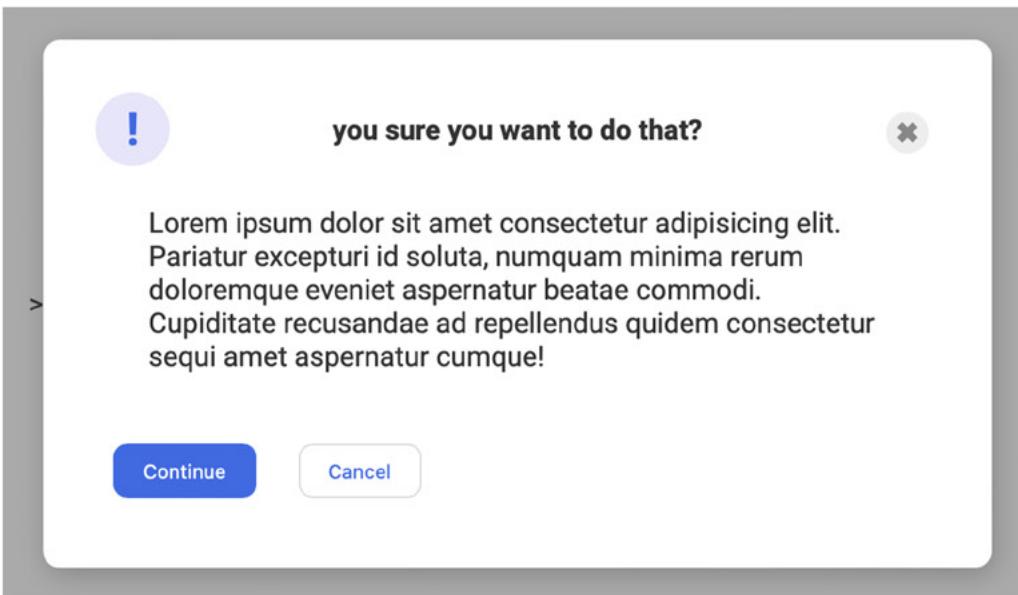
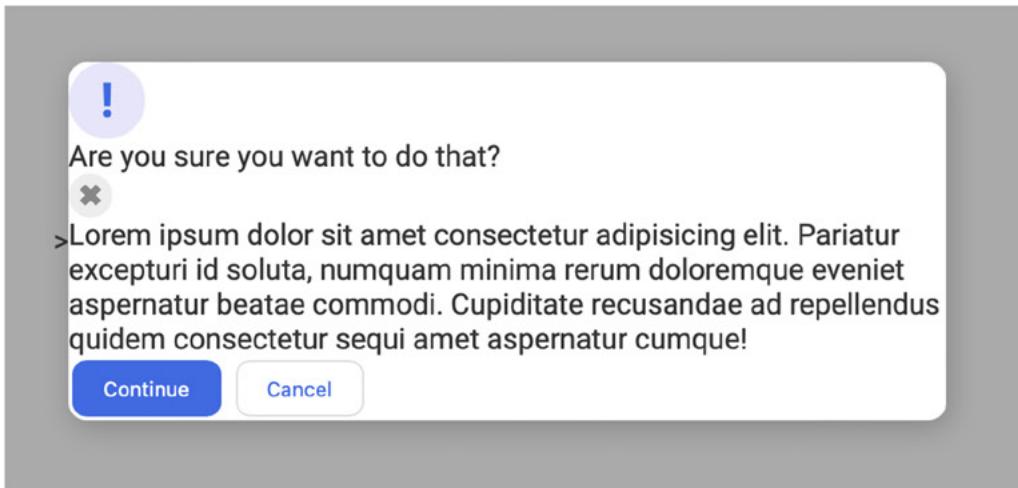
Two challenges trying to imporve my skills using flexbox in CSS.

Left : Challenge 1

Move the box to the middle of the container.

Right: Challenge 2

Re-arrange the nav bar so it's readable, spaced out etc. and is responsive. This took me a while to figure.



```
body {  
    font-family: Roboto, sans-serif;  
    margin: 0;  
    background: #aaa;  
    color: #333;  
    /* I'll give you this one for free lol */  
    display: flex;  
    align-items: center;  
    justify-content: center;  
}  
.modal {  
    background: white;  
    width: 480px;  
    border-radius: 10px;  
    box-shadow: 0px 16px 0px 16px rgba(0,0,0,.2);  
    padding: 30px;  
}  
.icon {  
    color: royalblue;  
    font-size: 26px;  
    font-weight: 700;  
    background: lavender;  
    width: 42px;  
    height: 42px;  
    border-radius: 50%;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
}  
button.continue {  
    background: royalblue;  
    border: 1px solid royalblue;  
    color: white;  
    width: 150px;  
    margin: 10px;  
}  
button.cancel {  
    background: white;  
    border: 1px solid #ddd;  
    color: royalblue;  
    width: 150px;  
    margin: 10px;  
}  
.header {  
    justify-content: center;  
    margin-top: -20px;  
    margin-left: 450px;  
}  
button {  
    padding: 8px 16px;  
    border-radius: 8px;  
}  
.text {  
    margin: 30px;  
}
```

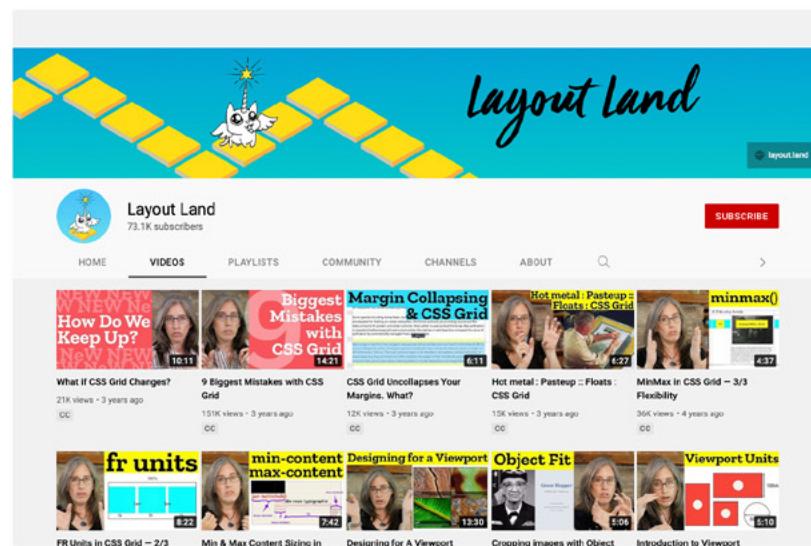
Jen Simmons

Graphic designer, web developer, educator and speaker known for her expertise in web standards, particularly HTML and CSS. She is a member of the CSS Working Group created by the World Wide Web Consortium and has been prominent in the deployment of CSS grid layout.

https://en.wikipedia.org/wiki/Jen_Simmons

It's Interesting for me to look at someones work who is involved in both Graphic design and web development.

She has a really useful youtube channel that mostly teaches layouts with css and web design.



THE EXPERIMENTAL LAYOUT LAB OF JEN SIMMONS
LAYOUT LAND VIDEOS
JENSIMMONS.COM

THE CONFERENCE TALKS

- 1. HOW CAN YOU GETTING OUT OF OUR BUBBLE
- 2. REVOLUTIONIZE YOUR PAGE: REDESIGNING YOUR SITE WITH CSS GRID
- 3. EVERYTHING YOU NEED TO KNOW ABOUT CSS GRID AND CHANGED DESIGNING INTERACTIVE LAYOUTS

Follow @jensimmons on Twitter for more info & updates.

Hi. I'm Jen Simmons.
I'm a Designer and Developer Advocate at Mozilla and a member of the CSS Working Group. I'm especially interested in the coming revolution in **GRAPHIC DESIGN ON THE WEB**.
I SPEAK AT CONFERENCES and MAKE VIDEOS. From 2011-2015, I was the host and executive producer of **THE WEB AHEAD**.

Her website

Making my first landing page

I want to code a landing page that looks organized and somewhat like a real website.

This is the initial HTML I coded for it, and how it currently looks.

I'm getting these challenges from a course called 'The Odin Project'.

Title

- Header 1
 - Header 2
 - Header 3

This website is awesome

“Lorem ipsum dolor sit amet consectetur adipisicing elit. Iste, reprehenderit!

Click Me!

Some random information.

Information



● ● ●



* * *



* * *

quotequotequotequotequotequotequotequotequote

-quote person

Call to action! It's time!

Sign up for our product by clicking that button right over there!

[Sign up](#)

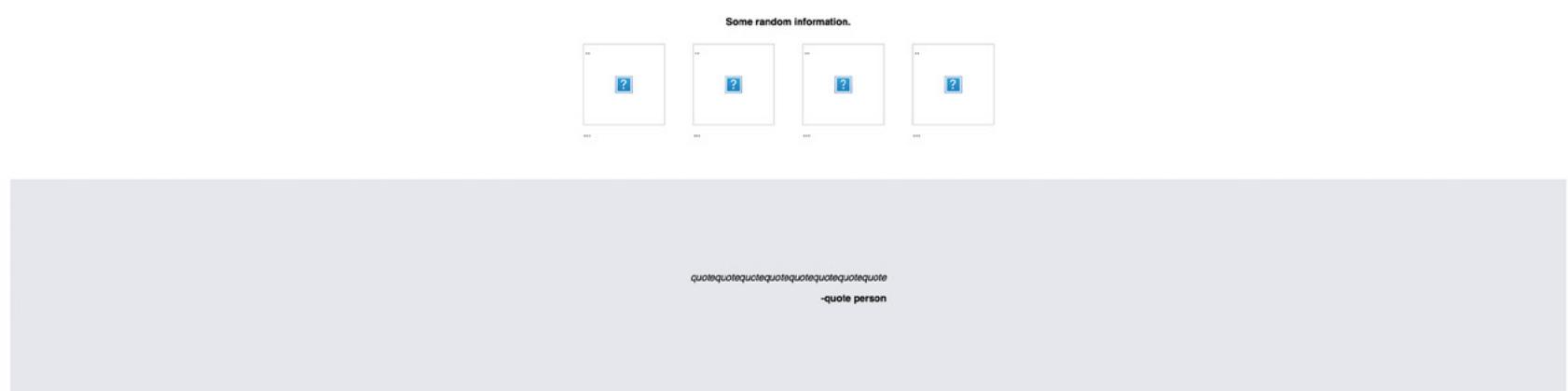
Copyright The Odin Project 2021

Making my first landing page

Website layed out. Now I just need to do the button styling and give the website a theme.

The site is zoomed out here so I can show all elements.

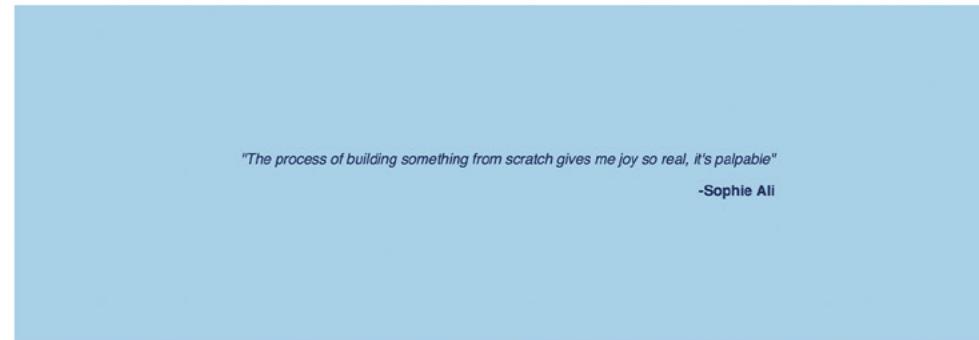
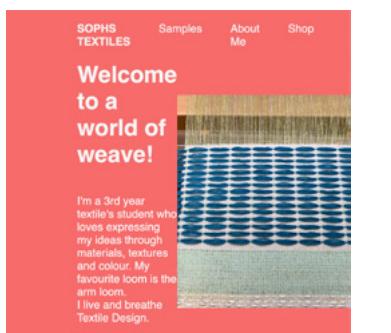
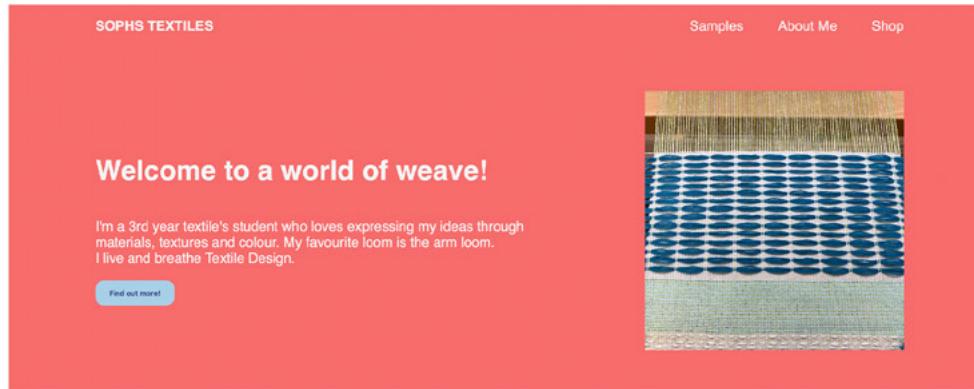
Once it's done I will show it's responsivness and how it would look at normal size.



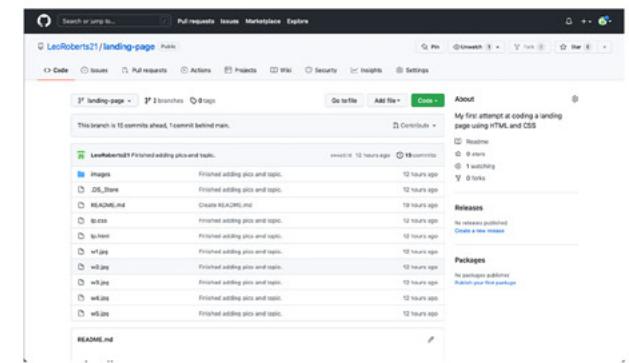
Making my first landing page

Final landing page.

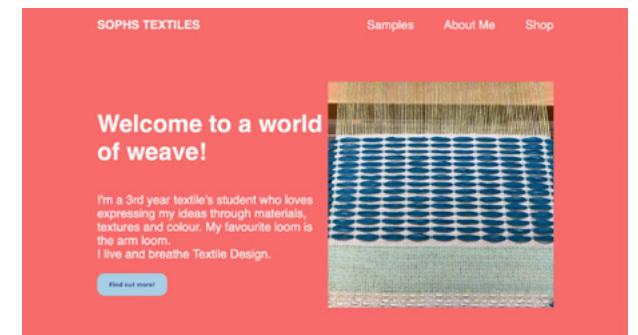
Happy with my progress using flex-box, and feel confident going forward with other projects.



Need to think more about
responsivness for my
next project.



Created a git hub repository to push my work to. This saves my work at different stages + stores all my coding projects.



HTML + CSS for Landing Page

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Landing page</title>
    <link rel="stylesheet" href="lp.css">
  </head>

  <body>
    <div class="header">
      <header class="title"><strong>SOPHS TEXTILES</strong></header>
      <ul class="tabs">
        <li>Samples</li>
        <br>
        <li>About Me</li>
        <br>
        <li>Shop</li>
      </ul>
    </div>

    <div class="front">
      <div class="left">
        <h1 class="h1"><strong>Welcome to a world of weave!</strong></h1>
        <p>I'm a 3rd year textile's student who loves expressing my ideas through<br>
        <button class="b" type="button"><strong>Find out more!</strong></button></p>
      </div>

      <div class="right">
        
      </div>
    </div>

    <div class="h2"><strong>SAMPLES</strong></div>
    <div class="container">

      <div class="info">
        
        <div class="techs">Twill</div>
      </div>

      <div class="info">
        
        <div class="techs">Basket</div>
      </div>
```

```
<div class="info">
  
  <div class="techs">Honeycomb</div>
</div>

<div class="info">
  
  <div class="techs">Plain</div>
</div>

<div class="quote-box">
  <div class="text">
    <p><i>The process of building something from scratch gives me great satisfaction.</i></p>
    <p><strong>--Sophie Ali</strong></p>
  </div>
</div>

<div class="c2">
  <div class="cta">
    <p><strong>Sign up for my newsletter!</strong></p>
    <p>Exclusive offers and tips lay ahead!</p>
  </div>

  <div class="b2">
    <button type="button"> Sign up</button>
  </div>
</div>

<div class="b">
  
</div>
```

```
.header {
  display: flex;
  border-left: 100px;
  border-right: 100px;
  justify-content: space-between;
  background-color: #f7fc6c;
}

.title {
  display: flex;
  width: auto;
  justify-content: flex-start;
  align-items: center;
  font-family: Helvetica, Arial, sans-serif;
  color: #whitesmoke;
  margin-left: 100px;
}

.tabs {
  display: flex;
  list-style: none;
  gap: 20px;
  font-family: Helvetica, Arial, sans-serif;
  color: #whitesmoke;
  margin-right: 100px;
}

.front {
  display: flex;
  height: 400px;
  border-left: 100px;
  border-right: 100px;
  justify-content: space-between;
  background-color: #f7fc6c;
}

.left {
  display: flex;
  flex-direction: column;
  width: 500px;
  justify-content: center;
  align-items: flex-start;
  font-family: Helvetica, Arial, sans-serif;
  color: #whitesmoke;
  margin-left: 100px;
}

.info {
  background-color: #e6eaf2;
  border: none;
  color: #337788;
  padding: 10px 15px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 8px;
  border-radius: 20px;
}

.b {
  background-color: #e6eaf2;
  border: none;
  color: #24305e;
  padding: 10px 15px;
  text-align: center;
  justify-content: center;
  text-decoration: none;
  display: inline-block;
  font-size: 8px;
  border-radius: 10px;
}

.quote {
  height: 400px;
  display: flex;
  background-color: #add8e6;
  justify-content: center;
}

.quote-box {
  height: 400px;
  display: flex;
  background-color: #add8e6;
  justify-content: center;
}

.h2 {
  text-align: center;
  font-family: Helvetica, Arial, sans-serif;
  margin: 30px;
  color: #24305e;
  margin-top: 40px;
  font-size: 12px;
}

.h2 {
  text-align: center;
  font-family: Helvetica, Arial, sans-serif;
  margin: 30px;
  color: #24305e;
  margin-top: 40px;
  font-size: 12px;
}

.Quote {
  display: flex;
  justify-content: space-between;
  margin-left: 150px;
  margin-right: 150px;
  background-color: #24305e;
  border-radius: 20px;
  width: max-content;
}
```

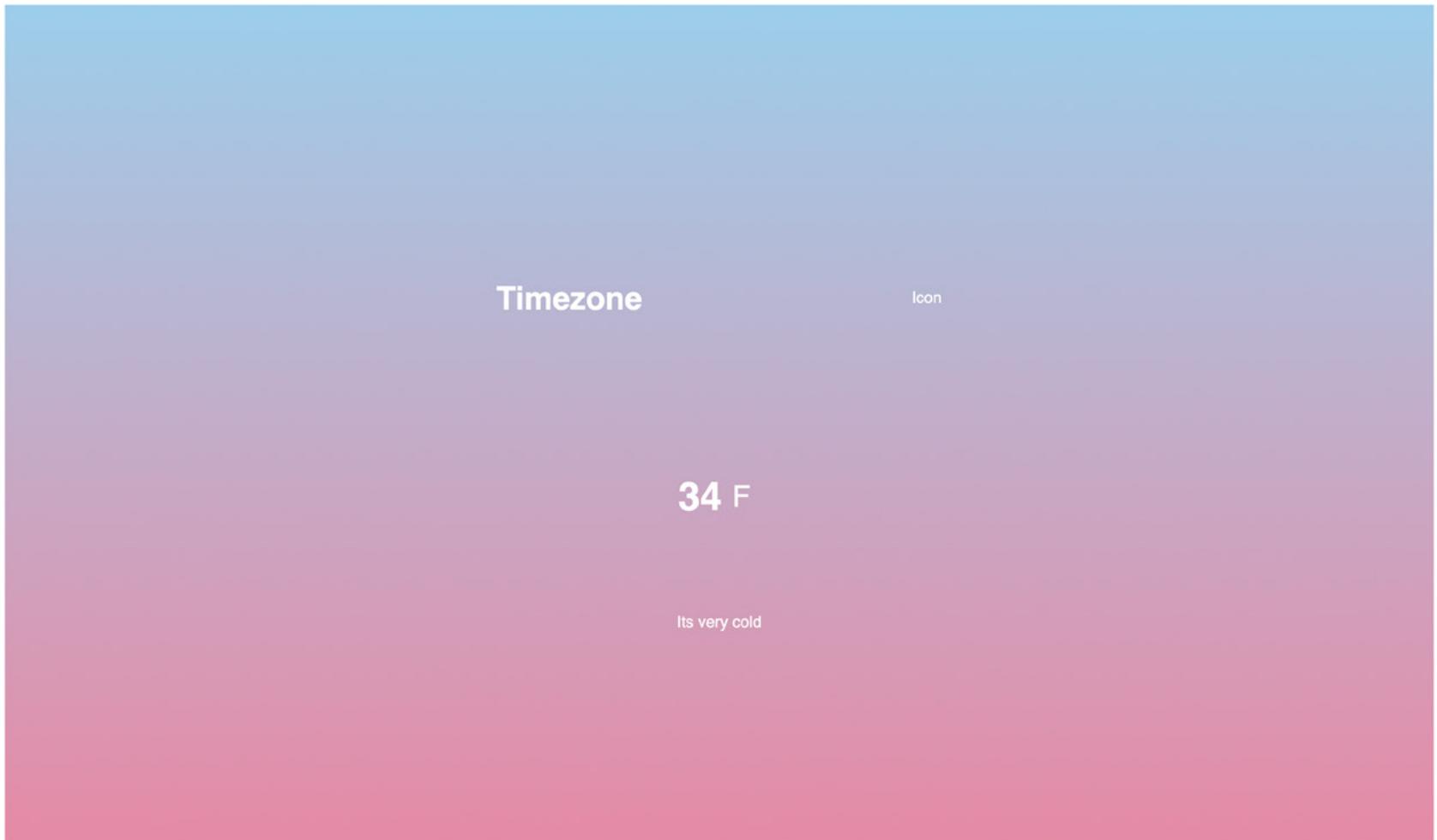
HTML

CSS

Weather app

Tried to make a dynamic weather app that changes depending on location using HTML, CSS and JavaScript. This is my first time trying to use Javascript and i'm still trying to get my head around the basics.

The weather app failed, I think because of the API I was using? I don't fully understand yet.



Weather app

```
window.addEventListener('load', ()=> {
  let long;
  let lat;
  let temperatureDescription = document.querySelector('.temperature-description');
  let temperatureDegree = document.querySelector('.temperature-degree');
  let LocationTimezone = document.querySelector('.Location-Timezone');

  if(navigator.geolocation){
    navigator.geolocation.getCurrentPosition(position => {
      long = position.coords.longitude;
      lat = position.coords.latitude;

      const proxy = "https://cors-anywhere.herokuapp.com/";
      const api = `https://www.weatherapi.com/v1/current/forecast/${lat},${long}`;

      fetch(proxy + api)
        .then(response =>{
          return response.json();
        })
        .then(data =>{
          console.log(data);
          const {temperature, summary} = data.currently;
          //Set DOM Elements from the API
          temperatureDegree.textContent = temperature;
          temperatureDescription.textContent = summary;
        })
    });
  }
});
```

JavaScript

CSS

```
body {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  height: 100vh;
  display: flex;
  justify-content: center;
  flex-direction: column;
  align-items: center;
  background: linear-gradient(45deg, #rgb(156, 206, 235), #rgb(229, 138, 165));
  color: #white;
}

.location,
.temperature{
  height: 30vh;
  width: 50%;
  display:flex;
  justify-content: space-around;
  align-items: center;
}

.temperature {
  flex-direction: column;
}
.degree-section {
  display: flex;
  align-items: center;
  cursor: pointer;
}

.degree-section span {
  margin: 10px;
  font-size: 30px;
}

.degree-section h2 {
  font-size: 40px;
}
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="style.css" />
  <title>Weather App</title>
</head>

<body>

  <div class="location">
    <h1 class="Location-Timezone">Timezone</h1>
    <p>Icon</p>
  </div>

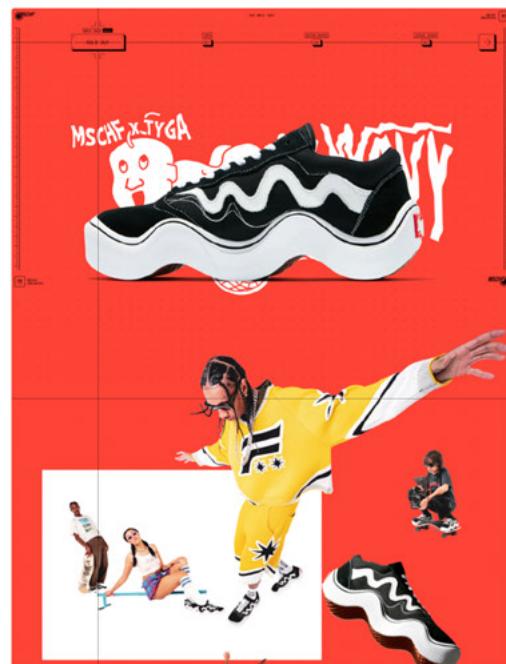
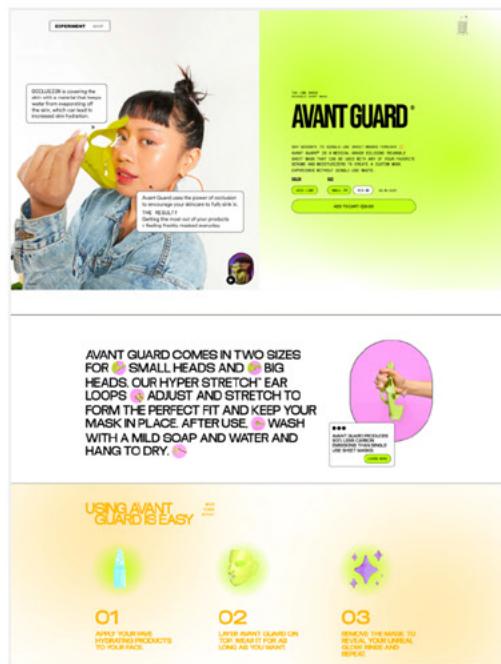
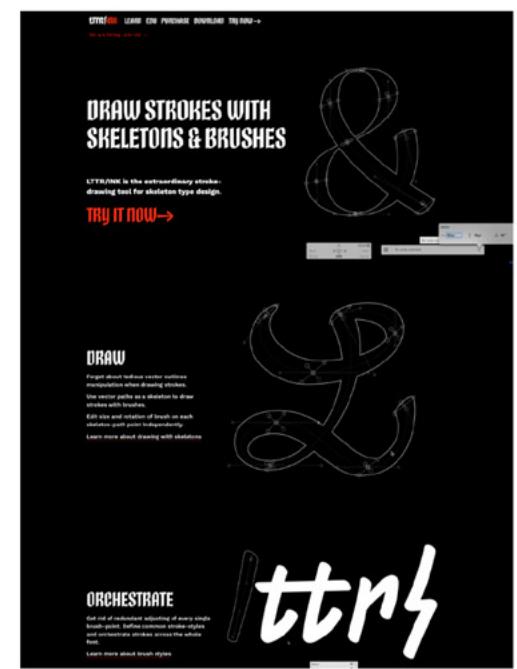
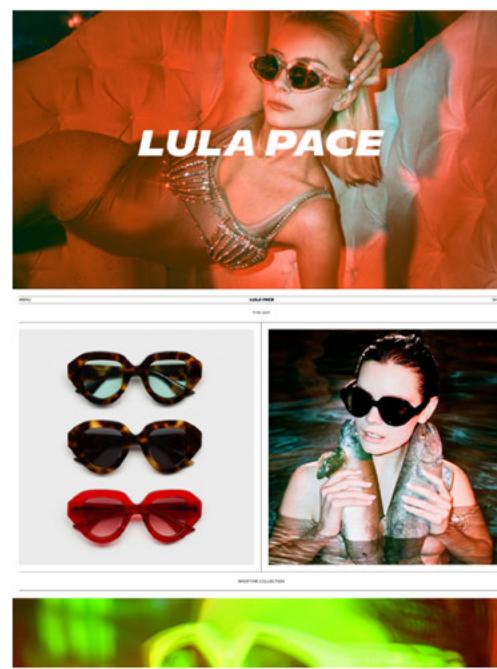
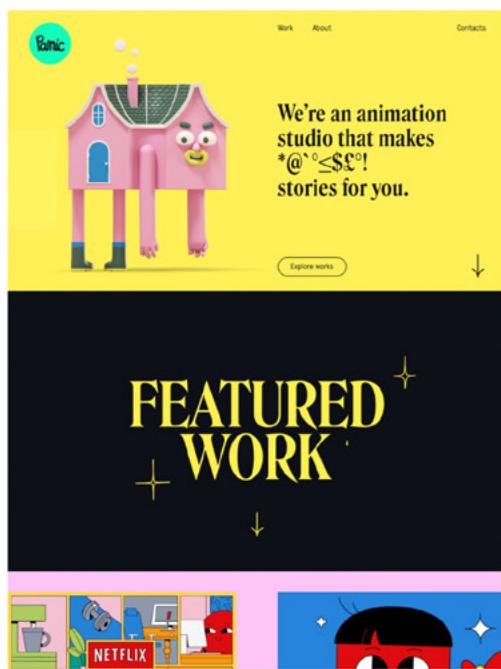
  <div class="temperature">
    <div class="degree-section">
      <h2 class="temperature-degree">34</h2>
      <span>F</span>
    </div>
    <div class="temperature-description">It's very cold</div>
  </div>

</body>

<script src="app.js"></script>
</body>
</html>
```

HTML

Inspiration



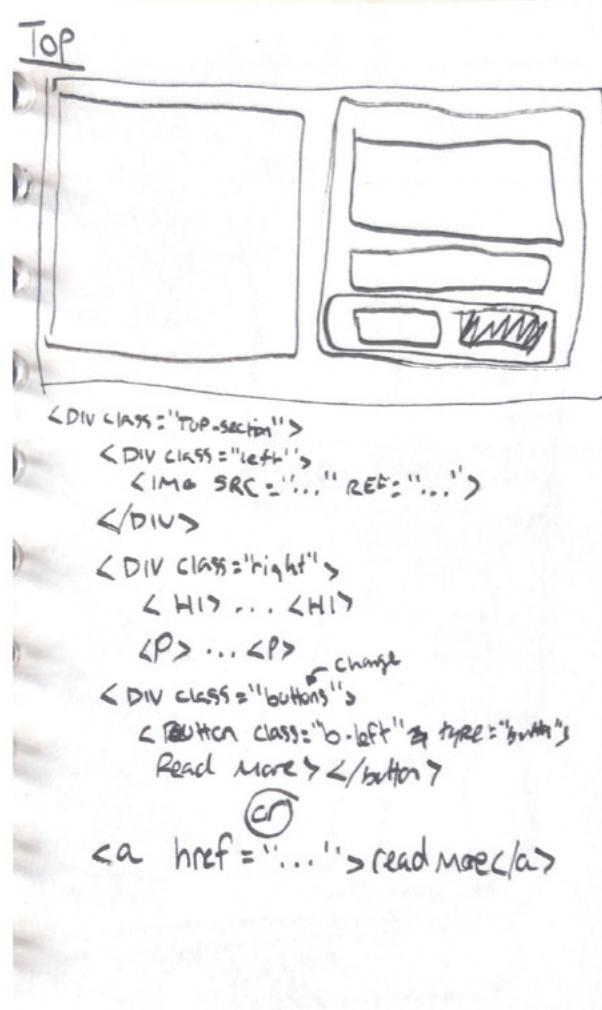
News website

Sketching plan for website

I want to code.

+

Planning code I will write.

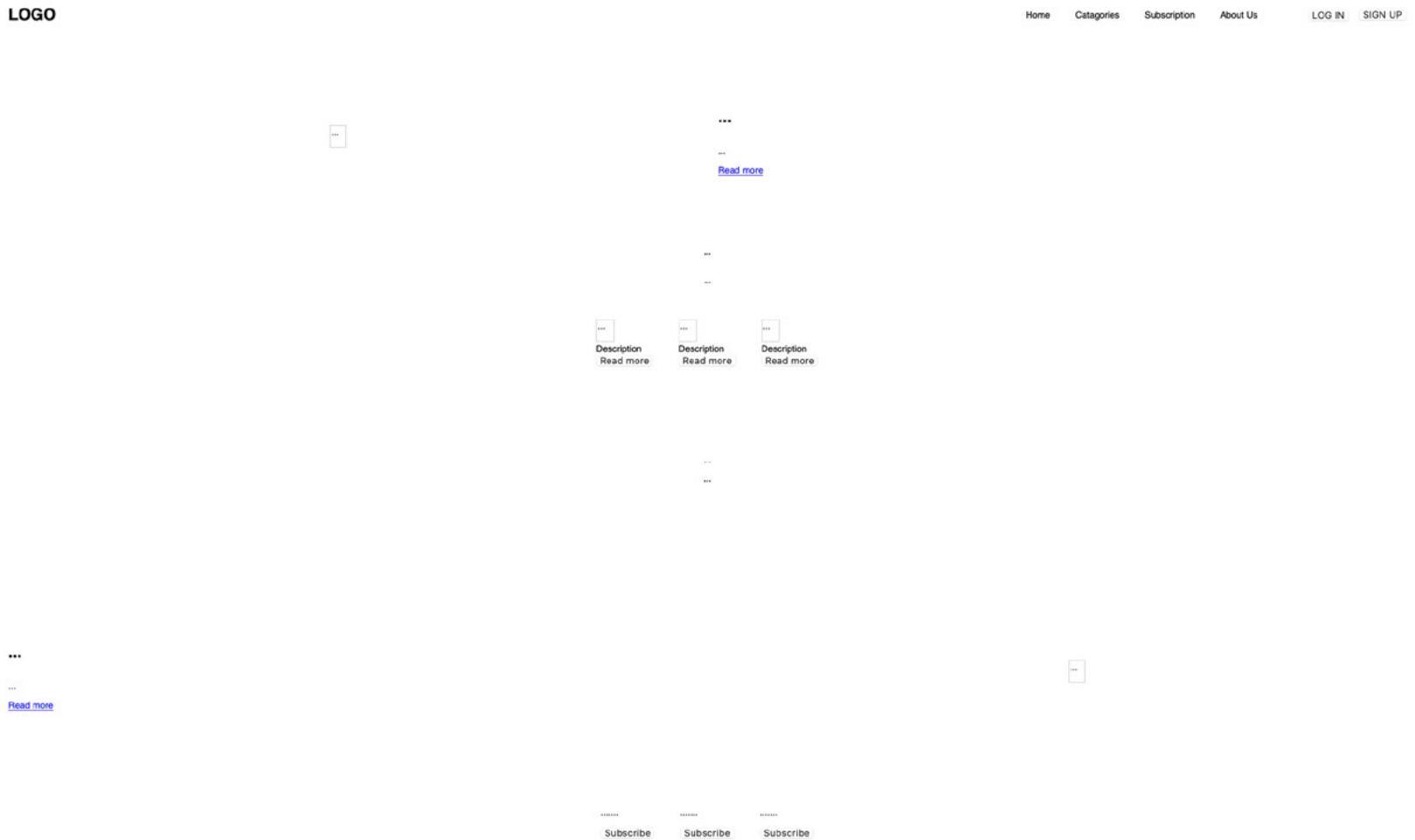


Header flex Space

```
<Header>
  <H1 class="logo">
    | <IMG src="..." ref="...">
  </H1>
  <DIV class="TOP-MENU">
    <UL>
      | <Li> ... </Li>
      | <Li> ... </Li>
      | <Li> ... </Li>
    </UL>
  </DIV>
  <DIV>
    <button class="sign-up" type="button">
      LOG IN </button>
    <button class="LOG-IN" type="button">
      SIGN UP </button>
  </DIV>
```

News website

Added all the elements and then used flexbox (CSS) to lay everything out on the page.



Wanted to practise more with html and CSS. I designed and coded a news website.

With this project I wanted to see if I could link pages, use drop down menus, buttons etc. Aswell as using CSS to create a website that looks good aestheticly.

I'm starting to see how I can combine design + code to yeild results.



Welcome

We are a company committed to bringing you the latest news, around emerging technologies and how they affect culture, the economy, and politics.

[Read more](#)

Keep up to date with the news in your city

Lorem ipsum dolor sit amet consectetur adipisicing elit. Illo facilis ipsum quis omnis ipsa aliquam earum voluptatibus ab aperiam quasi neque accusamus beatae assumenda molestiae error culpa minima, nulla molestias. Lorem ipsum dolor sit.



News website



App revolutionising how people pay their bills

Lorem ipsum, dolor sit amet consectetur adipisicing elit. Non officiis exercitationem optio earum odio neque.

[Read more](#)

US internet providers shutting down in Russia

Lorem ipsum, dolor sit amet consectetur adipisicing elit. Non officiis exercitationem optio earum odio neque.

[Read more](#)

The best deals on true wireless earbuds

Lorem ipsum, dolor sit amet consectetur adipisicing elit. Non officiis exercitationem optio earum odio neque.

[Read more](#)

WhatsApp's new browser extension is making web chats more secure

Code Verify is a new browser extension from WhatsApp parent company Meta that aims to improve the security of WhatsApp's web version, the company has announced. The extension works by verifying that the contents of WhatsApp's web version haven't been tampered with.

[Read more](#)

News website

```
<!DOCTYPE html>
<html lang="en">
<head>
  <link rel="stylesheet" href="https://use.typekit.net/vaz5xjc.css">
  <link rel="stylesheet" href="https://use.typekit.net/vaz5xjg.css">
  <link rel="stylesheet" href="https://use.typekit.net/vaz5xjv.css">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>News Website</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>

  <header class="header">
    
    <ul class="menu">
      <li><strong>Home</strong></li>
      <li><strong>Categories</strong></li>
      <li><strong>Subscription</strong></li>
      <li><strong>About Us</strong></li>
    </ul>
    <div class="top-button">
      <button class="button-50" role="button">LOG IN</button>
    <div class="log-in">
      <button class="button-51" role="button"><i>SIGN UP</i></button>
    </div>
  </div>
  <div class="top-section">
    <div class="left">
      
    </div>
    <div class="right">
      <h1>Welcome</h1>
      <p>We are a company committed to bringing you the latest news, around the world!<br/>
        <button class="button-32" role="button">Read more</button>
        <small>Change to a button element if I need to!</small>
      </div>
    </div>
    <div>
      <strong>Hand</strong>-----</strong>
    </div>
    <div class="text">
      <p>Text above</p>
      <p>Text below</p>
    </div>
  </div>
  <div class="three-news-stories">
    <div class="one">
      
      <div class="tit">
        <p><strong>App</strong>革命ising how people pay their bills</p>
      </div>
      <div class="mid-btn">
        <button class="button-31" role="button">Read more</button>
      </div>
    </div>
    <div class="two">
      
      <div class="tit">
        <p><strong>Oil</strong> Internet providers shutting down</p>
      </div>
      <div class="mid-btn">
        <button class="button-31" role="button">Read more</button>
      </div>
    </div>
    <div class="three">
      
      <div class="tit">
        <p><strong>Students</strong> The best deals on true wireless earbuds</p>
      </div>
      <div class="mid-btn">
        <button class="button-31" role="button">Read more</button>
      </div>
    </div>
  </div>
  <div class="news">
    <div>
      <strong>New!</strong>
      <small>Change to a button element if I need to!</small>
    </div>
    <div>
      <strong>What's New</strong> A new browser extension is making web chats easier<br/>
        <small>Code Verify is a new browser extension from WhatsApp</small>
        <small>Change to a button element if I need to!</small>
      </div>
    </div>
    <div class="right">
      
    </div>
  </div>
  <div class="hand">
    <strong>Hand</strong>-----</strong>
  </div>
  <div class="subscriptions">
    <div class="sub-one">
      <p><strong>Free Plan</strong></p>
      <p><strong>£0</strong>/mo</p>
      <ul class="perks">
        <li>Access to free articles</li>
      </ul>
    </div>
    <div class="sub-two">
      <p><strong>Standard Plan</strong></p>
      <p><strong>£5</strong>/mo</p>
      <ul class="perks">
        <li>Access to all articles</li>
        <li>Access to all videos</li>
        <li>Email newsletter</li>
        <li>25% Discount</li>
      </ul>
    </div>
    <div class="sub-three">
      <p><strong>Advanced Plan</strong></p>
      <p><strong>£15</strong>/mo</p>
      <ul class="perks">
        <li>All perks mentioned</li>
        <li>Our monthly Magazine</li>
      </ul>
    </div>
  </div>
  <div class="sub-btn">
    <button class="button-50" role="button">SUBSCRIBE</button>
  </div>
  <div class="sub-btn">
    <button class="button-50" role="button">SUBSCRIBE</button>
  </div>
  <div class="sub-btn">
    <button class="button-50" role="button">SUBSCRIBE</button>
  </div>
</body>
```

Free Plan

£0/mo

- Access to free articles

SUBSCRIBE

Standard Plan

£5/mo

- Access to all articles
- Access to all videos
- Email newsletter
- 25% Discount

SUBSCRIBE

Advanced Plan

£15/mo

- All perks mentioned
- Our monthly Magazine

SUBSCRIBE

Copyright Leo Roberts 2022

HTML

```
.header {  
  display: flex;  
  justify-content: space-between;  
  margin-left: 50px;  
  margin-right: 50px;  
  margin-bottom: 50px;  
  margin-top: 25px;  
}  
  
.scoot {  
  width: 250px;  
  height: auto;  
}  
  
.menu {  
  display: flex;  
  list-style: none;  
  gap: 20px;  
  justify-content: flex-end;  
  align-items: center;  
  justify-content: flex-end;  
  flex-grow: 1;  
  margin-right: 100px;  
  gap: 50px;  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
}  
  
.top-buttons {  
  display: flex;  
  justify-content: space-between;  
  align-items: center;  
  gap: 20px;  
}  
  
.top-section {  
  display: flex;  
  height: 550px;  
  margin-left: 50px;  
  margin-right: 50px;  
  justify-content: space-between;  
  margin-bottom: 20px;  
}  
  
.left {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  align-self: center;  
  margin-right: 25px;  
  width: max-content;  
}  
  
.image1 {  
  width: 700px;  
  height: auto;  
}  
  
.right {  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  align-items: flex-start;  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
  margin-left: 50px;  
  flex-grow: 1;  
  line-height: 30px;  
}  
  
.welcome {  
  font-family: "auger-mono", monospace;  
  font-weight: 700;  
  font-style: normal;  
  font-size: 100px;  
  margin-bottom: 30px;  
  color: #f94144;  
}  
  
.tu {  
  font-family: "balto", sans-serif;  
  font-weight: 400;  
  font-style: normal;  
}  
  
.a {  
  text-decoration: none;  
}  
  
.h2 {  
  display: flex;  
  flex-direction: column;  
  margin-left: 100px;  
  margin-right: 100px;  
  margin-bottom: 75px;  
  line-height: 30px;  
}  
  
.text-above {  
  text-align: center;  
  margin-left: 50px;  
  margin-right: 50px;  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
  font-size: 30px;  
}  
  
.text-below {  
  text-align: center;  
  margin-left: 100px;  
  margin-right: 100px;  
  font-family: "balto", sans-serif;  
  font-weight: 400;  
  font-style: normal;  
}  
  
.three-news-stories {  
  display: flex;  
  justify-content: space-between;  
  gap: 50px;  
  margin-top: 50px;  
  margin-left: 50px;  
  margin-right: 50px;  
  line-height: 30px;  
}  
  
.under-text {  
  font-family: "balto", sans-serif;  
  font-weight: 400;  
  font-style: normal;  
  margin-bottom: 50px;  
}  
  
.t1 {  
  width: 400px;  
  max-height: 150px;  
}  
  
.news-tit {  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
  font-weight: 400;  
  font-style: normal;  
  font-size: 22px;  
}  
  
.image2 {  
  width: 400px;  
  height: auto;  
}  
  
.image3 {  
  width: 400px;  
  height: auto;  
}  
  
.image4 {  
  width: 400px;  
  height: auto;  
}  
  
.emoji {  
  height: auto;  
  display: flex;  
  justify-content: center;  
  line-height: 30px;  
  margin-top: 50px;  
  margin-bottom: 50px;  
}  
  
.hand {  
  font-size: 70px;  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
  text-align: center;  
  margin-bottom: 30px;  
  margin-top: 30px;  
}  
  
.emoji {  
  height: 140px;  
}  
  
.mid-news {  
  display: flex;  
  height: 400px;  
  margin-left: 50px;  
  margin-right: 50px;  
  justify-content: space-between;  
  margin-bottom: 100px;  
}  
  
.right-2 {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  align-self: center;  
  margin-left: 50px;  
  width: max-content;  
  margin-top: 50px;  
  margin-bottom: 50px;  
}  
  
.image5 {  
  width: 700px;  
  height: auto;  
}  
  
.left-2 {  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  align-items: flex-start;  
  margin-right: 50px;  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
  margin-top: 50px;  
}  
  
.h1 {  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
}  
  
.para {  
  line-height: 30px;  
  font-family: "balto", sans-serif;  
  font-weight: 400;  
  font-style: normal;  
}  
  
.hand2 {  
  font-size: 70px;  
  font-family: "auger-mono", monospace;  
  font-weight: 400;  
  font-style: normal;  
  text-align: center;  
  margin-bottom: 30px;  
  margin-top: 14px;  
}  
  
.subscriptions {  
  display: flex;  
  justify-content: space-between;  
  gap: 50px;  
  margin-top: 150px;  
  margin-bottom: 175px;  
  gap: 100px;  
  line-height: 30px;  
  font-family: "balto", sans-serif;  
  font-weight: 400;  
  font-style: normal;  
  background-color: #f94144;  
}  
  
.foot {  
  color: #fff;  
}  
  
.button-50 {  
  appearance: button;  
  background-color: #0000;  
  background-image: none;  
  border: 1px solid #0000;  
  border-radius: 4px;  
  box-shadow: #ffff 4px 4px 0, #0000 4px 4px 0 1px;  
  box-sizing: border-box;  
  color: #fff;  
  cursor: pointer;  
  display: inline-block;  
  font-family: ITCAvantGardeStd-Bk,Arial,sans-serif;  
  font-size: 14px;  
  font-weight: 400;  
  line-height: 10px;  
  margin: 0 5px 10px 0;  
  overflow: visible;  
  padding: 12px 40px;  
  text-align: center;  
  text-transform: none;  
  touch-action: manipulation;  
  user-select: none;  
  -webkit-user-select: none;  
  vertical-align: middle;  
  white-space: nowrap;  
}  
  
.perks {  
  list-style-type: none;  
  text-align: left;  
  padding: 0px;  
}  
  
.plan {  
  font-family: "auger-mono", monospace;  
  font-weight: 700;  
  font-style: normal;  
  font-size: 20px;  
}  
  
.price {  
  font-size: 40px;  
  margin-top: 5px;  
  margin-bottom: 5px;  
}  
  
.sub-one {  
  outline-style: solid;  
  outline-color: #rgb(0, 0, 0);  
  outline-offset: 20px;  
  width: 300px;  
  display: flex;  
  flex-direction: column;  
  justify-content: space-between;  
}  
  
.button-50:hover {  
  text-decoration: none;  
}  
  
.button-50:active {  
  box-shadow: #rgba(0, 0, 0, .125) 0 3px 5px inset;  
  outline: 0;  
}
```

css

Learning JavaScript

I'm struggling to get my head round JavaScript.

Gonna go back to FreeCodeCamp.com to try and learn the basics properly before I move on to doing my own projects.

FreeCodeCamp gives u a lot of challenges and tutorials that walk you through different elements of coding. it's very easy and doesn't challenge me much but it does lay things out very clearly.

I used this when i first started HTML + CSS and it gave me a good knowledge to go forward with my own projects.

There are 300 lessons. I don't plan on completing all of them. I learn more from personal projects.

The screenshot shows a FreeCodeCamp tutorial page. At the top, there's a search bar, a logo, and a menu icon. Below the header, the title 'freeCodeCamp' is followed by a navigation bar with 'JavaScript Algorithms and Data Structures' and 'Basic JavaScript'. The main content area has a dark background with white text. It starts with a section titled 'Escaping Literal Quotes in Strings' which explains that to include a quote character inside a string, you need to escape it with a backslash. It shows a code example:

```
const sampleStr = "Alan said, \"Peter is learning JavaScript\";";
```

This signals to JavaScript that the following quote is not the end of the string, but should instead appear inside the string. So if you were to print this to the console, you would get:

```
Alan said, "Peter is learning JavaScript".
```

Below this, there's a note: 'Use backslashes to assign a string to the `myStr` variable so that if you were to print it to the console, you would see:' followed by another code example:

```
I am a "double quoted" string inside "double quotes".
```

At the bottom of the page, there are three buttons: 'Run the Tests (Ctrl + Enter)', 'Reset All Code', and 'Get Help'. A note at the bottom says: 'You should use two double quotes ("") and four escaped double quotes (\").' On the right side, there's a vertical sidebar with the text 'myStr = ""'.

Example of lesson.

What I've been learning

I wanted to step back and appreciate some of things I've taught myself so far. I've learnt a lot in a short amount of time and this can sometimes be overwhelming.

I think listing the things I've learnt will be good for me.

I'll add more to this list if I realised I forgot something. And will probably do another list later on in the project.

- HTML basics
- CSS basics
- Attributes
- Setting up file (!DOCTYPE)
- Linking pages together
- Tags
- Images and Video
 - Fonts
 - ID's
 - Classes
 - Flexbox
 - Grid
 - Animation
 - Box model
 - Styling pages
 - Forms
 - Block and inline
 - Semantic HTML
 - Validations
- Conventions/best practise?
 - Accessibility
 - Making layouts
 - Dynamic layouts
 - Responsive design
 - Positioning
 - Repo hosting
 - Github

- Git
- Terminal
- Repositories
- Developer tools
- Career paths
- Libraries
- Basic JavaScript
- Frameworks
- Emmet

Backend

Backend developers build code that allows a database and an application to communicate with one another. Backend developers take care and maintain the back-end of a website, including databases, servers, and apps, and they control what you don't see.

www.simplelearn.com

Although I'm much more interested in front-end web development, I think it's important for me to have some basic understanding of backend as well.

Some backend languages:

PHP

C#

Java

Ruby on Rails

Python



Frontend Frameworks

A front end framework is something coded by someone else that assists you in building front-end web technology at a faster pace compared to doing it on your own.

I don't think i'm ready to learn frameworks yet but It's useful to know that they exist, and it's something that I want to learn in the future as I improve.

A few examples of frameworks :
Angular
Vue.js
Bootstrap
Svelte

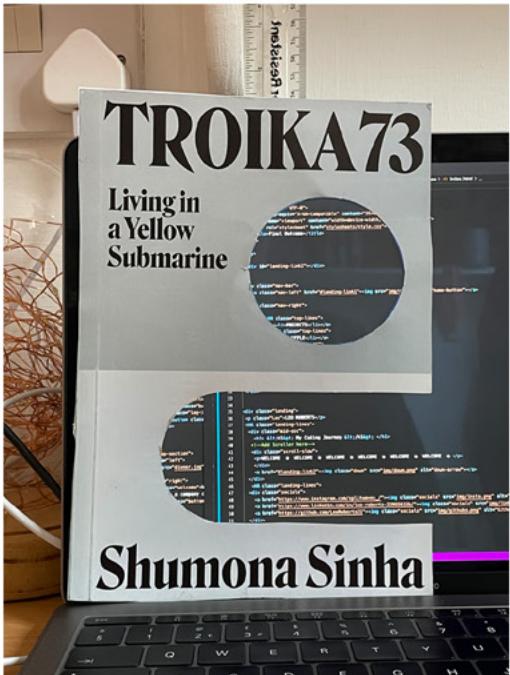
As soon as I have a stronger understanding of JavaScript I want to start learning some of these.



Collage

Found a nice poster in a magazine, wanted to cut out shapes and experiment with the empty space.

I think the actual code has a really nice visual aspect to it so I wanted to think about ways I can bring a more artful feeling to my code.



TROIKA73

Living in a Yellow Submarine

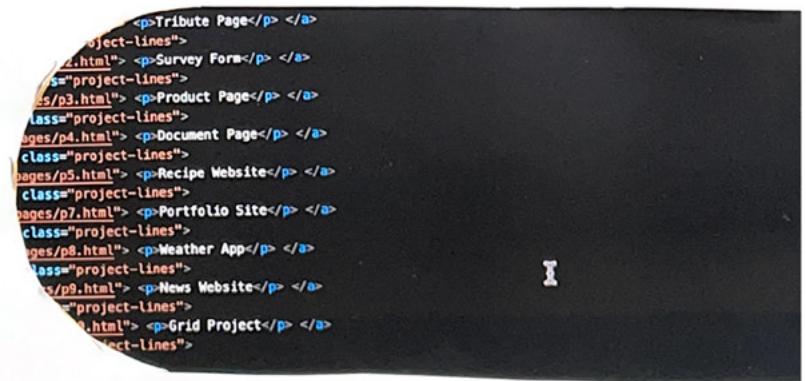


TROIKA73

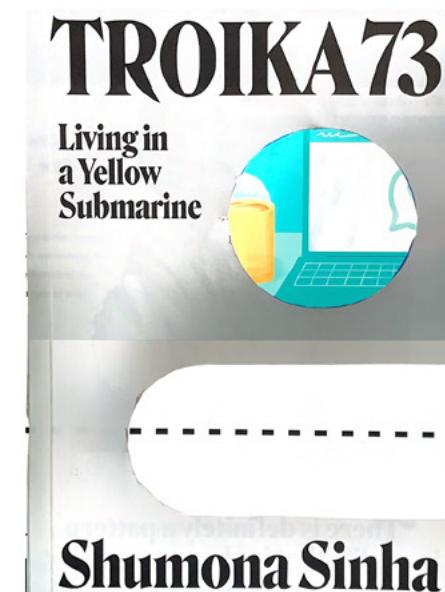
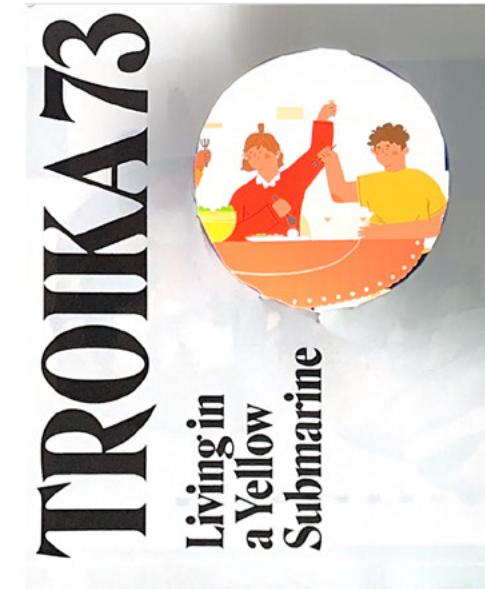
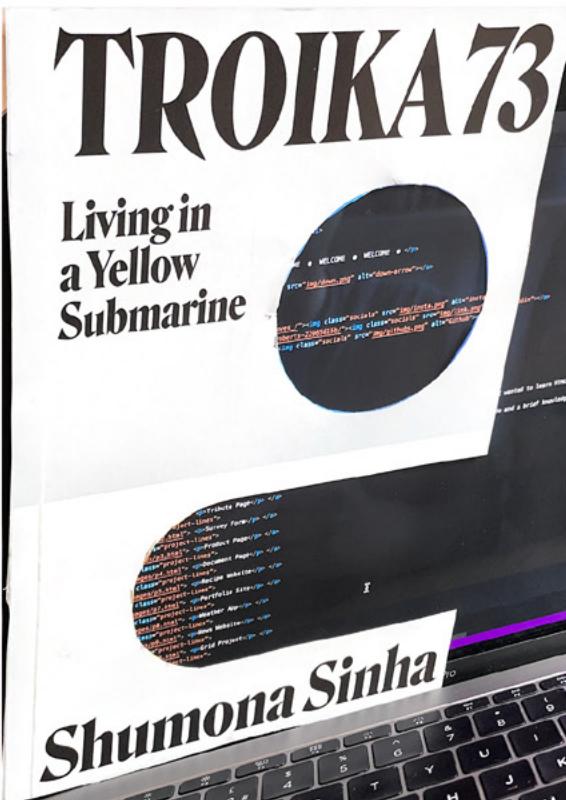
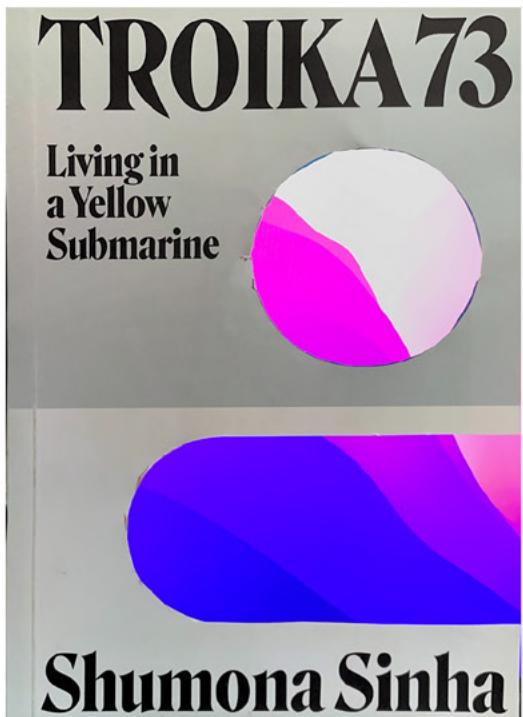
Living in a Yellow Submarine



Shumona Sinha



Collage



Learning CSS grid

Been learning how to organize items using grid in CSS. This feature allows you to arrange things in a webpage in a grid e.g YouTube videos..

Things I've been learning:

- What the grid is
- Setting up a grid
- Lines and tracks
- Columns and rows
- Explicit vs implicit grid

Gap

Cells

Repeat

MinMax

Fractional Units

Dynamic MinMax

Auto-Fit and Auto-Fill

Content First vs Layout-First Design

Combining Flexbox - and Grid

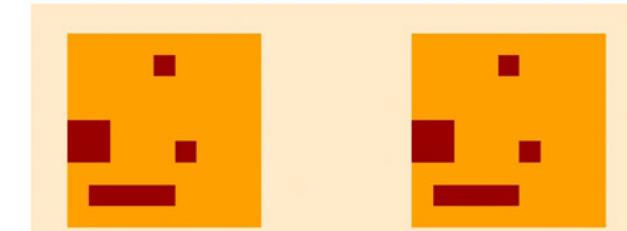
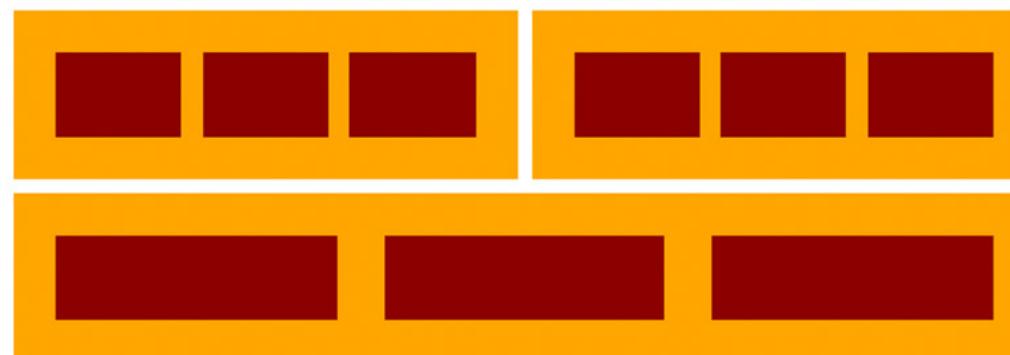
EDIT ON [CODEPEN](#)

HTML SCSS Result

```
.layout {  
  display: flex;  
  flex-wrap: wrap;  
}  
.child {  
  min-height: 100px;  
  background: orange;  
  flex: calc(100% / 3);  
  text-align: center;  
  line-height: 100px;  
  border: 5px solid white;  
}  
  
* {  
  box-sizing: border-box;  
}
```



Resources View Compiled 1x 0.5x 0.25x Rerun



The orange elements are **flex items** and **grid parents**.

The red elements are **grid items**.

The orange elements are **grid items** and **flex parents**.

The red elements are **flex children**

CSS Grid experimentation

Made this layout using grid, was tricky to get my head round. Using it in practise is a lot harder than reading and thinking you understand it.

```
.container {
  text-align: center;
  height: 100%;
  padding: 16px;
  box-sizing: border-box;
  display: grid;
  gap: 15px;
  grid-template-columns: 300px 900px;
  grid-template-rows: 100px 100px 500px 100px;
  grid-template-areas:
    "header header"
    "sidebar nav"
    "sidebar article"
    "footer footer"

}

.header {
  background-color: #FFDE22;
  grid-area: header;
}

.sidebar {
  background-color: #FF7755;
  grid-area: sidebar;
}

.nav {
  background-color: #00DDFF;
  grid-area: nav;
}

.article {
  background-color: #bccbde;
  grid-area: article;
}

.article p {
  font-size: 18px;
  font-family: sans-serif;
  color: white;
  text-align: left;
}

.footer {
  background-color: #393f4d;
  grid-area: footer;
```

Header

Sidebar

Nav

Article

Lore ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Curabitur bibendum turpis quis interdum semper. Sed at pharetra neque, nec lacinia diam. Suspendisse quis faucibus mi, aliquam porttitor ipsum. Vivamus condimentum eros id mattis pharetra. Duis varius eros nibh. Donec a venenatis eros. Fusce in mauris massa. Donec est metus, rhoncus eu leo sed, aliquet posuere nisl.

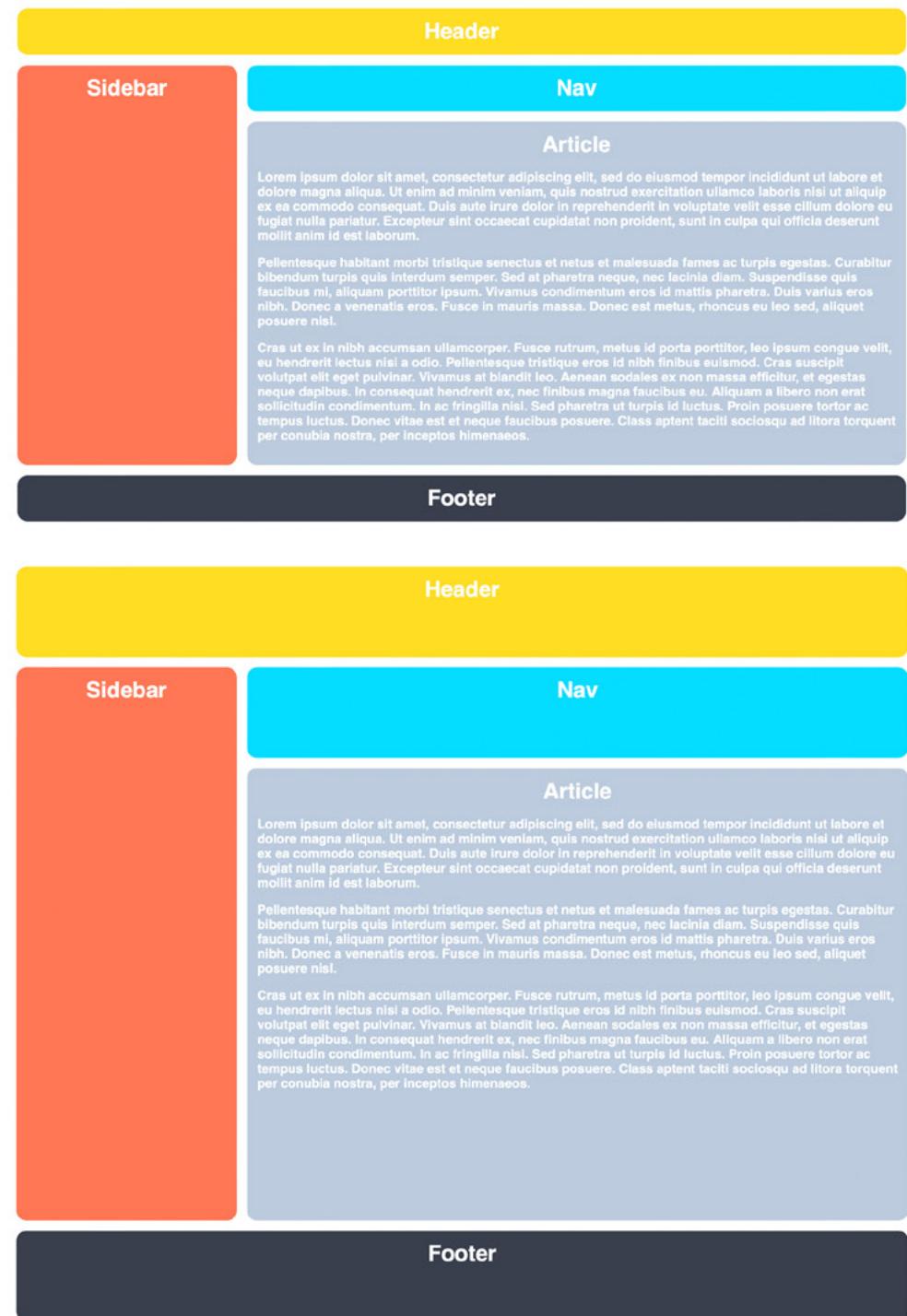
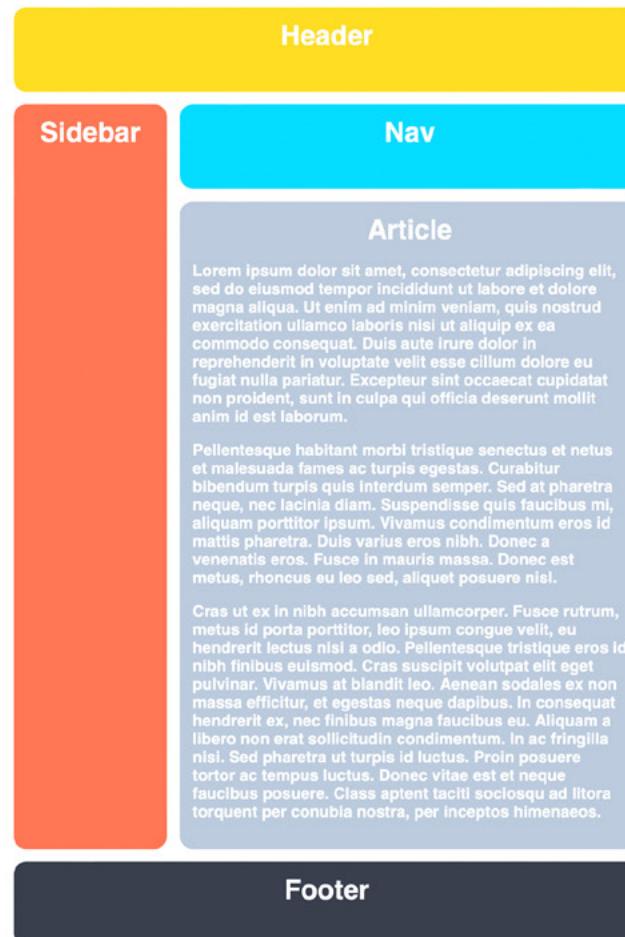
Cras ut ex in nibh accumsan ullamcorper. Fusce rutrum, metus id porta porttitor, leo ipsum congue velit, eu hendrerit lectus nisi a odio. Pellentesque tristique eros id nibh finibus euismod. Cras suscipit volutpat elit eget pulvinar. Vivamus at blandit leo. Aenean sodales ex non massa efficitur, et egestas neque dapibus. In consequat hendrerit ex, nec finibus magna faucibus eu. Aliquam a libero non erat sollicitudin condimentum. In ac fringilla nisi. Sed pharetra ut turpis id luctus. Proin posuere tortor ac tempus luctus. Donec vitae est et neque faucibus posuere. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

Footer

CSS Grid experimentation

I made the grid responsive to different window sizes using the 'fr' unit instead of pixels. This splits the grid into fractional units that will always apply no matter what size the window is.

```
grid-template-columns: 0.25fr 0.75fr;  
grid-template-rows: 1fr 1fr 5fr 1fr;
```



CSS Grid experimentation

Made this grid layout using grid and flex. It's also responsive to changing screen sizes.

Next I want to make something similar but with more design elements, something that looks more like a website.

The wireframe consists of a header section and a main content area. The header has a dark blue background with the text "Header Logo" and four navigation links: "header link one", "header link two", "header link three", and "header link four". Below the header is a "Placeholder for Image" box. The main content area is divided into three horizontal sections, each containing three columns of article cards. The first section is titled "LATEST ARTICLES", the second "MOST VIEWS", and the third "FEATURED". Each card includes an article title and a short description. At the bottom of the main content area is a copyright notice: "Copyright © The Odin Project 2021".

This wireframe is more complex than the first. It features a header with the text "Header Logo" and four navigation links. Below the header is a "Placeholder for Image" box. The main content area is organized into several sections: "LATEST ARTICLES", "MOST VIEWS", and "FEATURED", each containing multiple article cards. Each card displays an article title and a brief description. The entire layout is contained within a dark grey footer bar at the bottom. A copyright notice "Copyright © The Odin Project 2021" is located at the very bottom of the page.

This wireframe is identical to the one above it, featuring a header with "Header Logo" and four navigation links, a "Placeholder for Image" box, and a main content area with "LATEST ARTICLES", "MOST VIEWS", and "FEATURED" sections. Each section contains multiple article cards with titles and descriptions. The layout is framed by a dark grey footer bar at the bottom, which includes a copyright notice: "Copyright © The Odin Project 2021".

CSS Grid experimentation

Code from previous exercise. The HTML isn't mine as this was a challenge exercise and I'm learning grid which involves CSS.

```
* {
    margin: 0;
    padding: 0;
}

.container {
    text-align: center;
    display: grid;
    grid-template-columns: 1fr 3fr;
    grid-template-rows: 0.25fr 6fr 1fr;
    gap: 15px;
    grid-template-areas: " sidebar nav"
    "sidebar article"
    "sidebar article"
    "footer footer"
}

.container div {
    padding: 15px;
    font-size: 32px;
    font-family: Helvetica;
    font-weight: bold;
    color: #white;
}

.header {
    background-color: #393f4d;
    display: flex;
    justify-content: space-between;
    align-items: center;
    text-align: center;
    padding: 15px;
    font-size: 32px;
    font-family: Helvetica;
    font-weight: bold;
    color: #white;
    margin-bottom: 15px;
}

.logo {
    margin-left: 30px;
}

.menu {
}

.menu ul {
    display: flex;
    gap: 30px;
    list-style: none;
    align-items: center;
    margin-right: 30px;
}

.menu ul,
.menu li {
    font-size: 16px;
}

.sidebar {
    background-color: #FF7755;
    grid-area: sidebar;
    display: grid;
    grid-template-columns: 1fr;
    grid-template-rows: 1fr 1fr 1fr 1fr;
    gap: 15px;
}

.sidebar .photo {
    background-color: #white;
    color: black;
    font-size: 12px;
    font-weight: normal;
    border-radius: 10px;
    display: flex;
    align-items: center;
    justify-content: center;
}

.sidebar .side-content {
    background-color: #white;
    color: black;
    font-size: 16px;
    font-weight: normal;
    display: flex;
    align-items: center;
    justify-content: center;
}

.nav {
    background-color: #FF7755;
    grid-area: nav;
}

.nav ul {
    display: flex;
    list-style: none;
    justify-content: space-between;
    margin-right: 30px;
    align-items: center;
}

.nav ul li {
    font-size: 16px;
    text-transform: uppercase;
}

.article {
    background-color: #bccbde;
    grid-area: article;
    display: grid;
    grid-template-columns: 1fr 1fr 1fr;
    grid-template-rows: 1fr 1fr 1fr;
    gap: 15px;
}

.article p {
    font-size: 18px;
    font-family: sans-serif;
    color: #white;
    text-align: left;
}

.article .card {
    background-color: #FFFFFF;
    color: black;
    text-align: center;
}

.card p {
    color: black;
    font-weight: normal;
    font-size: 14px;
}

.card .title {
    font-size: 18px;
    text-align: center;
}

.footer {
    background-color: #393f4d;
    grid-area: footer;
}

.footer p {
    font-size: 13px;
    font-weight: normal;
}
```

CSS

Dropdown menu



Added a dropdown menu to my news website.



Home Categories Subscription About Us

LOG IN

SIGN UP

- Computing
- Gaming
- Home
- Entertainment
- Mobile
- Software

Welcome

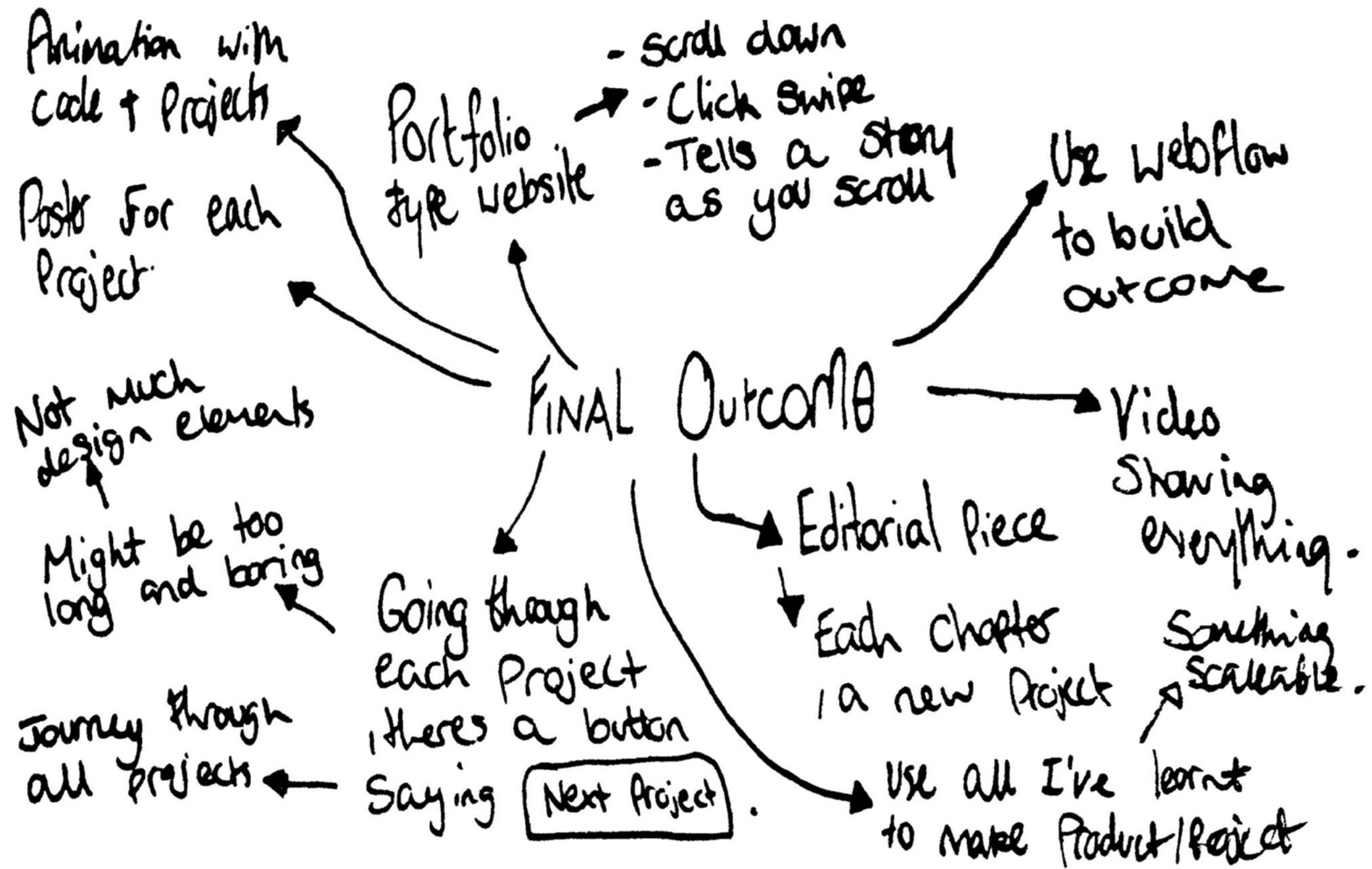
We are a company committed to bringing you the latest news, around emerging technologies and how they affect culture, the economy, and politics.

[Read more](#)

```
.dropdown {  
    position: relative;  
    display: inline-block;  
}  
  
.dropdown-content {  
    display: none;  
    position: absolute;  
    background-color: #f9f9f9;  
    min-width: 160px;  
    box-shadow: 0px 8px 16px 0px rgba(80, 80, 80, 0.2);  
    padding: 12px 16px;  
    z-index: 1;  
    line-height: 30px;  
}  
  
.dropdown:hover .dropdown-content {  
    display: block;  
}
```

```
<div class="dropdown">  
    <span><strong>Categories</strong></span>  
    <div class="dropdown-content">  
        <p>Computing<br>  
            Gaming<br>  
            Home Entertainment<br>  
            Mobile<br>  
            Software</p>  
    </div>  
</div>
```

Final outcome

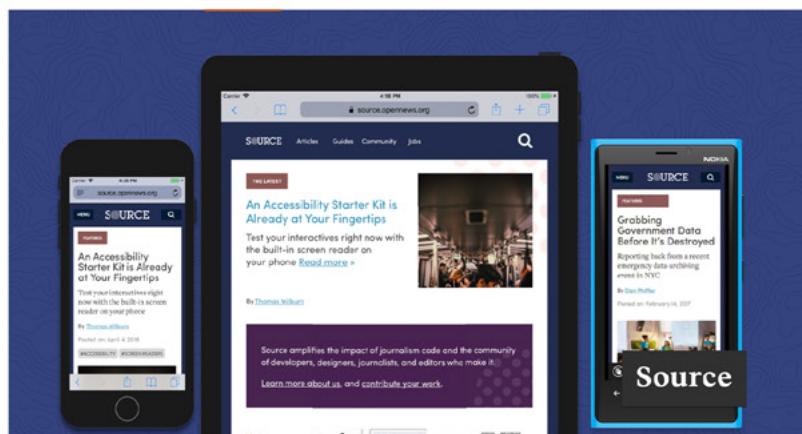
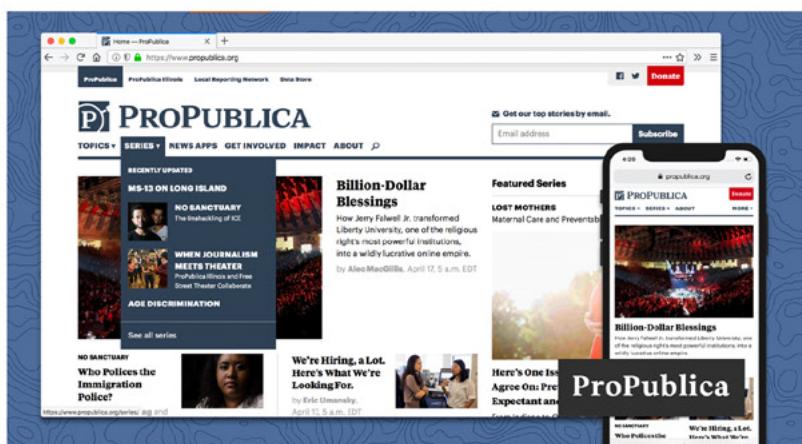


Ethan Marcotte

Ethan Marcotte is a web designer based in Boston, Massachusetts. He was the person who coined the term 'responsive web design' to describe a new way of designing for ever changing web.

This project has made me realise how important it is for a website to have be responsive, and it seems this need for responsive websites will only get larger in the future. Therefore I think it would be helpful for me to learn more about it, so I might asewll start with the person Ethan Marcotte.

These are some of the responsive websites he's worked on + his personal website and how it changes when it's scaled down.



ETHAN MARCOTTE

JOURNAL Work About Contact

My journal.

Here're a few words I've written. If you like them, you can sign up to have [new blog entries emailed directly to you](#), or you can [subscribe to my RSS feed](#).

Some of my more commonly-used tags: 2021

#responsive #work #process #personal #webdesign
#designsystems #regulation #design #politics #writing
#inclusion #justice #labor #performance #universalaccess
#conferences #designpatterns #googleamp #ursulafranklin

ETHAN MARCOTTE

JOURNAL Work About Contact

My journal.

Here're a few words I've written. If you like them, you can sign up to have [new blog entries emailed directly to you](#), or you can [subscribe to my RSS feed](#).

Some of my more commonly-used tags:

#responsive #work #process #personal #webdesign
#designsystems #regulation #design #politics #writing
#inclusion #justice #labor #performance #universalaccess
#conferences #designpatterns #googleamp #ursulafranklin

2021

SEPTEMBER

Locus.

What's better, where, and for whom. — 28 September 2021

013: The Map and the Territory

The talk is about the weight of the web, forks in the path, and the need for open, wide roads.

But first, I'd like to tell you a story. Specifically, the story of a philosopher, a map-maker, and a cup of coffee.

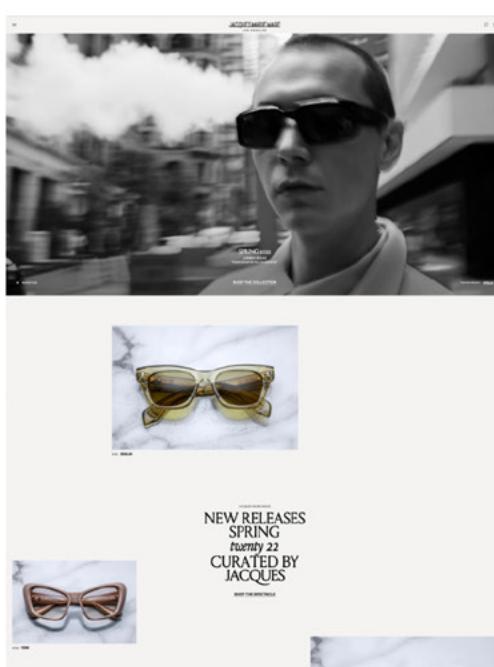
Sounds like the set-up to a bad joke, doesn't it? Well, maybe it is. But it's a story, and hopefully a good one at that. And as the best stories I've ever read start in the middle, let's do the same. Let's begin with a map.

Editorially

Inspiration

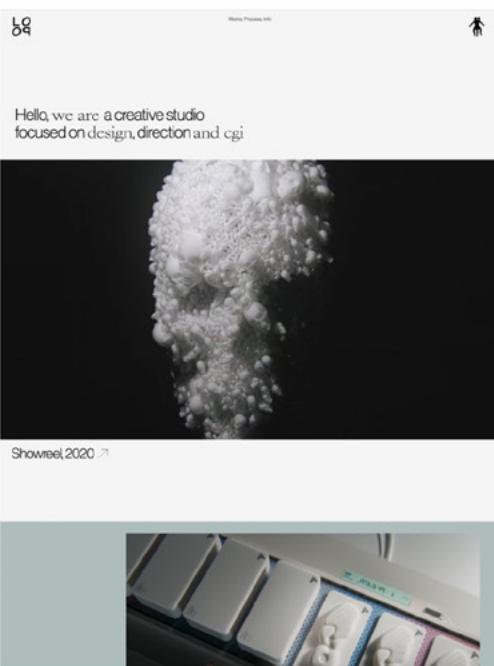
The image is a collage of various design-related snippets. At the top left, there's a 'TRY THIS' callout with a wavy arrow. The central part features a large, bold 'MULTI-DISCIPLINARY DESIGNER' title. Below it, the words 'showing off' are partially visible. To the left, a person is shown working at a desk with multiple monitors displaying design software. To the right, a basketball player is in motion. At the bottom, there's a logo for 'the great lake michigan' featuring a circular emblem with a lighthouse. Various other snippets include a 'CHECKLIST TO IMPROVE YOUR FLOW STATE' graphic, a 'THE LAST DROP' logo, and a snippet of text about design as a way to make the world better.

The image is a collage of various screenshots and illustrations from The New York Times website. At the top, there's a header with 'The New York Times' logo and navigation links like 'About', 'Opinion', 'Opinion View', 'Opinion Opinions', and 'Opinion Op-Ed'. Below this is a large, bold headline 'Welcome to the greatest adventure of your life.' followed by a subtext about sobriety and a 'Get Started' button. In the center, there's a colorful illustration of a diverse group of people standing together against a yellow background. Below this is a quote: 'The opposite of addiction isn't sobriety, it's connection.' On the left, there's a section titled 'The Connection' with a subtext about connecting with others who share similar interests. On the right, there's a section titled 'Psyching Out' with a subtext about mental health. At the bottom, there's a large, grainy image of a crowd of people.

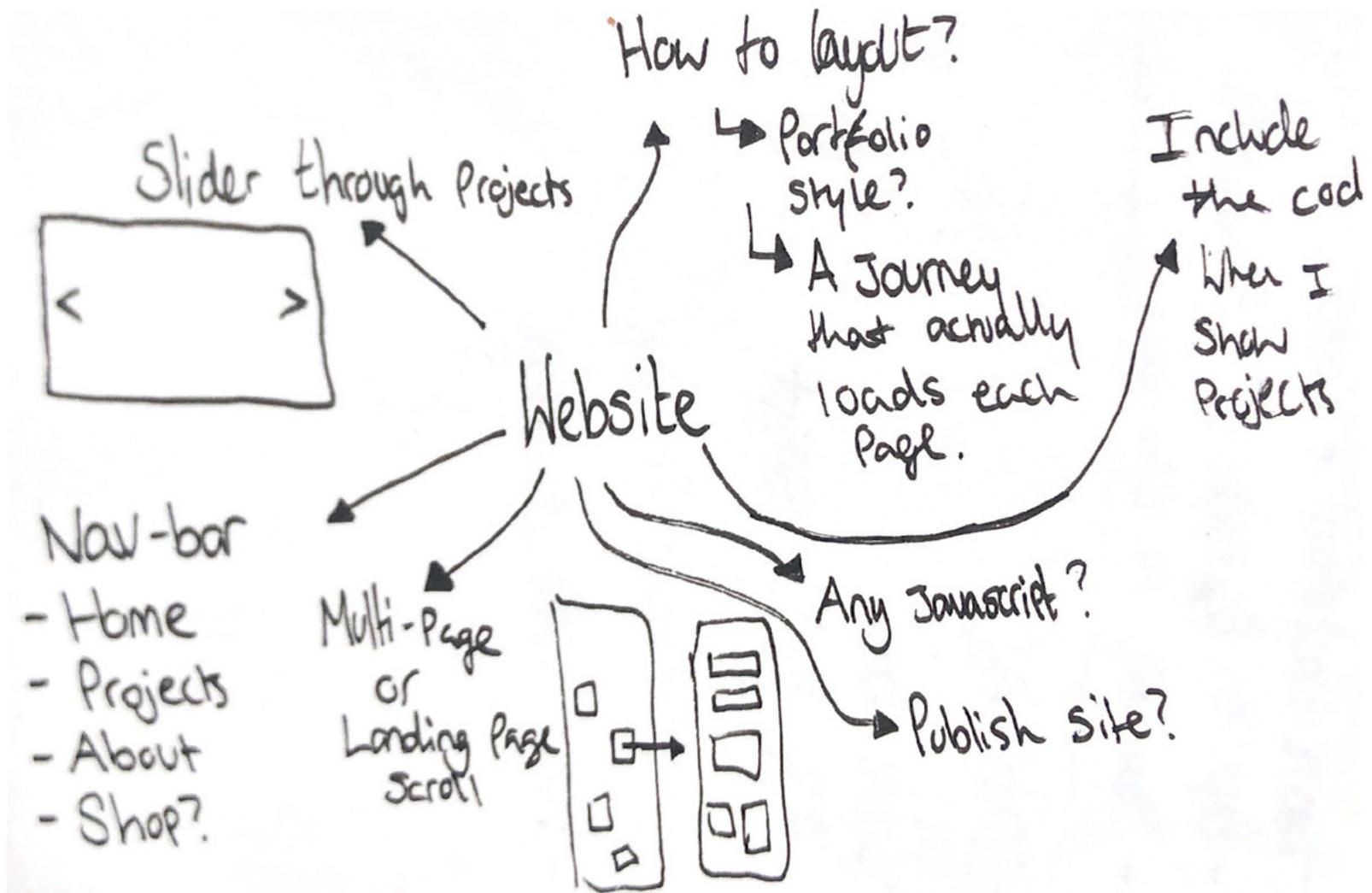


The page features a large green rectangular image at the top left with the text 'DEADSTOCK BAGGU' and a smaller image of a person wearing a colorful apron. Below these are two square images: one of a woman wearing a large, patterned sunhat, and another showing several colorful reusable bags hanging against a yellow background. At the bottom, there's a horizontal row of four smaller images showing different ways to use the bags.

The screenshot shows the Harvest website's homepage. At the top, there's a navigation bar with links for 'About Harvest', 'Features', 'Customers', 'Integrations', and 'Pricing'. On the far right, there are 'Sign In' and a 'Get Started Free' button. The main headline reads 'Elevate your time tracking' with a subtext 'Simple time tracking software and powerful reporting that helps your team thrive.' Below this is a large orange 'Get Started' button. To the right of the headline is a stylized illustration of a hand holding a smartphone displaying a clock, with a laptop, a tree, and a bar chart in the background. A small note below the button says 'Free 14-day trial. No credit card required.' Further down, there's a section titled 'How Harvest can help you work faster' with icons for 'Invoicing', 'Time tracking', 'Reports & analysis', and 'Integrations'. Each icon has a brief description below it. At the bottom, there's a decorative footer featuring a stylized flower or gear design.

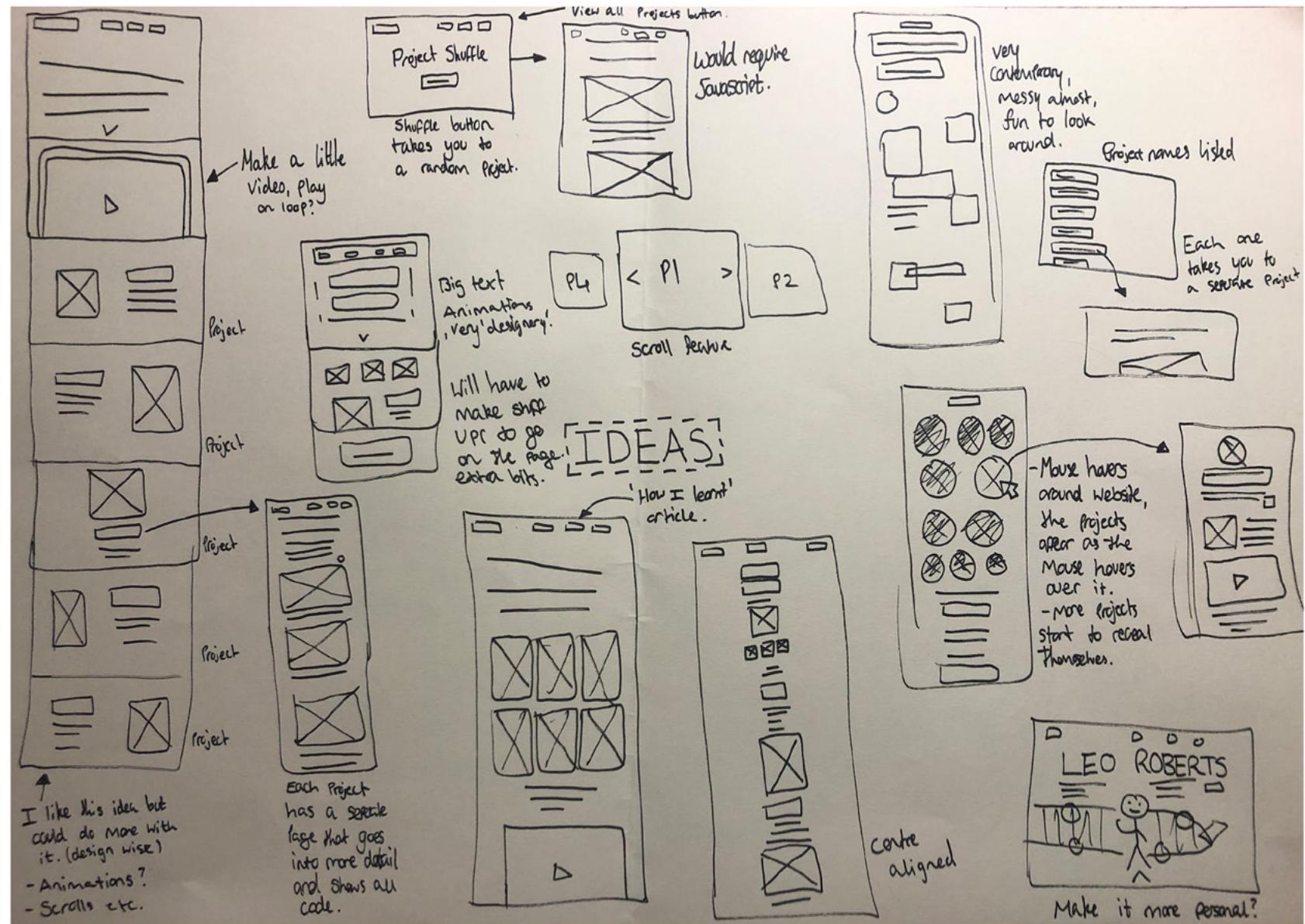


Final Outcome

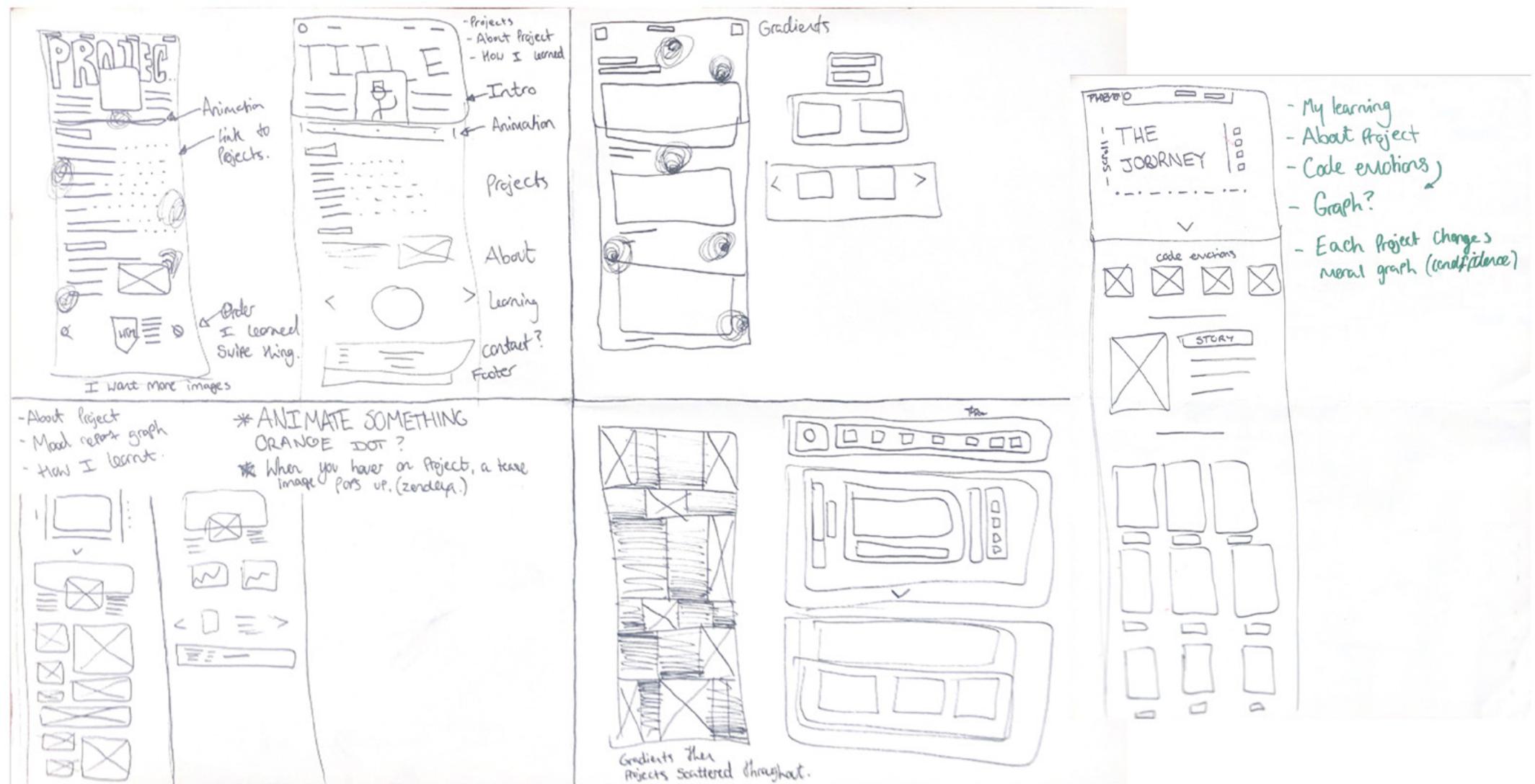


Final Outcome sketches

Want to do website for final outcome, it only makes sense to code somethin , considering that's what this project has been about.



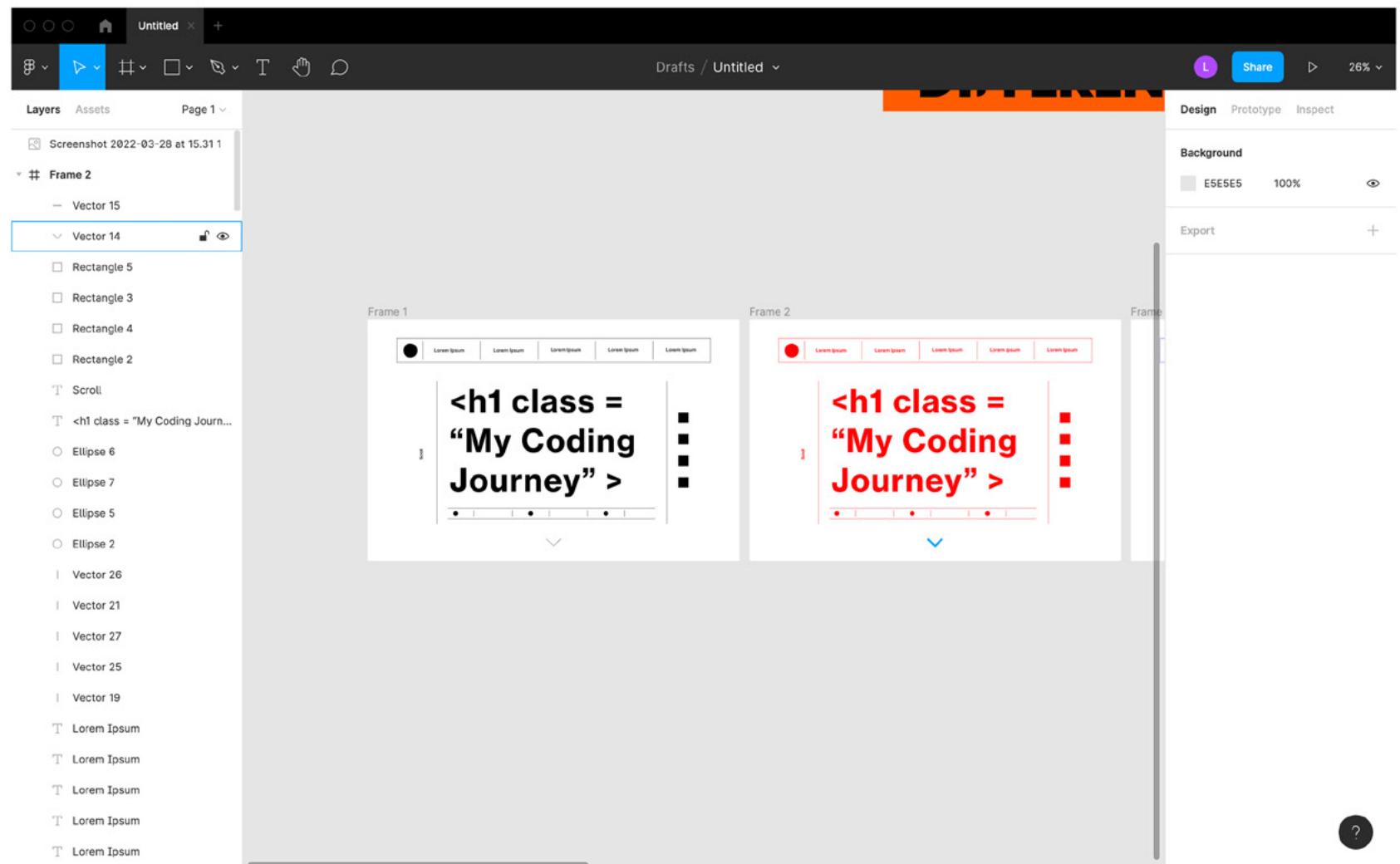
Sketches



Figma.

I have experience in both Adobe XD and Figma for wire framing websites, I chose to use Figma because I find it's slightly more user friendly and I enjoy the animation features.

I'm going to roughly mock-up my website before I start coding it. This will make it much easier for me to visualize it and bring it to life through code. If I didn't do this, I would be designing the website for the first time through code, I think that would be challenging.



Final Outcome

Some landing page mock-ups made on Figma. I really like this minimal style and I think it suits the theme of the project well. This is a mock-up so I will probably end up changing fonts etc. but I think this is the style I want to go with. These are some different colour schemes I've been thinking of. I like the simplicity of having a 2 colour scheme, but I also like the idea of using the same colours of my code, for the website. (Fig 1)

```
<div class="mid-news">  
    <div class="left-2">  
        <h1><strong>WhatsApp</strong></h1>  
        <p class="para"> Code v  
        <button class="button-3">  
            <!--Change to a button  
        </button>  
    </div>  
  
    <div class="right-2">  
          
    </div>
```

(Fig 1)

The image displays six Figma mock-up screens arranged in a grid, illustrating different color schemes for a landing page. Each screen features a header with five 'Lorem Ipsum' placeholder cards and a main content area with a large 'h1' heading and a horizontal navigation bar.

- Top Left:** Black header, black 'h1' text ('My Coding Journey'), black navigation dots, black vertical scroll bar.
- Top Right:** Red header, red 'h1' text ('My Coding Journey'), red navigation dots, red vertical scroll bar.
- Middle Left:** Blue header, blue 'h1' text ('My Coding Journey'), blue navigation dots, blue vertical scroll bar.
- Middle Right:** Orange header, orange 'h1' text ('My Coding Journey'), orange navigation dots, orange vertical scroll bar.
- Bottom Left:** Dark green header, dark green 'h1' text ('My Coding Journey'), dark green navigation dots, dark green vertical scroll bar.
- Bottom Right:** White header, white 'h1' text ('My Coding Journey'), white navigation dots, white vertical scroll bar.

Each screen includes a small 'Scroll' indicator at the bottom left and a downward arrow at the bottom center.

Final Outcome experimentation

More mockups on Figma that demonstrate how my final outcome could look.

I'll hand in these mockups in video format aswell to show a better example of how they would work.

Could I link this page and the ones reviously shown? so theres a 'projects section' on the website. I would have to change the design to keep it consistent, and also have a navigation system. (talking about fig.1)

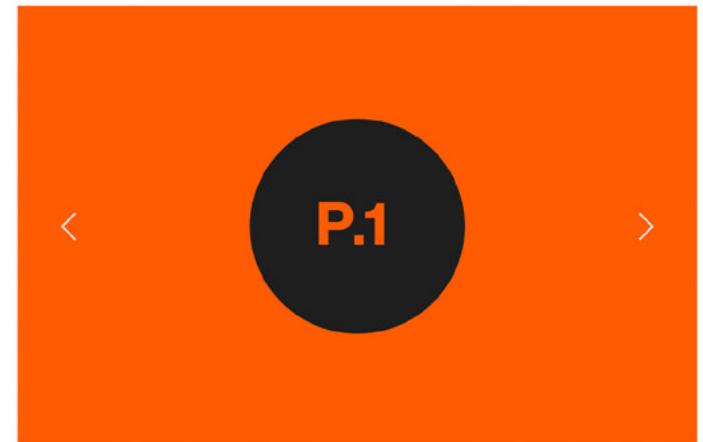
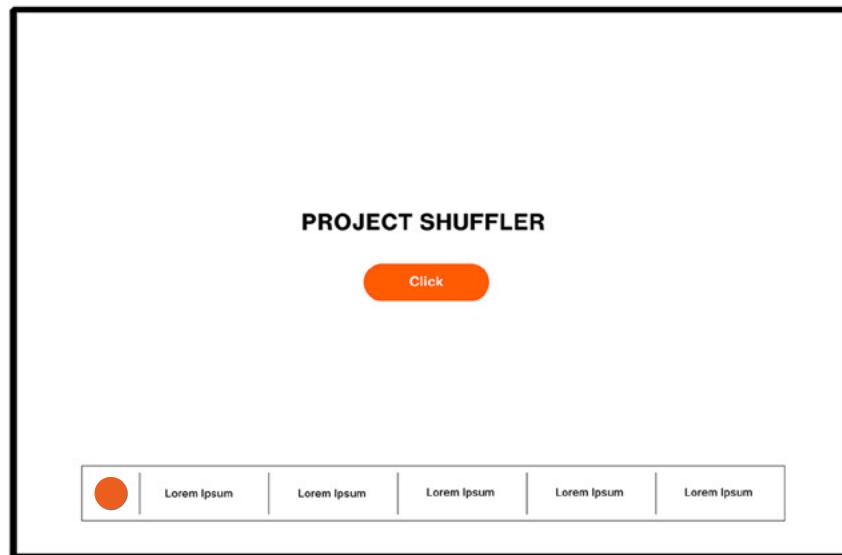
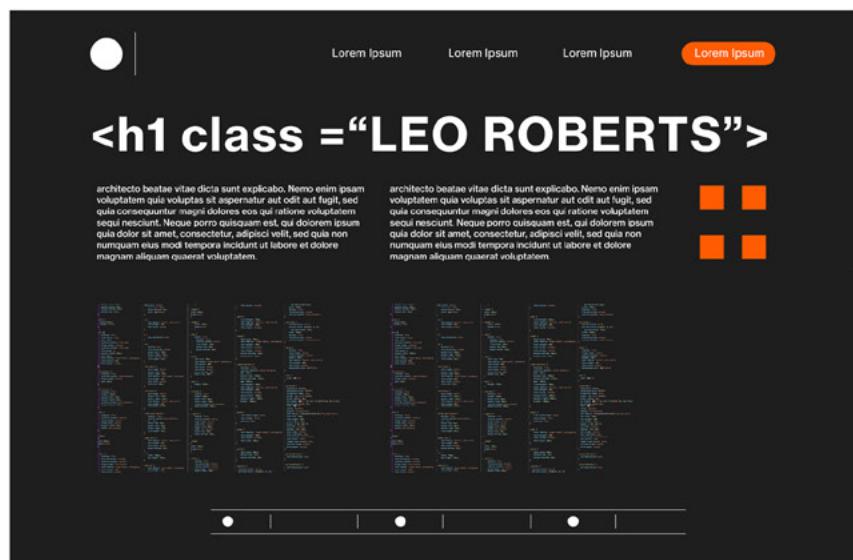
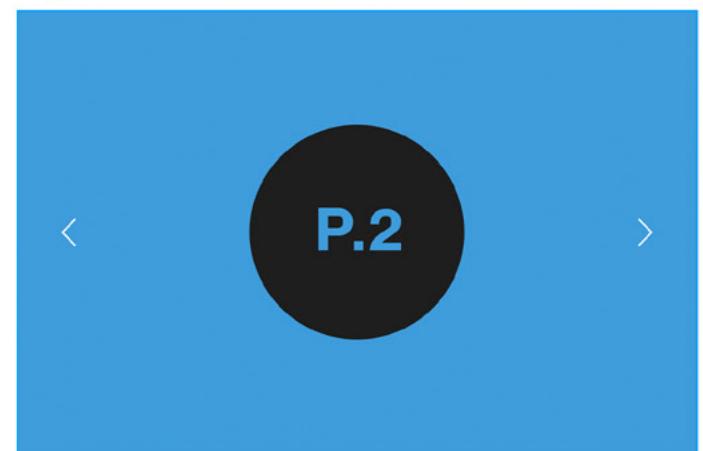


fig 1

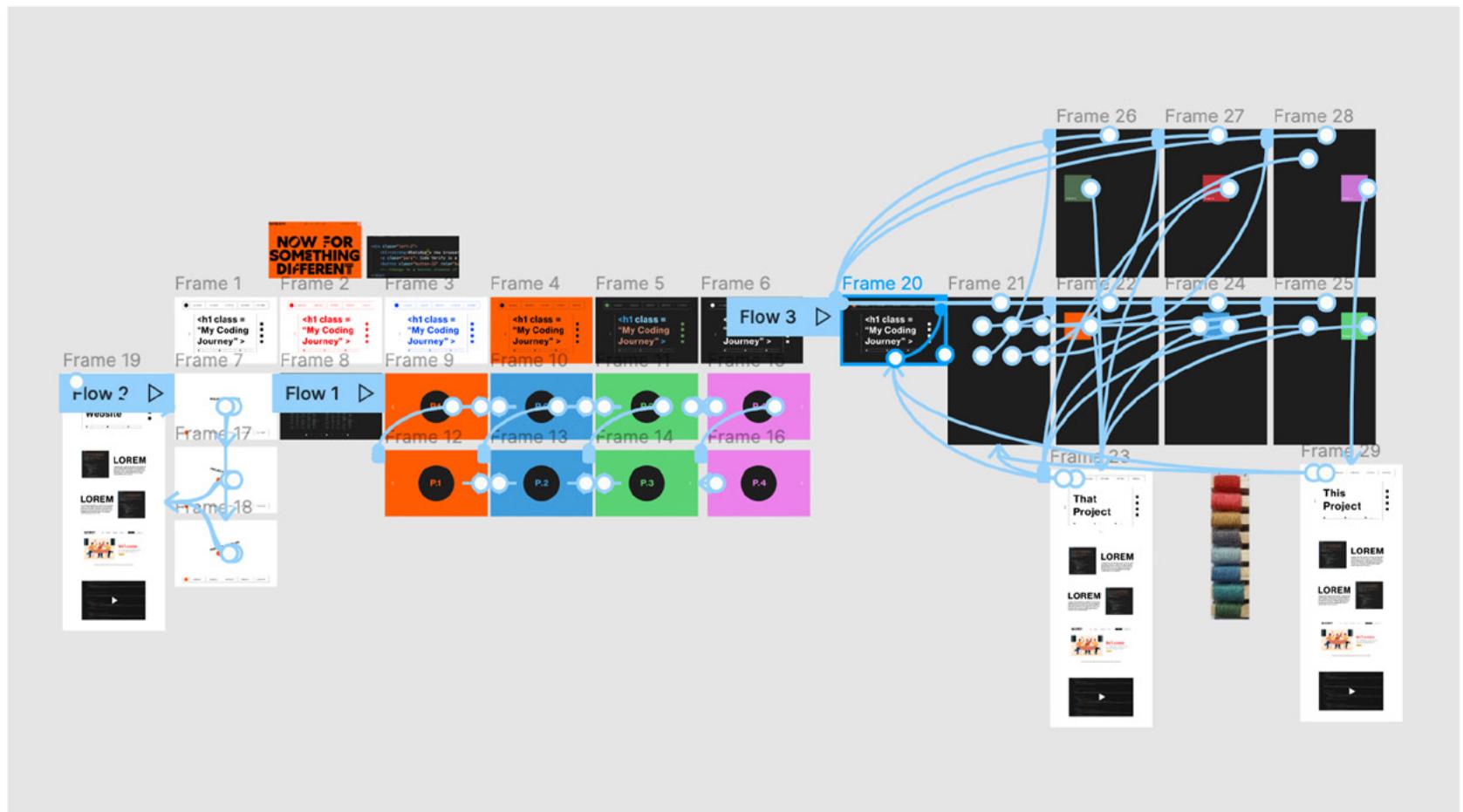


Scroll through pages with arrows.



My Figma board

Example of how i've prototyped the websites using wireframes that can give me a real insight into how I want the different elements of my website to come together and look.



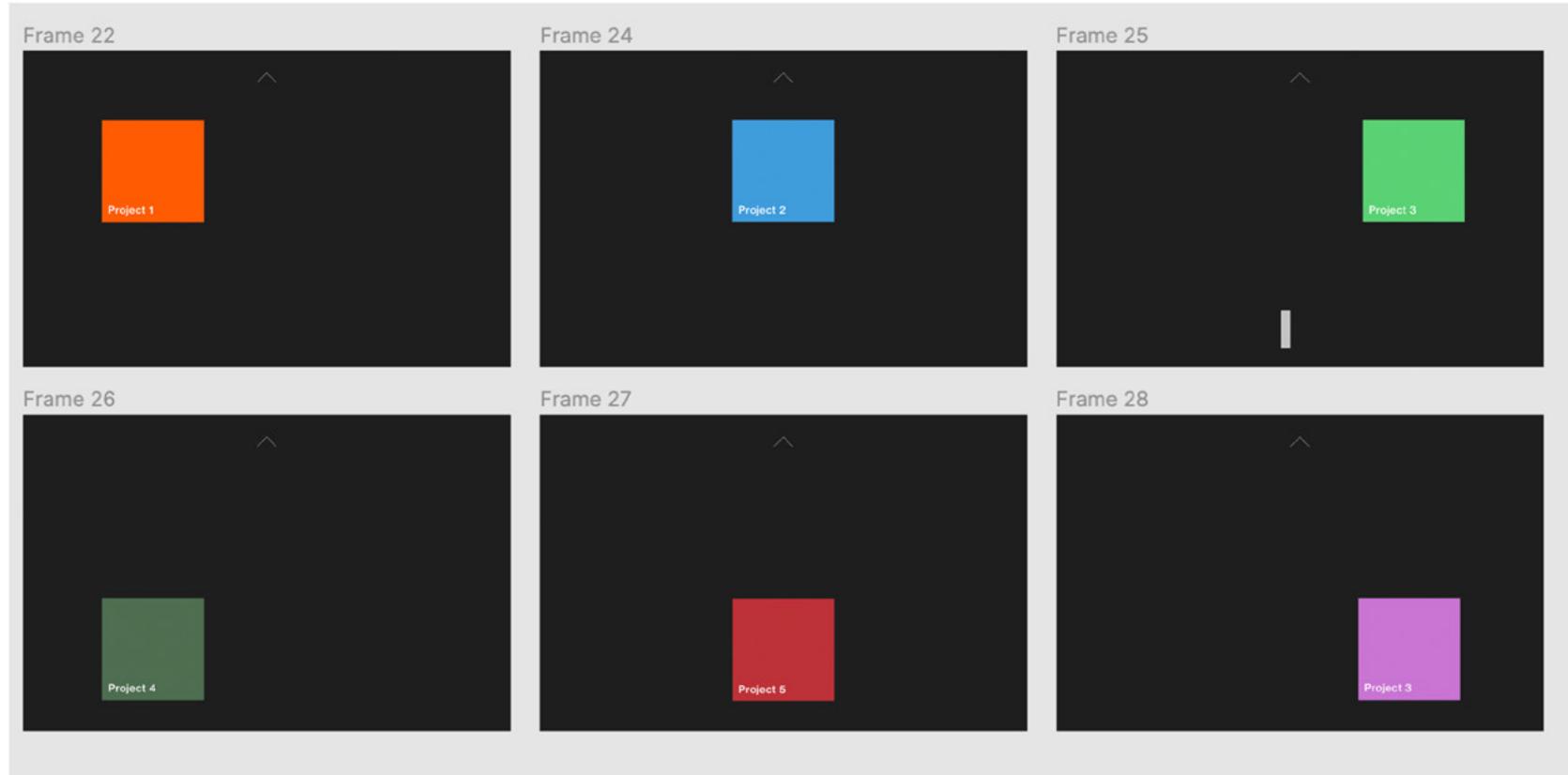
Final Outcome experimentation

Black screen but when you hover the mouse over certain areas, squares pop up that can be clicked on to be taken to a particular project.

Will have a video showing how this looks when prototyped.

Orange dot acts as a home button.

Projects tab takes you back to the project screen.



Physical Moodboard for Final Outcome

Playing with potential layouts, colours, fonts and patterns for final outcome.



Final Outcome

Thought

At the moment, the website would just be showing all of my projects in order, basically a portfolio site.

I want there to be a more personal focus, a focus on the 'journey'. How I went from no coding experience to be able to build websites, how I was feeling throughout the projects, what I was learning specifically etc. I think leaning into this approach will allow for more content on the site as well as a more meaningful final outcome.



Final Outcome experimentation

I think this layout I made on figma is what i'm going to try and code for my final outcome. I wanted something fun that looked Professional design wise. I needed it to be simple enough for me to code it well but still have challenging elements and room to learn while building. With this design, I'm not committing to one idea, I can always add or get rid of elements as I code.

What I've made on figma is very rough and would look better when I add the real content.

I can also use ideas from previous mock-ups to influence this one.

The wireframe shows a clean layout with a large, bold title 'JOURNEY'. Below the title is a central image of a shirtless man with his arms raised, standing on a beach. The image is framed by a red gradient background. To the left and right of the image are two columns of placeholder text ('Lorem ipsum'). At the bottom are two downward-pointing chevrons.

The polished wireframe adds a header with five placeholder cards labeled 'Lorem Ipsum'. The main content area features a large 'JOURNEY' title and a central image of a shirtless man. The footer includes a downward chevron.

PROJECTS

LALA P1

DIDI P2

RAJI P3

LOPO P4

CHASI P5

TITIP P6

JAMA P7

GREY P8

LOAP P9

ABOUT THE PROJECT

The wireframe for the 'ABOUT THE PROJECT' section includes a code editor window showing HTML and CSS code, and a text area with placeholder text.

JOURNEY

The wireframe for the 'JOURNEY' section features two line graphs on the left and right, and a large 'HTML 5' logo on the right side.

Final Outcome experimentation

More layout developments from my previous designs.

I wanted to come up with a few variations of the initial layout. This will help me with experimentation, seeing what looks best, and gives me multiple options when it comes to coding.



Frame 31

Frame 34

<h1 class = "My Coding Journey">

PROJECTS

LALA P1
DIDI P2
RAJI P3
LOPO P4
CHASI P5
TITIP P6
JAMA P7
GREY P8
LOAP P9

JOURNEY

HTML5

Property of Leo Roberts

<h1 class = "My Coding Journey">

PROJECTS

JOURNEY

HTML5

Property of Leo Roberts

Initial HTML

Initial html of final outcome. The top picture is my first attempt before I decided to change the design and layout.

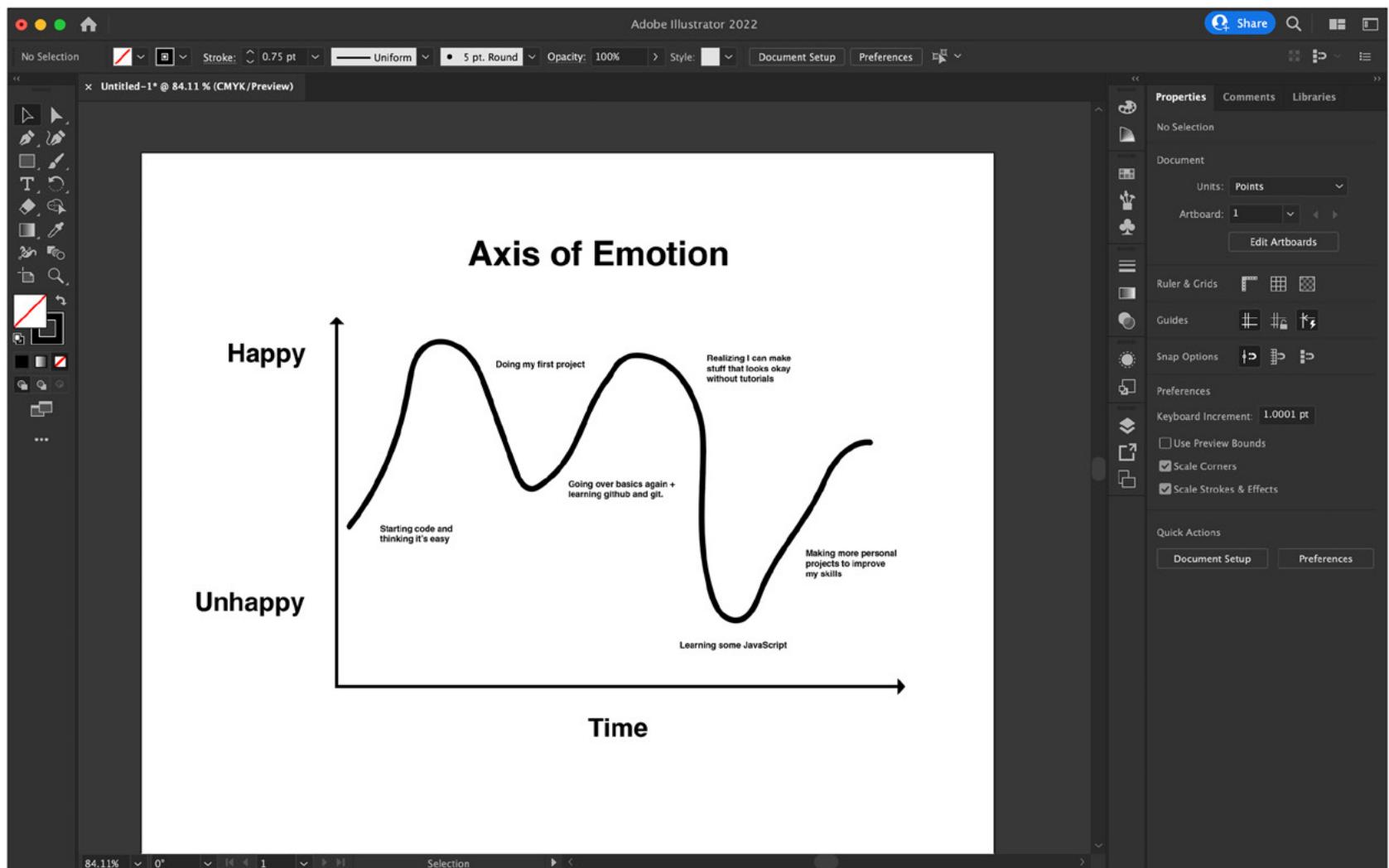
The bottom shows the skeleton of the website without any design elements added.

The screenshot displays the initial HTML structure of a website. At the top, there's a navigation bar with a dropdown menu containing 'Projects', 'Randomizer', and 'About Project'. Below the navigation is a large **<h1> My Coding Journey </h1>**. Underneath the title are two paragraphs of placeholder text ('Lorem ipsum' and 'Iste ex'). A 'PROJECTS' section follows, featuring a numbered list from 1 to 9. Below that is an 'ABOUT' section with another paragraph of placeholder text. At the bottom, there's a contact section for 'Leo Roberts' with 'contact info etc.' and a footer area containing four 'Lorem' placeholder boxes.

This screenshot shows the skeletal structure of the website. It includes the header, main content area with the title and placeholder text, the projects section with numbered items, the about section, and the footer. The contact information for 'Leo Roberts' is also present. The entire layout is defined by simple rectangular boxes and lines, representing the raw HTML structure without CSS styling.

Website asset

Made a graph documenting how my mood towards coding changed throughout the project. This will be a part of my final outcome.



Final Outcome

This is how the landing page currently looks, it's hard to show in my sketchbook but I will be handing in a video that shows all aspects of the website + the code.

I wanted the website to be simple enough to code, but still look stylish and include lots of different things I've learnt, for example, flex box, animation (moving bars) and scrolling.

I also wanted it to have a more personal touch, hence the picture of me and the 'approach section' that shows how I found the project.

The screenshot shows a landing page with a light gray header bar containing a large orange circle icon and five navigation links: 'PROJECTS', 'SHUFFLE', 'APPROACH', and 'CONTACT ME'. The main title 'My Coding Journey' is displayed in a large, bold, black font. Below the title is a smaller orange bar with the word 'WELCOME' repeated five times. To the left of the title is the author's name, 'LEO ROBERTS'. To the right are three social media icons: Instagram, LinkedIn, and GitHub. A large, bold, black 'PROJECTS' heading is centered at the bottom. At the very bottom, there is a small image of a person's head and shoulders, and two columns of text: one about the website's purpose and another about learned skills.

PROJECTS

LEO ROBERTS

WELCOME • WELCOME • WELCOME • WELCOME • WELCOME

Instagram LinkedIn GitHub

This website is the result of an overarching project that documents my learning of coding. I wanted to learn HTML, CSS, and JavaScript. When I took on this



I ended up learning a lot more than that however, things like Git, GitHub, VisualStudioCode and a brief knowledge on front end libraries such as Bootstrap,

Final Outcome

challenge, I had an interest in web design, but I had no idea what to expect when it came to code. Now I feel confident in my understanding of HTML and CSS and have a basic understanding of JavaScript.



Sass and React seemed to be part of my natural journey as a web developer. Below is a chronological list of all the projects that I have done in order to develop my skills to where they are now.

Tribute Page

Survey Form

Product Page

Documentation Page

Recipe Website

Flexbox Challenges

Landing Page

Weather App

News Website

Grid Projects

Final Outcome

APPROACH • APPROCH • APPROACH

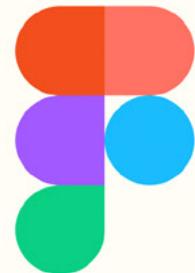
I had a lot of ups and downs throughout this project, learning something new always brings new challenges in which I enjoyed. The information below gives you an insight into how I approached learning to code and how my feelings developed throughout.



HOW I LEARNT

What I have been learning, in chronological order.

Final Outcome



HTML



CSS



LEO ROBERTS! - Celebrate Crafts Brief - 2022

Final Outcome code (so far)

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <link rel="stylesheet" href="style.css">
8      <title>Final Outcome</title>
9  </head>
10 <body>
11 
12     <div id="landing-link1"></div>
13 
14 
15     <div class="nav-bar">
16         <a class="nav-left" href="#landing-link1"></a>
17 
18         <div class="nav-right">
19 
20             <HR class="top-lines">
21             <a href="#"><li>PROJECTS</li></a>
22             <a href="#"><HR class="top-lines">
23                 <a href="#"><li>SHUFFLE</li></a>
24             <HR class="top-lines">
25                 <a href="#"><li>APPROACH</li></a>
26             <HR class="top-lines">
27                 <a href="#"><li>CONTACT ME</li></a>
28 
29 
30         </div>
31     </div>
32 
33 
34 
35     <div class="landing">
36         <p class="Leo">LEO ROBERTS</p>
37         <HR class="landing-lines">
38         <div class="mid-sec">
39             <h1> &lt;h1&gt; My Coding Journey &lt;/h1&gt; </h1>
40             <!--Add Scroller here-->
41             <div class="scroll-slow">
42                 <p>WELCOME ● WELCOME ● WELCOME ● WELCOME ● WELCOME ● WELCOME ● </p>
43             </div>
44             <a href="#landing-link2"></a>
45         </div>
46         <HR class="landing-lines">
47         <div class="socials">
48             <a href="https://www.instagram.com/splitwaves_/"></a>
49             <a href="https://www.linkedin.com/in/leo-roberts-22965615b/"></a>
50             <a href="https://github.com/LeoRoberts21"></a>
51         </div>
52     </div>
53 
54 </body>
55 
56 <div id="landing-link2"></div>
57 
58 <div class="projects">
59     <h2>PROJECTS</h2>
60     <div class="me">
61         <p class="selftext">This website is the result of an overarching project that docu...
62         
63         <p class="selftext">I ended up learning a lot more than that however, things like ...
64     </div>
65 
66     <div class="project-list">
67         <a href="#"><p>Tribute Page</p> </a>
68             | <HR class="project-lines">
69         <a href="#"><p>Survey Form</p> </a>
70             | <HR class="project-lines">
71         <a href="#"><p>Product Page</p> </a>
72             | <HR class="project-lines">
73         <a href="#"><p>Documentation Page</p> </a>
74             | <HR class="project-lines">
75         <a href="#"><p>Recipe Website</p> </a>
76             | <HR class="project-lines">
77         <a href="#"><p>Flexbox Challenges</p> </a>
78             | <HR class="project-lines">
79         <a href="#"><p>Landing Page</p> </a>
80             | <HR class="project-lines">
81         <a href="#"><p>Weather App</p> </a>
82             | <HR class="project-lines">
83         <a href="#"><p>News Website</p> </a>
84             | <HR class="project-lines">
85         <a href="#"><p>Grid Projects</p> </a>
86             | <HR class="project-lines">
87     </div>
88 
89     <div class="journey1">
90         <div class="scroll-fast">
91             <p>APPROACH ● APPROACH ● APPROACH ● APPROACH</p>
92         </div>
93         <h3></h3>
94         <p class="journey2">I had a lot of ups and downs throughout this project, learning ...
95     </div>
96 
97     <div class="journey3">
98         
99         
100    </div>
101 
102    <div class="journey-scroll">
103        <!--Add language Scroller here-->
104    </div>
105 
106    <h4>HOW I LEARN</h4>
107    <p class="what">What I have been learning, in chronological order.</p>
108 
109    <div class="horizontal-snap">
110        
111        
112        
113        
114        
115        
116        
117        
118    </div>
119 
120    <p class="right-scroll"></p>
121 
122 </body>
123 
124 <footer>
125     <p class="footer-text">LEO ROBERTS! - Celebrate Crafts Brief - 2022</p>
126 </footer>
127 
128 </html>
```

html (everything pushed to GitHub)

Final Outcome code (so far)

```
1  /+BREAK BREAK BREAK BREAK BREAK */
2
3  body {
4    | background-color: ■#ffffd5;
5  }
6
7
8
9
10 .nav-bar {
11   display: flex;
12   justify-content: space-between;
13   margin-left: 50px;
14   margin-right: 50px;
15   margin-bottom: 50px;
16   margin-top: 50px;
17   align-items: center;
18   border-style: solid;
19   border-width: 1px;
20   position: -webkit-sticky;
21   position: sticky;
22   top: 0;
23   background-color: ■#ffffd5;
24   font-family: Arial, Helvetica, sans-serif;
25   font-weight: bold;
26 }
27
28 .nav-right {
29   display: flex;
30   list-style: none;
31   gap: 20px;
32   justify-content: space-between;
33   align-items: center;
34   margin-right: 100px;
35   gap: 50px;
36   width:1000px;
37 }
38
39 .nav-left {
40   display: flex;
41   justify-content: center;
42   flex-grow: 1;
43   height: 100px;
44   width: 100px;
45 }
46
47 .top-lines {
48   border: 0;
49   clear:both;
50   display:block;
51   width: 1px;
52   background-color:■#000000;
53   height: 75px;
54   margin-left: 0px;
55   margin-right:0px;
56   margin-top: 10px;
57   margin-bottom: 10px;
58 }
59
60
61
62 /*BREAK BREAK BREAK BREAK BREAK */
63
64
65
66 .landing {
67   display: flex;
68   justify-content: space-between;
69   margin-left: 50px;
70   margin-right: 50px;
71   align-items: center;
72   font-family: Arial, Helvetica, sans-serif;
73   margin-top: 100px;
74 }
75
76 .landing-lines {
77   border: 0;
78   clear:both;
79   display:block;
80   width: 2px;
81   background-color:■#000000;
82   height: 500px;
83   margin-left: 0px;
84   margin-right:0px;
85 }
86
87 .leo {
88   -webkit-transform: rotate(-90deg);
89 }
90
91 .socials {
92   display: flex;
93   flex-direction: column;
94   align-items: center;
95   justify-content: center;
96   margin-right: 100px;
97   width: 40px;
98   height: 40px;
99   border-radius: 20px;
100 gap: 30px;
101 }
102
103 .down {
104   max-width: 50px;
105   height: 50px;
106   left: 0px;
107   right: 0px;
108   margin-top: 100px;
109   margin-left: 325px;
110 }
111
112 h1 {
113   font-size: 90px;
114   text-align: center;
115   width:700px;
116 }
117
118 .mid-sec {
119   display: flex;
120   align-self: flex-start;
121   justify-content: center;
122   flex-direction: column;
123 }
124
125 .scroll-slow {
126   height: 50px;
127   width: 700px;
128   overflow: hidden;
129   position: relative;
130   background: ■#ED662C;
131   color: ■rgb(0, 0, 0);
132   border: 2px solid ■#ED662C;
133 }
134
135 .scroll-slow p {
136   position: absolute;
137   width: 100%;
138   margin: 0;
139   line-height: 50px;
140   text-align: center;
141 }
142
143 -moz-transform:translateX(100%);
144 -webkit-transform:translateX(100%);
145 transform:translateX(100%);
146
147 -moz-animation: scroll-slow 12s linear infinite;
148 -webkit-animation: scroll-slow 12s linear infinite;
149 animation: scroll-slow 12s linear infinite;
150
151 @-moz-keyframes scroll-slow {
152   0% { -moz-transform: translateX(100%); }
153   100% { -moz-transform: translateX(-100%); }
154 }
155 @-webkit-keyframes scroll-slow {
156   0% { -webkit-transform: translateX(100%); }
157   100% { -webkit-transform: translateX(-100%); }
158 }
159 @keyframes scroll-slow {
160   0% {
161     -moz-transform: translateX(100%);
162     -webkit-transform: translateX(100%);
163     transform: translateX(100%);
164   }
165   100% {
166     -moz-transform: translateX(-100%);
167     -webkit-transform: translateX(-100%);
168     transform: translateX(-100%);
169   }
170 }
171
172 /*BREAK BREAK BREAK BREAK BREAK */
173
174 h2 {
175   display: flex;
176   margin-top: 100px;
177
178 font-family: Arial, Helvetica, sans-serif;
179   font-size: 24px;
180   margin-bottom: 15px;
181   width: 100px;
182   justify-content: center;
183
184 }
185
186 .me {
187   display: flex;
188   justify-content: space-between;
189   flex-direction: row;
190   margin-left: 50px;
191   margin-right: 50px;
192   gap: 50px;
193   font-family: Arial, Helvetica, sans-serif;
194 }
195
196 .self {
197   width: 400px;
198   height: 40px;
199   border-radius: 20px;
200 }
201
202 .selftext {
203   font-size: 17px;
204   line-height: 25px;
205   /*for editing the text next to my picture */
206
207 }
208
209 /*BREAK BREAK BREAK BREAK BREAK */
210
211 .project-list {
212   margin-left: 50px;
213   margin-right: 50px;
214   margin-top: 100px;
215   font-family: Arial, Helvetica, sans-serif;
216   font-size: 40px;
217 }
218
219
220 .project-lines {
221   border-top: 2px dotted;
222   margin: 0;
223 }
224
225 .project-list {
226   font-weight: bolder;
227   animation: scroll-slow 12s linear infinite;
228   line-height: 5px;
229 }
230
231 a:link {
232   color: ■rgb(0, 0, 0);
233   text-decoration: none;
234 }
235 a:hover {
236   color: ■#ED662C;
237 }
238
239
240 /*BREAK BREAK BREAK BREAK BREAK */
241
242 .journey1 {
243   display: flex;
244   justify-content: center;
245   align-items: center;
246   margin-top: 100px;
247   flex-direction: column;
248   font-family: Arial, Helvetica, sans-serif;
249 }
250
251
252 /*BREAK BREAK BREAK BREAK BREAK */
253 .journey2 {
254   margin-left: 450px;
255   margin-right: 450px;
256   font-family: Arial, Helvetica, sans-serif;
257   font-size: 17px;
258   line-height: 25px;
259 }
260
261 .journey3 {
262   display: flex;
263   justify-content: center;
264   gap: 50px;
265 }
266
267 .g1 {
268   height: 600px;
269   width:600px;
270 }
271
272 .g2 {
273   height: 600px;
274   width: 600px;
275 }
276
277 footer {
278   background-color: ■#ED662C;
279   height:100px;
280   display:flex;
281   justify-content: center;
282   align-items: center;
283   margin-top: 100px;
284   font-size: 50px;
285 }
286
287 .horizontal-snap {
288   margin: 0 auto;
289   display: grid;
290   grid-auto-flow: column;
291   gap: 1rem;
292   height: 380px;
293   padding: 1rem;
294   max-width: 1000px;
295   overflow-y: auto;
296   scroll-snap-type: x mandatory;
297   background-color: ■#ffffd5;
298 }
299
300 .horizontal-snap > a {
301   scroll-snap-align: center;
302 }
303
304 .horizontal-snap img {
305   width: 300px;
306   max-width: none;
307   object-fit: contain;
308   border-radius: 1rem;
309 }
310
311 .right-scroll {
312   font-family: Arial, Helvetica, sans-serif;
313   font-weight: bolder;
314   font-size: 40px;
315   text-align: center;
316   -moz-transform:translateX(100%);
317   -webkit-transform:translateX(100%);
318   transform:translateX(100%);
319
320 -moz-animation: scroll-fast 12s linear infinite;
321   -webkit-animation: scroll-fast 12s linear infinite;
322   animation: scroll-fast 12s linear infinite;
323 }
324
325 @-moz-keyframes scroll-fast {
326   0% { -moz-transform: translateX(100%); }
327   100% { -moz-transform: translateX(-100%); }
328 }
329 @-webkit-keyframes scroll-fast {
330   0% { -webkit-transform: translateX(100%); }
331   100% { -webkit-transform: translateX(-100%); }
332 }
333 @keyframes scroll-fast {
334   0% {
335     -moz-transform: translateX(100%);
336     -webkit-transform: translateX(100%);
337     transform: translateX(100%);
338   }
339   100% {
340     -moz-transform: translateX(-100%);
341     -webkit-transform: translateX(-100%);
342     transform: translateX(-100%);
343   }
344
345 .horizontal-snap {
346   margin: 0 auto;
347   display: grid;
348   grid-auto-flow: column;
349   gap: 1rem;
350   height: 380px;
351   padding: 1rem;
352   max-width: 1000px;
353   overflow-y: auto;
354   scroll-snap-type: x mandatory;
355   background-color: ■#ffffd5;
356 }
357
358 .horizontal-snap > a {
359   scroll-snap-align: center;
360 }
361
362
363 .horizontal-snap img {
364   width: 300px;
365   max-width: none;
366   object-fit: contain;
367   border-radius: 1rem;
368 }
369
370
371 h4 {
372   font-family: Arial, Helvetica, sans-serif;
373   font-weight: bolder;
374   font-size: 40px;
375   text-align: center;
376   margin-bottom: 25px;
377 }
378
379 .right {
380   font-family: Arial, Helvetica, sans-serif;
381   font-weight: bolder;
382   font-size: 60px;
383   text-align: center;
384 }
```

Final Outcome

These are some of the project pages. They show each mini project I have done throughout the duration of this brief.

When you load into the page, you can only see the title and the write up. Scroll down and there is a window which allows the user to scroll up and down my project and see it all. Then at the bottom is all my code, this can be scrolled through aswell .

Recipe Website

The Burger consists of two 1.6 oz (45.4 g) beef patties, iceberg lettuce, American cheese, pickles, chopped onions and special Mac Sauce (a Thousand Island dressing variant, though the recipe remains a secret) served on a three part sesame seed bun.

Ingredients

- Buns
- Burger Patties
- Tomatoes

HTML

CSS

LEO ROBERTS - Celeste Crafts Brief - 2022

News Website

We are a company committed to helping you, the latest news, most recipes, how-to guides and how they affect culture, the economy, and politics.

Welcome

HTML

CSS

LEO ROBERTS - Celeste Crafts Brief - 2022

Survey Form

This website is the result of an intensive project that documents my learning of how to interact with HTML, CSS, and JavaScript. While I am still a beginner, I feel confident in web design, but I feel no less able to express what I have to code. I have now learned how to use the validation in JavaScript correctly, and I understand of JavaScript.

I ended up learning a lot more than the brief. From the UI, GitHub, and documentation, I learned a lot about development in front end development as well. I feel like I have learned a lot about myself as a developer. I have now learned how to use the validation in JavaScript correctly, and I understand of JavaScript.

HTML

CSS

Box for Large Chunks of Text or How to Create a Floating Text Box

LEO ROBERTS - Celeste Crafts Brief - 2022

Final Outcome

This was the first project I did without a tutorial or any form of guidance. It was a brief set by [Freecodecamp.com](#), the aim was to make a tribute page for someone famous, write about them and list their achievements. I found it very hard and it took me multiple hours to figure out. In the end I had added a title, a picture, a list and some text. None of it aligned. It looked terrible but it introduced me to coding. I wasn't proud of it at the time but it gave me an insight into how I could use html and css in the future to bring my designs to life.

This was another [Freecodecamp](#) project. I had to build a survey form for a random topic. I made one that let you pick options and customise a flamethrower as if you were about to order one. This was another hard project at the time and it introduced me to lots. I was able to use css to colour the background and text but still was unable to arrange stuff on the page. I also learnt how to do radio buttons, checkbox, text box, dropdown menus and buttons. Overall it was a really helpful project.

Another [Freecodecamp](#) project. This one taught me how to embed videos(even though that didn't work) and create empty links for my text. I also managed to do some styling with css, I created a very basic nav-bar above the image. This project was very tricky and I found laying it out really hard, it was all over the place for a while but I went back to it a few days later and fixed it up. It's still nothing special but it was progress in the right direction and out of everything I'd made so far, this looked the most like an actual website.

This task was to create a document page, I chose to do it on f1 teams. It looks very basic and I didn't use any css styling on this project, it was more about the html. I managed to implement links within the page, this meant when you click on one of the team names at the top, the page will jump down to the information about that specific team. This skill is very useful to know and I don't remember finding this project very hard. By this point I was getting the hang of html but still struggled with css a lot.

The recipe website was the first project I did for The Odin Project course. This course really sped up my learning, the recipe page taught me more about css styling mainly but I also had a better idea of layout (minus that burger). I was using Visual Studio Code by this point. This project meant I had to learn how to link css pages to html along with external html pages. This was the point where I really could imagine making websites and I could see clear progress when looking back at my projects.

This was a flex-box project that I also found through The Odin Project. I had spent so long trying to understand how to properly lay out everything on a website, and when I was introduced to flex-box it only confused me more. This project was so satisfying because it felt like everything I had been learning suddenly clicked and I was able to lay it all out on the page and style it in a way that kind of looked like a real website. This gave me a lot of confidence going forward, I felt like I could actually code.

My first JavaScript project was a website that asks for the users location and then fetches weather updates from an API to show a live weather reading. This website was meant to show, timezone, temperature and a comment about how cold or hot it was. Nothing worked. It was just a static webpage. The only thing my Javascript did was add a pop up requesting users location. This really knocked my confidence as I was struggling a lot with Javascript and knew I would need a lot more practise.

After I failed at my first Javascript project, I needed a confidence boost. So I decided to build another website using html and css. It was gonna be a news website and I wanted it to have a lot more design elements than my previous work. I visualized on paper what my layouts would look like before coding, and then added new fonts, styled buttons, more colour, pictures and patterns to it. Even the buttons had click animations! I think it turned out really nice.

This project was made after spending a long time learning how to use the grid system in css. Using grids allows you to out responsive layouts on your website and it's a really useful thing to know. I found (and still find) the grid system in css a bit confusing but I made this without too much trouble, I think a more complex layout would trouble me though. I had been learning how to make pages responsive using Bootstrap as well by this point.

News Website

After I failed at my first Javascript project, I needed a confidence boost. So I decided to build another website using html and css. It was gonna be a news website and I wanted it to have a lot more design elements than my previous work. I visualized on

paper what my layouts would look like before coding, and then added new fonts, styled buttons, more colour, pictures and patterns to it. Even the buttons had click animations! I think it turned out really nice.

Final Outcome (code for project page)

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>News Website Project</title>
8   <link rel="stylesheet" href="style2.css">
9 </head>
10 <body>
11
12   <div id="landing-link1"></div>
13
14
15   <div class="nav-bar">
16     <a class="nav-left" href="#">![home-button](ad.png)PROJECTS</a>
21       <a href="#">SHUFFLE</a>
22       <a href="#">APPROACH</a>
23       <a href="#">CONTACT ME</a>
24
25     </div>
26   </div>
27
28   <!--BREAK BREAK BREAK BREAK-->
29
30
31   <h2>News <br>Website</h2>
32   <div class="projects">
33     <div class="me">
34       <p><small>selftext</small>After I failed at my first Javascript project, I needed a confidence boost. So I decided to build another website using html and css. It was gonna be a news website and I wanted it to have a lot more design elements than my previous work. I visualized on</p>
35       <p><small>selftext</small>paper what my layouts would look like before coding, and then added new fonts, styled buttons, more colour, pictures and patterns to it. Even the buttons had click animations! I think it turned out really nice.</p>
36     </div>
37   </div>
38
39   <br>
40   <br>
41   <br>
42   <!--BREAK BREAK BREAK BREAK-->
43   <div id="show">
44     
45   </div>
46
47   <p>News Website coded with html and css. </p>
48
49   <!--BREAK BREAK BREAK BREAK-->
50
51   <div class="coding">
52     <div class="html">
53       
54     </div>
55     <div class="css">
56       
57     </div>
58
59     <div class="labels">
60       <p>HTML</p>
61       <p>CSS</p>
62     </div>
63
64   </div>
65
66
67   </body>
68
69
70 </html>
71
72 <footer>
73   <p>LEO ROBERTS! - Celebrate Crafts Brief - 2022</p>
74 </footer>
75
76
77

```

html

```

body {
  background-color: #fffff5;
}

.nav-bar {
  display: flex;
  justify-content: space-between;
  margin-left: 50px;
  margin-right: 50px;
  margin-bottom: 50px;
  margin-top: 50px;
  align-items: center;
  border-style: solid;
  border-width: 1px;
  position: -webkit-sticky;
  position: sticky;
  top: 0;
  background-color: #fffff5;
  font-family: Arial, Helvetica, sans-serif;
  font-size: 15px;
  margin-bottom: 15px;
  justify-content: flex-start;
  margin-right: 50px;
  margin-bottom: 50px;
}

.nav-right {
  display: flex;
  list-style: none;
  gap: 20px;
  justify-content: space-between;
  align-items: center;
  margin-right: 100px;
  gap: 50px;
  width: 100px;
}

.nav-left {
  display: flex;
  justify-content: center;
  flex-grow: 1;
  height: 100px;
  width: 100px;
}

.top-lines {
  border: 0;
  clear:both;
  display:block;
  width: 1px;
  background-color:#000000;
  height: 50px;
  margin-left: 0px;
  margin-right:0px;
  margin-top: 10px;
  margin-bottom: 10px;
}

.eim {
  width: 1324px;
  height: 650px;
  overflow: auto;
  margin: 50px;
  margin-bottom: 20px;
  background-color: #000000;
  border: 3px;
  border-color: #191919;
  border-style: solid;
}

.ein {
  width: 1324px;
  height: 100px;
  margin-left: 50px;
  margin-right: 50px;
  margin-top: 10px;
  font-size: 17px;
  font-family: Arial, Helvetica, sans-serif;
  line-height: 25px;
}

.sub-head {
  margin-left: 50px;
  margin-right: 50px;
  margin-top: 10px;
  font-size: 17px;
  font-family: Arial, Helvetica, sans-serif;
  line-height: 25px;
}

.me {
  display: flex;
  justify-content: space-between;
  flex-direction: row;
  margin-left: 50px;
  margin-right: 50px;
  gap: 50px;
  width:100px;
}

.self {
  width: 400px;
  height: 400px;
  border-radius: 20px;
}

.selftext {
  font-size: 17px;
  line-height: 25px;
  max-width: 700px;
}

/* BREAK BREAK BREAK BREAK */

.js {
  max-height: 500px ;
  overflow: auto;
  width: 500px;
}

.codeimg {
  width: 500px;
}

/* BREAK BREAK BREAK BREAK */

.footer {
  background-color: #ED662C;
  height:100px;
  display:flex;
  justify-content: center;
  align-items: center;
  margin-top: 100px;
  font-size: 50px;
}

.footer-text {
  color: #rgb(255, 255, 255);
  font-family: Arial, Helvetica, sans-serif;
  font-size: 17px;
}

.labels {
  font-size: 17px;
  font-family: Arial, Helvetica, sans-serif;
  line-height: 25px;
  font-weight: bold;
  display: flex;
  justify-content: center;
  margin-top: 50px;
  gap: 50px;
}

.html {
  max-height: 500px ;
  overflow: auto;
  width: 500px;
}

.css {
  max-height: 500px ;
  overflow: auto;
  width: 500px;
}

/* BREAK BREAK BREAK BREAK */

```

CSS

Final Outcome Overview

Final outcome:

Overall I'm really happy with how my final outcome turned out. I set myself the task of coding a website that depicts clear development of me learning to code. I wanted the website to also look good design wise and I think for the most part I achieved this. Using the things I'd learnt on the website was also important to me.

Website elements:

Animation
Links
Flex-box
Side-scroll
Hover effects
Buttons
Lists
Gesalt principles

What I would do better:

If I was to do this website again I would want to make the website responsive, I think this is really essential to a good website and I'm disappointed I didn't try and do it, even after I learnt how to. I also think I could have designed the product pages to be a bit more exciting visually.

Gesalt Principles:

While I was making my website I was thinking about the Gesalt principles and how I could incorporate them into my final outcome. I think using them improved the design of my website.

Laws I used:

The law of similarity
The law of proximity
The law of common-region
The law of focal point
The law of continuity

Project summary

I'm glad I chose to base this project around code, I've learned so much and discovered something that I would like to pursue as a career. The process of creating a design and building each individual element of it seperately until it comes together as one is so fulfilling. It gives me a bigger sense of achievement and satisfaction compared to working on flat designs for a product, brand etc. This project has helped me overcome a big creative block and re-ignited my passion for design.

```
1  
2  
3  
4  
5 <h1>THE END</h1>  
6  
7 |
```