HTML & CSS Part 2

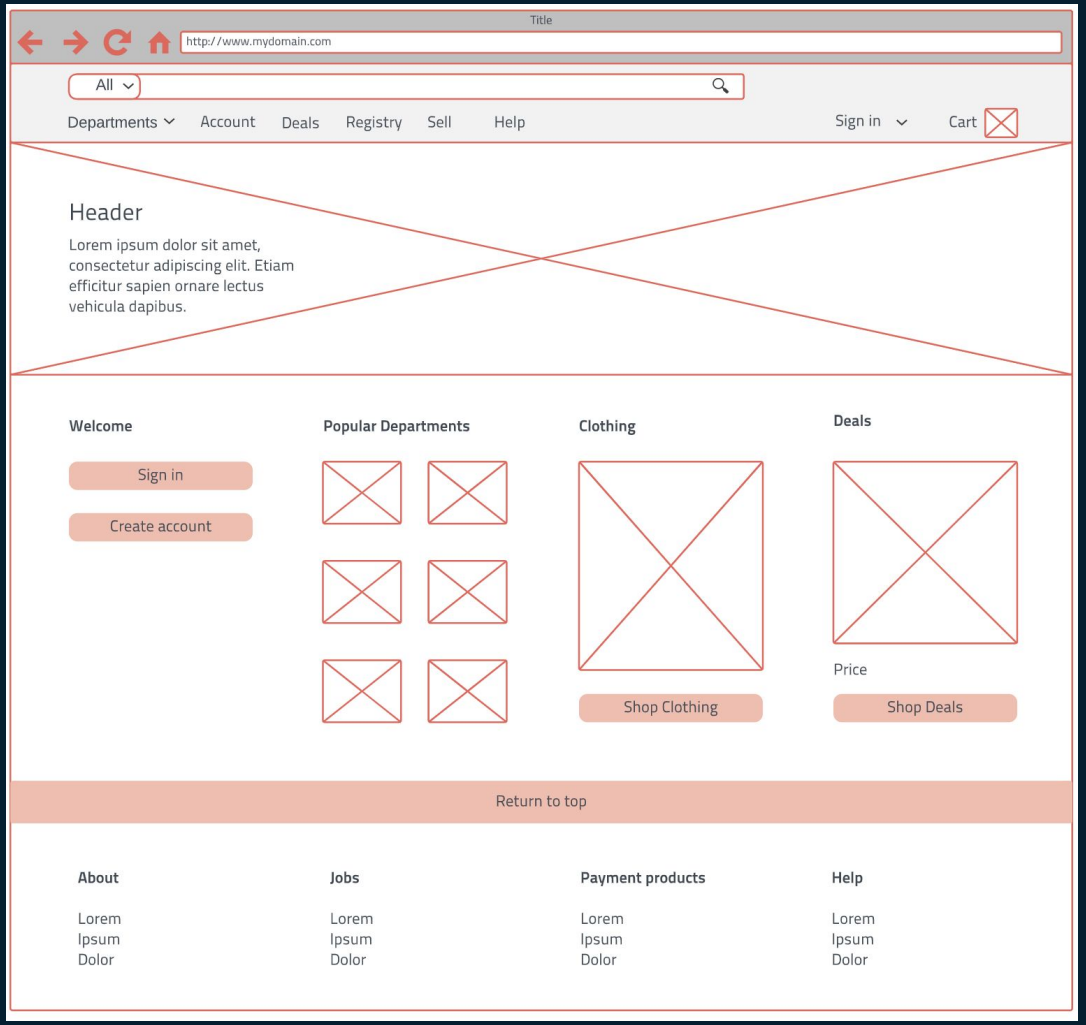
=> wireframes
=> transition
=> 3d transformation
=> keyframes
=> parallax
=> filters

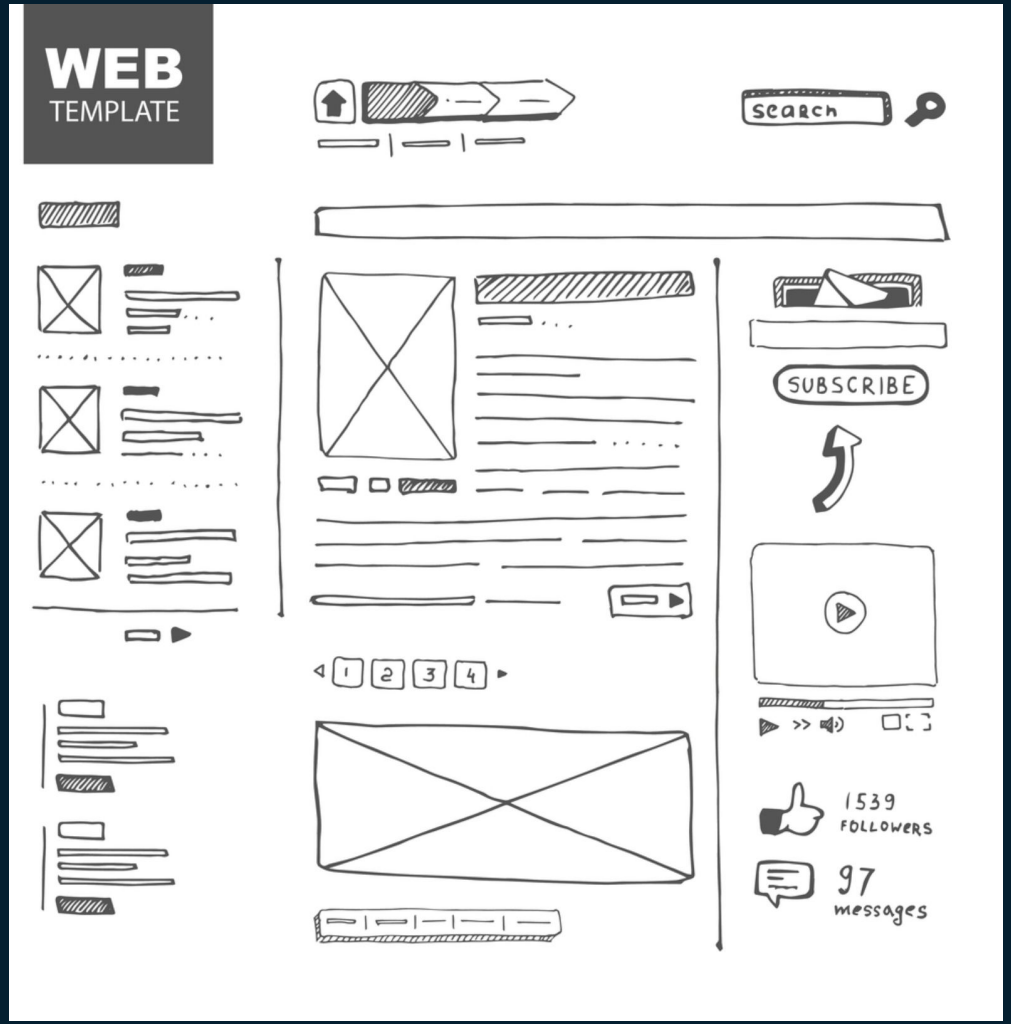
1. Name me 5 semantic tags?
2. What semantic tag do we use to make text italic?
3. What semantic tag do we use to make text bold?
4. Which two types of elements do we have in flexbox?
5. How to align elements horizontally with flexbox?
6. How to align elements vertically with flexbox?
7. How to change direction for elements flow with flexbox?
8. How to align only one element with flexbox?
9. What `*flex-wrap: wrap*` does?
10. What options do we have for `*justify-content*`?

What is Wireframes?

“Wireframe design is the process of creating a preliminary visual guide of a potential website's framework. This document can then serve as a blueprint for the web design team to expand upon”

“The goal of wireframe design is to establish a schema for how the information will be presented and prioritized to the visitor”


Title
http://www.mydomain.com
All
Departments
Account
Deals
Registry
Sell
Help
Sign in
Cart
Header
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam efficitur sapien ornare lectus vehicula dapibus.
Welcome
Sign in
Create account
Popular Departments
Clothing
Shop Clothing
Deals
Price
Shop Deals
Return to top
About
Lorem
Ipsum
Dolor
Jobs
Lorem
Ipsum
Dolor
Payment products
Lorem
Ipsum
Dolor
Help
Lorem
Ipsum
Dolor


WEB
TEMPLATE
search
SUBSCRIBE
1 2 3 4
1539
FOLLOWERS
97
messages

Simple tool

https://wireframe.cc/

How many times did you refactor your HTML?

If you don’t have a clear understanding of how need to structure your layout you will waste your time for refactoring HTML, rewriting your CSS, and **CSS**, in general, become a **nightmare**.

Development process

Tools

Apps

- Photoshop
- Sketch
- Figma

Online

- Mockflow - https://www.mockflow.com/
- Ninjamock

Paper

How to do that?

Use a different colour for each level to separate it visually.

Site Title

ST PHILIPS

WHY US | OUR TEAM | CASE STUDIES | OUR SITES | INSIGHT | CONTACT

Case Studies

View All

Cromwell Lane
Coventry, West Midlands

TBC ACRES | 240 HOMES | RESIDENTIAL

Webheath
Redditch, Worcestershire

336 ACRES | 2800 HOMES | RESIDENTIAL

Eccleshall
Eccleshall, Staffordshire

6.2 ACRES | 20 HOMES | RESIDENTIAL

Wrap your image

```
1 <div class="img-wrapper">
2     <img src="./kitten.jpeg" alt="kitten" />
3 </div>
```

Group similar types of elements

```
<div class="cards">
  <div class="card">...</div>
  <div class="card">...</div>
  <div class="card">...</div>
  <div class="card">...</div>
</div>
```

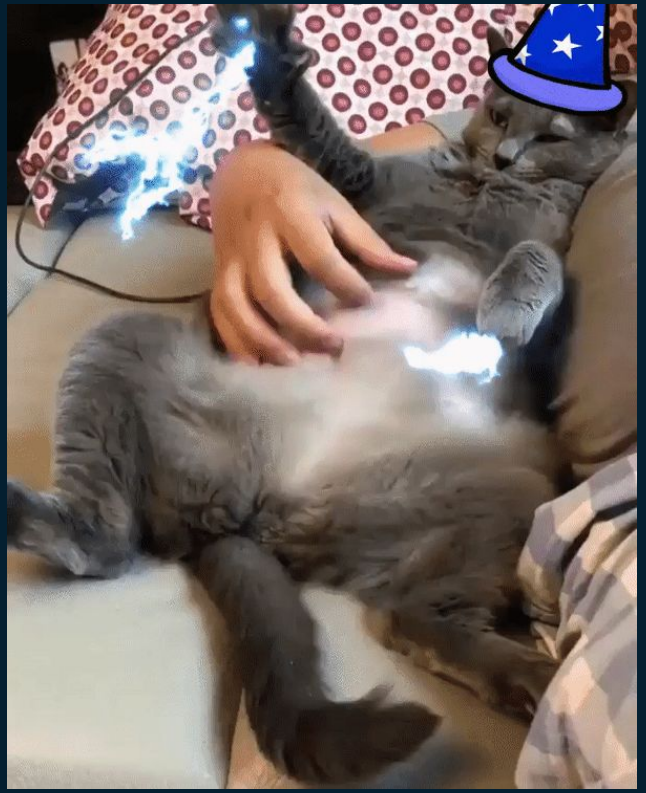
Divide visually

```
1 <article class="article">
2   <header> ... </header>
3   <section> ... </section>
4   <footer> ... </footer>
5 </article>
```

Change flow direction

```
<div class="column">
  <h1 class="title">...</h1>
  <h5 class="subtitle">...</h5>
  <div class="row">
    <div class="img_wrapper"></div>
    <div class="img_wrapper"></div>
    <div class="img_wrapper"></div>
  </div>
</div>
```

Emmet



“Ultra-fast coding. With Emmet you can quickly write a bunch of code, wrap code with new tags, quickly traverse and select important code parts and more!”

https://emmet.io/

Examples

Bonus

Pixel perfect - https://cantunsee.space/

Transform

“The transform property provides you ability visually manipulate an element by rotating, scaling, skewing or translating”

Most popular

=> transform: rotate(0.5turn);

=> transform: rotateX(10deg);

=> transform: rotateY(10deg);

=> transform: translate(12px, 50%);

=> transform: translateX(2em);

=> transform: translateY(3in);

=> transform: scale(2, 0.5);

=> transform: scaleX(2);

=> transform: scaleY(0.5);

** You can find examples in git folder: /lesson_6/examples/index.html*

More

=> transform: rotate3d(1, 2.0, 3.0, 10deg);

=> transform: rotateZ(10deg);

=> transform: translate3d(12px, 50%, 3em);

=> transform: translateZ(2px);

=> transform: scale3d(2.5, 1.2, 0.3);

=> transform: scaleZ(0.3);

=> transform: matrix(1.0, 2.0, 3.0, 4.0, 5.0, 6.0);

=> transform: matrix3d(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1);

=> transform: perspective(17px);

=> transform: skew(30deg, 20deg);

=> transform: skewX(30deg);

=> transform: skewY(1.07rad);

** You can find examples in git folder: /lesson_6/examples/index.html*

Resources

https://3dtransforms.desandro.com/perspective

https://rupl.github.io/unfold/

Transition

“The transition allows you to change values smoothly instead of having property changes immediately.”

Transition

```
1 .animation {
2     transition: "transition-property"
3                 "transition-duration"
4                 "transition-timing-function"
5                 "transition-delay"
6 }
```

Transition-timing-function

=> ease

=> linear

=> ease-in

=> ease-out

=> ease-in-out

=> step-start

=> step-end

https://css-tricks.com/almanac/properties/t/transition-timing-function/

Example

```
1 .animation {
2     transition: height 0.5s ease 3s;
3 }
```

Keyframes

“The @keyframes CSS at-rule controls the intermediate steps in a CSS animation sequence by defining styles for keyframes (or waypoints) along the animation sequence. ” - ~MDN

```
@keyframes slideIn {
  from {
    // your stylings
  }
  to {
    // your stylings
  }
}
```

```
@keyframes slideIn {
  0% {
    // your stylings
  }
  100% {
    // your stylings
  }
}
```

https://keyframes.app/

Animation

```
.box {
  animation:    "animation-name"
                "animation-duration"
                "animation-timing-function"
                "animation-delay"
                "animation-iteration-count"
                "animation-direction"
                "animation-fill-mode"
                "animation-play-state";
}
```

Example

```
.box {
  animation: slidein 3s ease-in 1s infinite reverse both running;
}
```

Animation timing function

Animation-timing-function sets how an animation progresses through the duration of each cycle.

=> ease;
=> ease-in;
=> ease-out;
=> ease-in-out;
=> linear;
=> step-start;
=> step-end;

Function values

=> cubic-bezier(0.1, 0.7, 1.0, 0.1);
=> steps(5, end);

Steps functions keywords

=> steps(4, jump-start);
=> steps(10, jump-end);
=> steps(20, jump-none);
=> steps(5, jump-both);
=> steps(6, start);
=> steps(8, end);

Animation fill mode

sets how a CSS animation applies styles to its target before and after its execution

=> none;

=> forwards;

=> backwards;

=> both;

Animation direction

sets whether an animation should play forwards, backwards, or alternating back and forth.

=> normal;

=> reverse;

=> alternate;*(The animation is played forwards first, then backwards)*

=> alternate-reverse;

(The animation is played backwards first, then forwards)

Animation play state

sets whether an animation is running or paused.

=> running

=> paused

Tools

Simple

https://matthewlein.com/tools/ceaser

With examples

http://animista.net/

Parallax

“background-attachment”

=> **scroll** – default state, image **scrolls with your page**
=> **initial** – resetting to default state
=> **inherit** – inherit from the parent
=> **fixed** – your image will stay on the same place
=> **local** – **scrolls with element content**

Example

https://www.w3schools.com/howto/howto_css_parallax.asp

Filter

*The **filter** CSS property applies graphical effects like blur or color shift to an element.*

=> filter: blur(5px);
=> filter: brightness(0.4);
=> filter: contrast(200%);
=> filter: drop-shadow(16px 16px 20px blue);
=> filter: grayscale(50%);
=> filter: hue-rotate(90deg);
=> filter: invert(75%);
=> filter: opacity(25%);
=> filter: saturate(30%);
=> filter: sepia(60%);

Text gradient

```css
p {
  background: -webkit-linear-gradient(#ddd, #f00);
  -webkit-background-clip: text;
  -webkit-text-fill-color: transparent;
}
```