

GRAND CIRCUS
DETROITGRAND CIRCUS
DETROITGRAND CIRCUS
DETROITGRAND CIRCUS
DETROIT

HTML

GRAND CIRCUS
DETROITGRAND CIRCUS
DETROITGRAND CIRCUS
DETROITGRAND CIRCUS

HTML & CSS

Language Short For...

HTML

Hypertext Markup Language

CSS

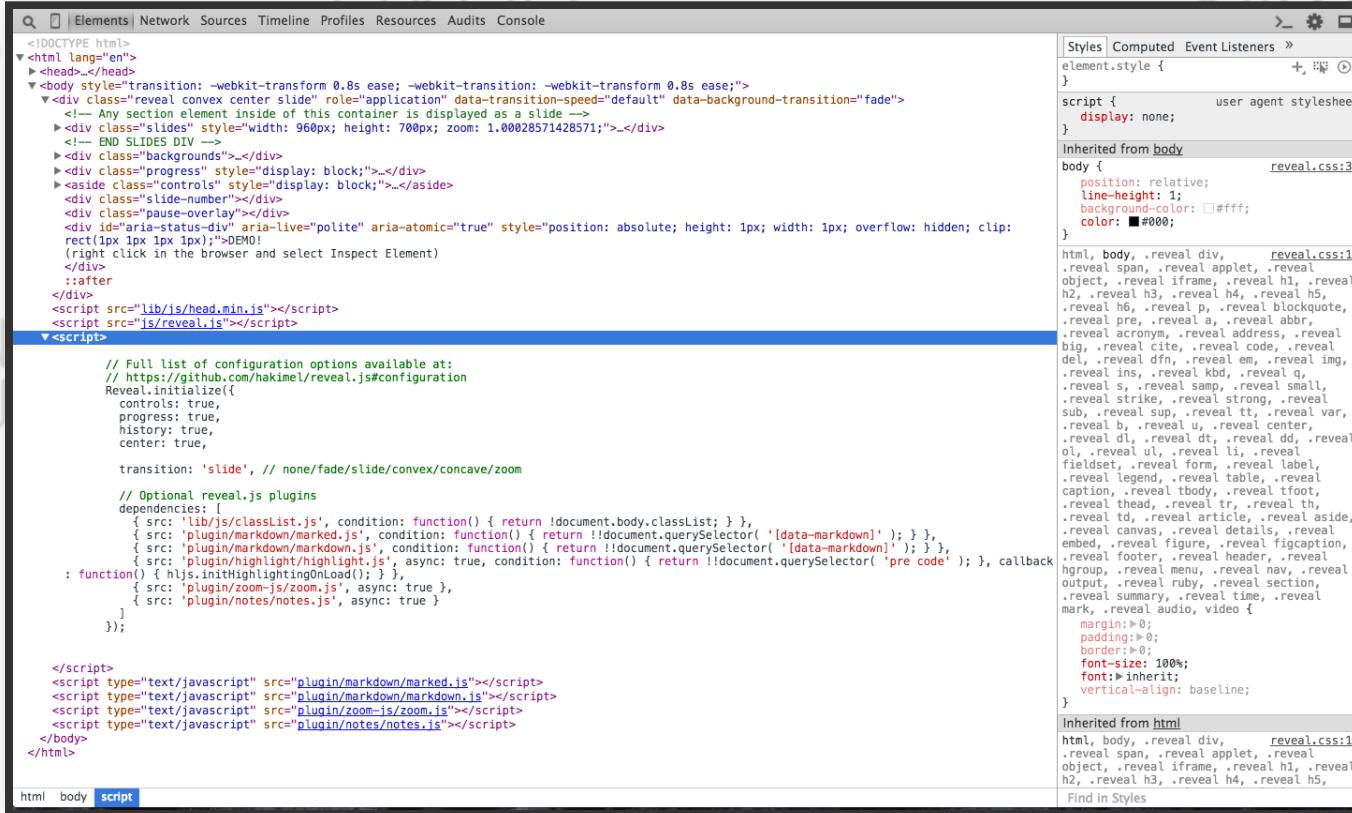
Cascading Style Sheets

Every website on the internet uses HTML & CSS.

Just about 96% of all websites online are using JavaScript.

YOUR NEW BEST FRIEND

It is called the Dev Tools Inspector (right click in the browser and select **Inspect Element**)



The screenshot shows the DevTools Inspector open in a browser. The left pane displays the DOM tree for a presentation slide. The right pane shows the Styles tab of the Inspector, which includes the Computed style rules for the selected element, the Event Listeners tab, and the Inherited from body and Inherited from html tabs.

```
<!DOCTYPE html>
<html lang="en">
  <head></head>
  <body style="transition: -webkit-transform 0.8s ease; -webkit-transition: -webkit-transform 0.8s ease;">
    <!-- Any section element inside of this container is displayed as a slide -->
    <div class="slides" style="width: 960px; height: 700px; zoom: 1.00028571428571;">_</div>
    <!-- END SLIDES DIV -->
    <div class="backgrounds">_</div>
    <div class="progress" style="display: block;">_</div>
    <aside class="controls" style="display: block;">_</aside>
    <div class="slide-number">_</div>
    <div class="pause-overlay">_</div>
    <div id="aria-status-div" aria-live="polite" aria-atomic="true" style="position: absolute; height: 1px; width: 1px; overflow: hidden; clip: rect(1px 1px 1px 1px);">_<!--DEMO!
      (right click in the browser and select Inspect Element)
    </div>
    </div>
    <script src="lib/js/head.min.js"></script>
    <script src="js/reveal.js"></script>
  </body>
```

// Full list of configuration options available at:
// https://github.com/hakimel/reveal.js#configuration
Reveal.initialize({
 controls: true,
 progress: true,
 history: true,
 center: true,

 transition: 'slide', // none/fade/slide/convex/concave/zoom

 // Optional reveal.js plugins
 dependencies: [
 { src: 'lib/js/classList.js', condition: function() { return !document.body.classList; } },
 { src: 'plugin/markdown/markdown.js', condition: function() { return !!document.querySelector('[data-markdown]'); } },
 { src: 'plugin/markdown/markdown.js', condition: function() { return !!document.querySelector('[data-markdown]'); } },
 { src: 'plugin/highlight/highlight.js', async: true, condition: function() { return !!document.querySelector('pre code'); } }, callback
 : function() { hljs.initHighlightingOnLoad(); } ,
 { src: 'plugin/zoom-js/zoom.js', async: true },
 { src: 'plugin/notes/notes.js', async: true }
]});

</script>
<script type="text/javascript" src="plugin/markdown/markdown.js"></script>
<script type="text/javascript" src="plugin/markdown/markdown.js"></script>
<script type="text/javascript" src="plugin/zoom-js/zoom.js"></script>
<script type="text/javascript" src="plugin/notes/notes.js"></script>
</body>
</html>

html body script

Styles	Computed	Event Listeners
element.style {	+ user agent stylesheet	
script {	display: none;	
Inherited from body		
body {	reveal.css:30	
position: relative;		
line-height: 1;		
background-color: #ffff;		
color: #000;		
html, body, .reveal div, .reveal span, .reveal applet, .reveal object, .reveal iframe, .reveal h1, .reveal h2, .reveal h3, .reveal h4, .reveal p, .reveal blockquote, .reveal pre, .reveal a, .reveal abbr, .reveal acronym, .reveal address, .reveal big, .reveal cite, .reveal code, .reveal del, .reveal dfn, .reveal em, .reveal img, .reveal ins, .reveal kbd, .reveal q, .reveal s, .reveal samp, .reveal small, .reveal strike, .reveal strong, .reveal sub, .reveal sup, .reveal tt, .reveal var, .reveal b, .reveal u, .reveal center, .reveal dl, .reveal dt, .reveal dd, .reveal ol, .reveal ul, .reveal li, .reveal fieldset, .reveal form, .reveal label, .reveal legend, .reveal table, .reveal caption, .reveal tbody, .reveal tfoot, .reveal thead, .reveal tr, .reveal th, .reveal td, .reveal article, .reveal aside, .reveal canvas, .reveal details, .reveal embed, .reveal figure, .reveal figcaption, .reveal footer, .reveal header, .reveal hgroup, .reveal menu, .reveal nav, .reveal output, .reveal ruby, .reveal section, .reveal summary, .reveal time, .reveal mark, .reveal audio, video {	reveal.css:11	
margin: 0;		
padding: 0;		
border: 0;		
font-size: 100%;		
font: inherit;		
vertical-align: baseline;		
Inherited from html		
html, body, .reveal div, .reveal span, .reveal applet, .reveal object, .reveal iframe, .reveal h1, .reveal h2, .reveal h3, .reveal h4, .reveal h5,		



NESTING & INDENTING

HTML elements "nest" inside of one another.

The element that opens first closes last.

Understanding how to nest elements is crucial to writing clean and maintainable code.

COMMENTS

HTML offers comments.

Comments are useful for keeping code organized when developing.

The browser engine ignores them.

```
<!-- Hello, I am a comment. -->
```

FOLDER STRUCTURE

Folder structure is not exciting but it is *important*.

How a folder is structured is going to depend on the team/individual building the project.



CIRCUS
DETROIT



GRAND
CIRCUS
DETROIT



CTR
DET



GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT



CONTENT TAGS



GRAND
CIRCUS
DETROIT



GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT



GRAND
CIRCUS



CONTENT TAGS

Tag	Description
<code>div</code>	The defacto container element.
<code>p</code>	Used for body copy.
<code>h1 thru h6</code>	Designating titles/subtitles.
<code>ol</code>	Create a numbered list.
<code>ul</code>	Create an unordered list.
<code>li</code>	List elements.
<code>a</code>	Link to other pages or sites.
<code>link</code>	Import external documents.

HTML5 SEMANTIC TAGS

Tag	Description
<code>section</code>	The "new" div tag used for page organization.
<code>header</code>	Introductory and navigational content.
<code>main</code>	Content inside should not be repeated.
<code>footer</code>	Footer content (site map, internal links, etc.).
<code>nav</code>	A list of navigation links.
<code>audio</code>	Multimedia tag for playing audio files.
<code>video</code>	Multimedia tag for playing video files.
<code>canvas</code>	Element can be used to draw graphics via JavaScript.

TABLES

Tables are used to display data, not organize and position content.

The only exception to this rule is for developing emails due to client restrictions.

WHAT ABOUT BREAKS, BOLDS, AND HORIZONTAL RULES?

Any inline styling elements such as ``, `<i>`, or `
` should not be used.

If an element needs to have emphasis (bold or italics), use a `` or `` element.

Anyone on a screen reader or PDA will appreciate the thoughtful consideration.

If an element needs spacing, use `margin`.

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

css

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS

GRAND
CIRCUS
DETROIT

GR
CIR

GRAND
CIRCUS
DETROIT

GR
CIR

WHAT IS CSS?

CSS is short for Cascading Stylesheets.

It is the language for styling HTML.

CSS can be written to do anything from changing fonts and colors to creating beautiful transitions and animations.

CSS SELECTORS

The selector instructs the browser to search the page for any HTML element that matches the given criteria.

It applies any applicable declarations to the matched element(s).

CSS SELECTORS

Selector Description

`tagname` Selector based on the element's tag name.

`.` Selector based on a class name.

`#` Selector based on an id.

CSS DECLARATIONS

Declarations are made up of the property needing to be altered and the value given to the property.

Declarations can be grouped so that more than one declaration may be applied to a selected element.

Curly brackets must surround declaration groups.

Declarations must end in a semicolon.

```
.selector {  
    background-color: red;  
    color: white;  
    border: 1px solid black;  
    border-radius: 5px;  
}
```

COMMENTS

Just like HTML, CSS offers comments:

```
/* This is a CSS comment, it can be multi-line */
```

CSS PSUEDO-CLASSES & PSUEDO-SELECTORS

These are used to add special effects to some selectors.

```
a:link { /* unvisited link */
  color: aliceblue;
}
a:visited { /* visited link */
  color: darkblue;
}
a:hover { /* mouse over link */
  color: lightblue;
}
a:active { /* mouse click link */
  color: yellow;
}
```

OTHER PSUEDO-CLASSES

Class	Description
:first-child	The first direct child of its parent.
:first-letter	First letter of an element's text content.
:first-line	First line of an element's content.
:last-child	Any element which is the last child of its parent.
:only-child	Any element which is the only child of its parent.
:empty	Any element which has no content.
:nth-child	Any element which is the nth child of its parent.

Demo

NTH-CHILD SYNTAX

`:nth-child` takes an expression to determine which children to select.

It is common to use `nth-child` for alternating styles of large groups of data like table rows, columns, or lists.

```
p:nth-child(an + b) {  
  property:value;  
}  
div:nth-child(3) {  
  color:red;  
}  
ul:nth-child:(2n + 5) {  
  color:yellow;  
}
```

CSS PROPERTIES

There are hundreds of CSS properties that are available for use.

We will go over a few of the most common, however, the best strategy is to Google if curiosity strikes.

CSS PROPERTIES

Property	Description
background-color	Background color for an element.
color	Color of text <i>and</i> border of an element.
font-family	Typeface for text.
font-size	Size for text (px, %, em, pt).
font-weight	Used to bold text (if possible).
text-decoration	Used for changing the decoration (underlining and strikethroughs).
height	Specifies the height of an element.
width	Specifies the width of an element.

COLOR IN CSS

Method	Syntax	Description
color name	white	One of the 140 predefined colors.
hexadecimal	#FF0000	Rgb values in hex. Each value can be within a range is 0-9 or A-F.
rgb	rgb(255, 0, 187)	Rgb values.
rgba	rgba(255, 0, 187, 0.5)	Rgb values with an added alpha (opacity) value.

CSS UNITS

Method	Syntax	Description
em	1em	Scalable unit based on font size of the parent element.
rem	1rem	Scalable unit based on font size of the html root.
pixels	16px	Fixed number of pixels.
percent	120%	Percent value based on font size of the parent element.



EXERCISE

1. Pick from one of two images linked below.
2. Try to re-make these images with HTML and CSS.
3. Be ready to demo! Option 1 -- Option 2

GRAND
CIRCUS
DETROIT

DISPLAY PROPERTIES

DISPLAY

Display Description

block

Takes up the horizontal space of a line. This will stretch to fill all the space from left to right of its parent container.

inline

Rendered without starting a new line. They appear side by side until reaching the edge of its parent container.

inline-block

Inline element that can have a height and width declared.

flex

Element that will be stretched or shrunk to fit the container.

DISPLAY DEMOS

BLOCK:



INLINE:



Demo

BLOCK VS. INLINE ELEMENTS

Inline elements: `<a>`, ``, ``.

Block elements: `<div>`, `<p>`, ``, ``, `<table>`,
and just about everything else.

GRAND
CIRCUS
DETROIT

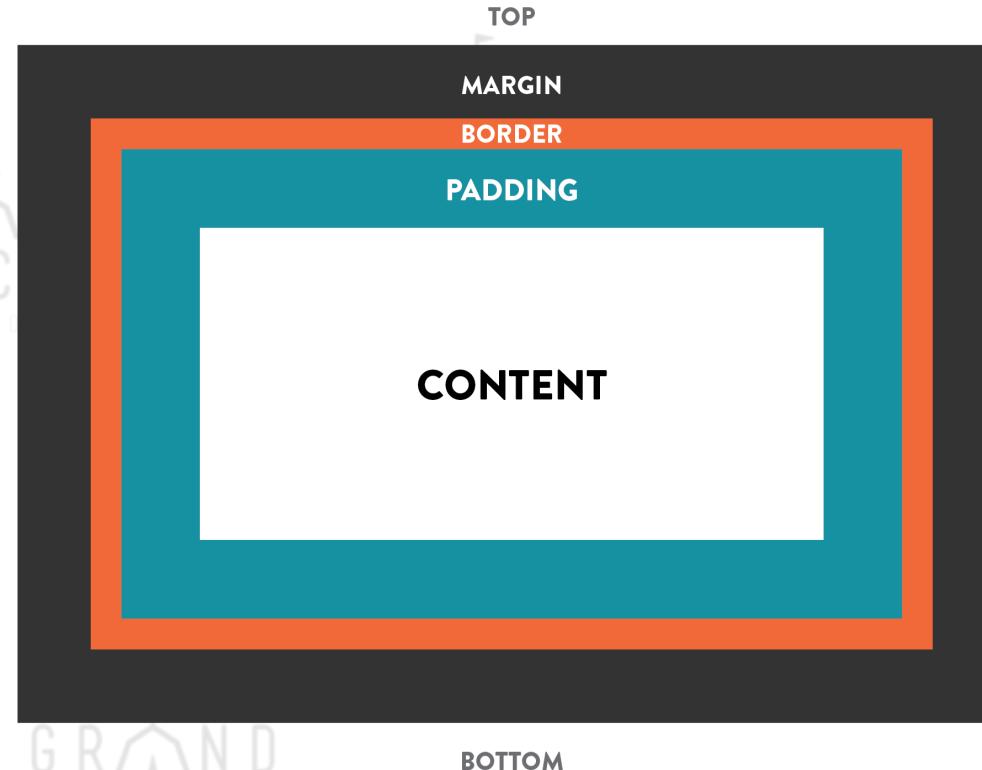
THE BOX MODEL

THE BOX MODEL

Every HTML element is in a box, regardless of its visible shape.

The total size of an element is a combination of the following: **content**, **padding**, **border**, and **margin**.

THE BOX MODEL



Demo

THE BOX MODEL

Property Description

padding

The space between the content within an element and the border. Padding can be specified in units of pixels (px), Ems (em), or points (pt) or percentage (%) of the containing element.

border

The edge around the element. It has elements of thickness, style, and color.

margin

The space outside the element that separates it from other elements. It is 'outside the fence' in terms of its relationship to the border.



MARGIN

```
p {  
  margin: 10px; /* 10 pixels of margin on all sides */}  
p {  
  margin: 10px 5px; /* 10 px on top/bottom & 5px on left/right */}  
p {  
  margin-top: 5px;  
  margin-right: 10px;  
  margin-bottom: 15px;  
  margin-left: 20px;  
}  
p {  
  margin: 5px 10px 15px 20px; /* top, right, bottom, left */}
```





CENTERING WITH MARGIN

`margin: 0 auto` can be used on an element that has a set width to center the element.

This method only works for horizontally centering.

Demo

MARGIN AUTO VS. TEXT-ALIGN CENTER

`margin: 0 auto` centers the given element within its parent

`text-align: center` centers the text that is inside the given element

CENTERING ON X AND Y

A solid choice for centering (both horizontally and vertically) is using a translate.

```
.vertical {  
  position: relative;  
  top: 50%;  
  transform: translateY(-50%);  
}  
  
.horizontal {  
  position: relative;  
  left: 50%;  
  transform: translateX(-50%);  
}  
  
.horizontal-vertical {  
  position: relative;  
  top: 50%;  
  left: 50%;  
  transform: translate(-50%, -50%);  
}
```

POSITIONING

CSS POSITIONING

Every HTML element has a property called **position** which dictates how that element flows on a document.

This property can be set to many different values, each of which behaves slightly differently.

Positioning gives a web developer more significant control over the design and overall look of a page.

POSITIONING

Position	Description
static	Default for all HTML elements. Adheres to the previously discussed behavior of block and inline elements.
relative	Relative positioned elements appear in the normal flow of the document but can be offset by using the top, bottom, left and right properties.
absolute	Absolutely positioned are removed from the normal flow of the document. They appear relative to their parent container elements, using offsets.
fixed	Fixed positioned elements are removed from the normal flow of the document. They aren't affected by and don't affect other elements.

ABSOLUTE WITHIN RELATIVE

When an absolutely positioned element is inside of a relatively positioned container, it is positioned relative to that container rather than the whole page.

DEMO

EXERCISE:

You try! Make a web page with 4 elements.

Heads up: This will look bad.

1. One should have a header tag with a background color of your choice, 100% width, and fixed positioned to the top, right of the viewport.
2. Give the body a viewport height of 200vh.
3. Make a div (give it a width and a height) under the header and give it a border of 1px solid black.
4. Put a paragraph tag with your name inside the div and absolutely position it to the bottom, right of the div.
5. Add an image of your choice without any positioning (this is your static positioned item).



MORE PROPERTIES OF INTEREST

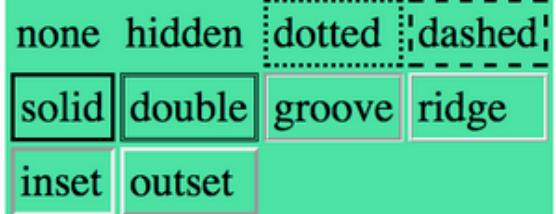
FLOAT

Floating takes an element in the normal flow and pushes it as far to the left or right of its parent element as possible.

When an element is floated, other elements will wrap around it.

To float an element, two things must be specified to the element: a width and direction to float.

FLOAT



Lorem ipsum dolor sit amet, consectetur adipisicing 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.

Lorem ipsum dolor sit amet, consectetur adipisicing 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. Lorem ipsum dolor sit amet, consectetur adipisicing 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.

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CLEAR

Clear specifies on which side of an element other elements cannot appear.

Clears can be applied to left, right, or both.

Demo

Z-INDEX

When elements are moved out of the normal flow of content (i.e., not position static), they can overlap.

Z-index can be used to define the order of overlapping elements.

The element with the highest z-index goes on top.

The element *cannot* have a position of static for z-index to work.

Demo

@FONT-FACE

The @font-face property allows the use of non-default fonts on websites.

Using external fonts is quite easy. Follow these steps:

1. Download your font
2. Place the font file in your web site
3. Create a @font-face CSS rule

```
@font-face {  
    font-family: "myCoolFont"; src: url("path/to/myCoolFont.ttf");  
}  
div.demo {  
    font-family: "myCoolFont";  
}
```

EXERCISE!

1. Download a free font from font-squirrel.
2. Place the font file in your pre-work, this cannot be a CDN.
3. Create a @font-face CSS rule and use it somewhere in your project.

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CSS3 PROPERTIES AND PREFIXES

VENDOR PREFIXES

Not all browsers support all of the newer CSS3 properties, which means additional rules must be created for specific browsers.

Great Prefix Resource

The order matters, so make sure the non-prefixed property goes last.

```
-webkit-*Android*/  
-webkit-*Chrome*/  
-webkit-*iOS*/  
-webkit-*Safari*/  
-o-*Opera*/  
-moz-*Firefox*/  
-ms-*Internet Explorer*/
```

CSS3 PROPERTIES

Property	Description
border-radius	Creates rounded corners or circular borders for elements.
opacity	Controls the transparency of an element.
box-shadow	Adds a calculated shadow effect to an element.
text-shadow	Adds a calculated shadow effect to an element.
transforms	Adds the ability to transform HTML elements (rotation, scale, translate).
animations	Allows for one or more property changes to be applied as an animation.

GRAND
CIRCUS
DETROIT

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

HTML/CSS LAB PART 1

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT



GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

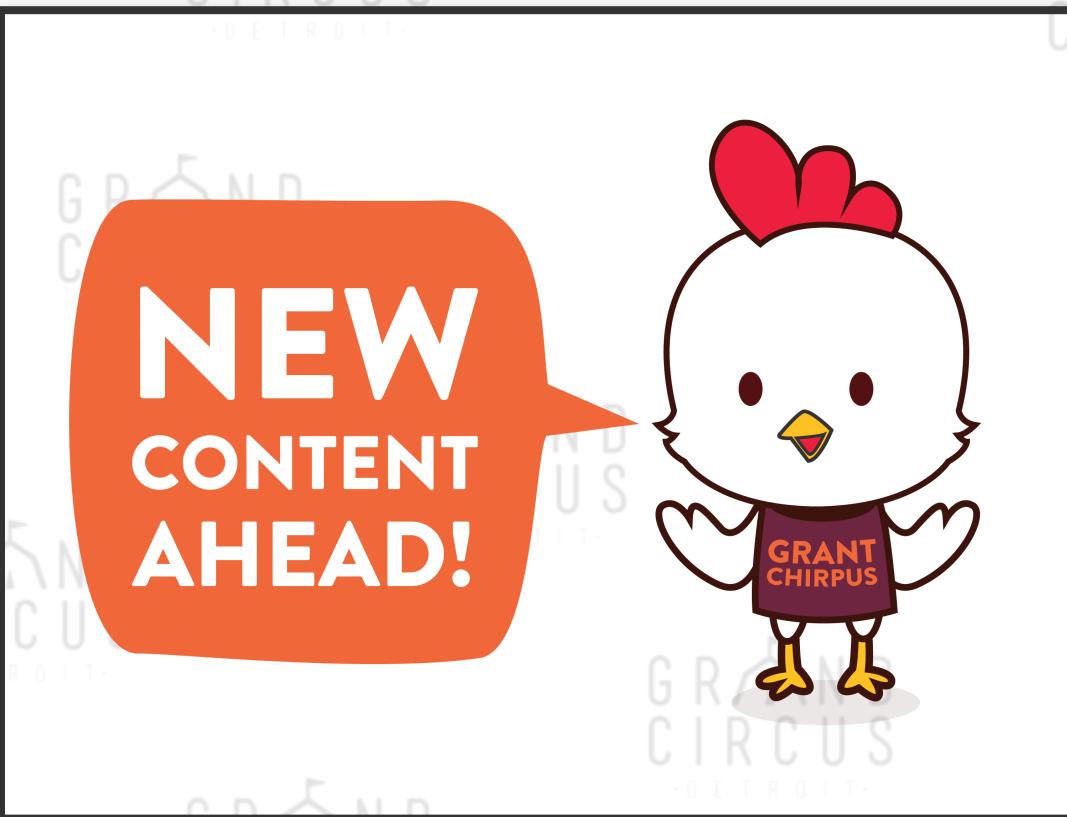
GRAND

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT



GRAND
CIRCUS
DETROIT

FLEXBOX

EXERCISE

A client wants you to make a site for their new dog walking business.

The client's head developer wants you to build our navigation using floats because that is what is being used on the rest of their site and she is most familiar with it.

The designer gives you the .png in the next slide, so you should make sure it looks like what the client wants!

EXERCISE

Remember - You can only use `floats` and `display: inline-block` right now.

The font they want you to use is `Lato`.

The navigation's background color is `#551A8B`.

The font color is `rgba(255, 255, 255, 0.8)`.

Good luck!

FLEXBOX

Think of a world where people do not have to use floats or inline-blocks for their layout!

Flexbox is a display property within CSS that allows a developer to plan and visually display their site's content.

There are tons of values that can be leveraged with flexbox.

BENEFITS OF USING FLEXBOX

1. Eliminates the need for calculating the heights of elements in JavaScript to set equal heights of divs.
2. No more floats or need for clear fixes.
3. Flexbox works beautifully for responsive layouts.
4. Flexbox calculates the widths of each element and sets the margins for us.

Here's using flexbox to accomplish equal heights through CSS!

Here's using flexbox eliminates the need for clear fixes.



GRAND
CIRCUS
DETROIT



HOW TO FLEX

THE PARENT CONTAINER

To use flexbox, wrap the items needing to be flexed in a container.

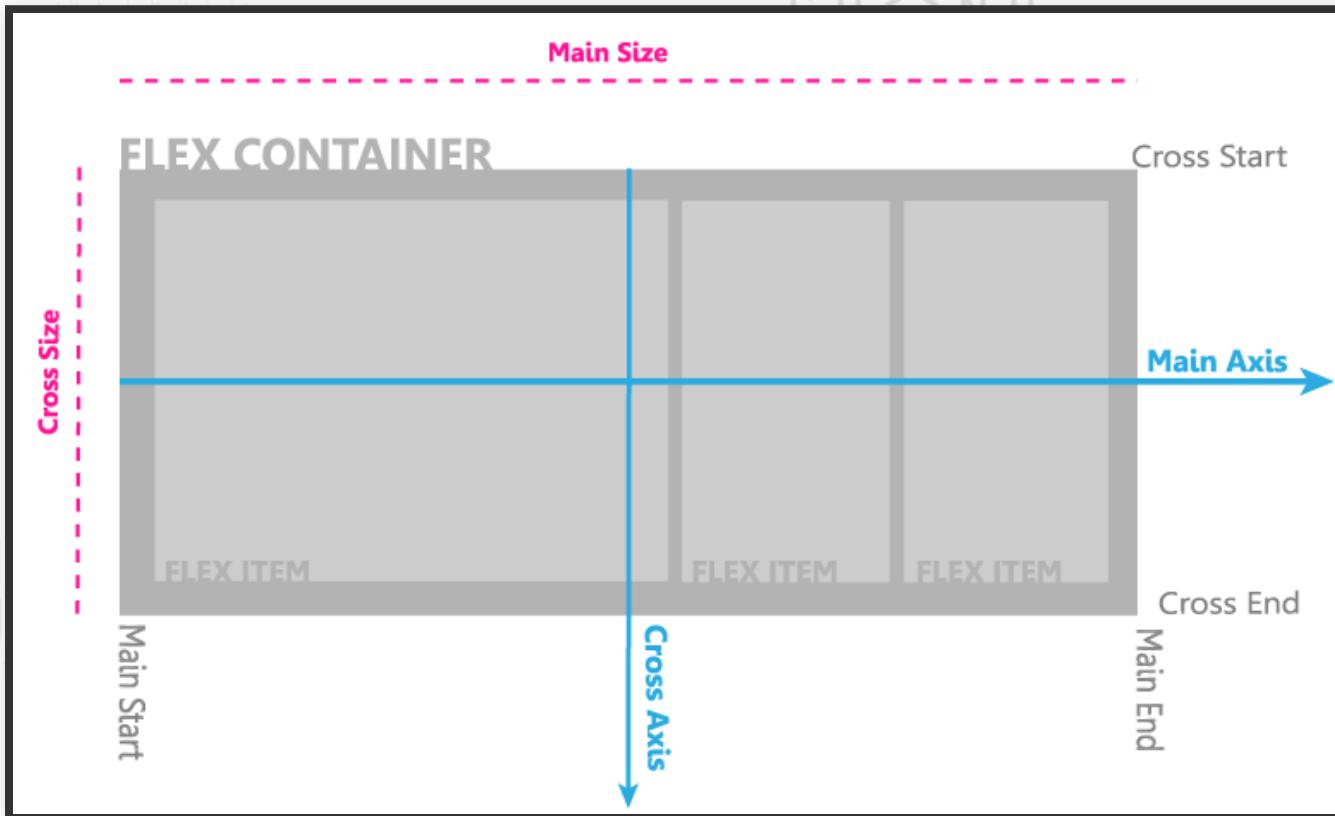
The container is often referred to as the **flex container** or as the **parent container**.

Do this by giving the display property a flex value. It should look like this.

```
.parent-container {  
  display: flex;  
}
```

Here's how we do it

CROSS AXIS AND MAIN AXIS



CROSS AXIS

The cross axis is the vertical axis along the flex container.

This allows for elements to move between the top, middle or bottom of the container.

The *align-items* property is what controls this axis.

The *align-items* property goes on the parent container and will be applied to child items.

ALIGN-ITEMS

Value	Description
flex-start	Items will line up at the top of the container.
flex-end	Items will line up at the bottom of the container.
center	Items will be centered in the container.
baseline	Items will be lined up along their baseline.
stretch	Items will stretch as tall as the container. This is the default.

Demo

MAIN AXIS

The main axis is the horizontal axis along the flex container.

This allows for elements to move between the left, middle or right of the container.

Spacing can also be applied either around or between these items.

The *justify-content* property is what controls this axis.

The *justify-content* property goes on the parent container and will be applied to child items.

MAIN AXIS

Value

Description

`flex-start`

Items will line up at the top of the container.

`flex-end`

Items will line up at the bottom of the container.

`center`

Items will be centered in the container.

`space-between`

Flex items are evenly distributed between items.

`space-around`

Flex items have space around all sides.

Demo

FLEX-DIRECTION

Flexbox items can go in two directions, rows and columns.

The direction can also be reversed.

Value	Description
row	Items will flow left to right.
row-reverse	The order of items will reverse row order.
column	Items will stack from top to bottom.
column-reverse	Items will stack in a reversed order from top to bottom.

Demo

FLEX-WRAP

The *flex-wrap* property controls whether the flex container is single-line or multi-line, which determines the direction new lines are stacked in.

Value	Description
<code>nowrap</code>	The default value. Items fall on a single line, even if they start to overlap each other.
<code>wrap</code>	Items fall on another line if there isn't enough room on the current line.
<code>wrap-reverse</code>	Items go on top of the previous line if there isn't enough room on the current line.

Demo!

EXERCISE

Oh no!

It turns out the head developer took a flexbox course
and realized the magic of flexbox.

Convert your old code to use flexbox, but still achieve
the same look as before.

Here's the image of the navigation again for you to
reference.

Grant's Dog Walking!

[Home](#) [Prices](#) [Contact Us](#)



EXERCISE

Great job!

The client was super happy with your flexibility and knowledge of flexbox.

Less code for them to implement in their project does not hurt either.





RESPONSIVE DEVELOPMENT

WHAT IS RESPONSIVE?

For a site to be responsive, it has to be able to be viewed on multiple screen sizes and screen orientations (landscape or portrait).

With so many different types of devices out there, all of them must be recognized and coded for (welcome to front-end development).

This is done by creating fluid layouts and using media queries.

MEDIA QUERIES

One of the ways to accomplish having responsive websites is by using media queries.

Media queries can detect a bunch of properties about the device used to view the site.

It allows the browser to use specifically developed CSS styles for the different screen sizes that it detects.

MEDIA QUERIES

Media queries are written like this:

```
@media [not|only] type [and] (expression) {  
    /* rules */  
}
```

Types of media can be all, screen, print, and speech.
Here's an example of how to use media queries.

MIN & MAX WIDTH

min-width applies to anything greater than or equal to the min-width will get the styles

max-width applies to anything less than or equal to the max-width will get the styles

YOU TRY IT!

Create a container element with three paragraph tags that have the same font size value.

Make their font size get more prominent by increasing the font-size as the viewport increases using two media query breakpoints.

Things to think about: will you need to use max-width or min-width for this situation?

VIEWPORT META TAG

The following code must be included within the `<head>` element.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

The previous tag tells the browser to render the width of the page at the width of its screen.

If that screen is 320px wide, the browser window will be 320px wide, rather than way zoomed out.



MOBILE-FIRST DEVELOPMENT

MOBILE-FIRST DEVELOPMENT

In this class, we will emphasize coding all sites mobile-first.

Mobile-first means any default styles will go above media queries, and any overriding code will go within media queries.

Typically, min-width is used for mobile-first development.



MOBILE-FIRST

In 2019, it is expected that there will be 4.68 billion mobile users.

It is far easier to build up than to scale down.

Mobile-first is also great for quicker load times and has a much higher SEO (Google ranks mobile-first sites higher).

MOBILE-FIRST

The standard device breakpoints to use are:

Breakpoint Device Types

480px

Larger phone screens.

768px

For most tablets.

992px

Large tablets to small laptops.

1200px

For anything larger than a tablet, think desktop screens.

EXERCISE

Let's go through how a mobile-first developer would typically code and test their site as they work.

Let's come up with a wireframe of what we want our site to look like together going from mobile to desktop.

We should minimally include a title, an about us section with two facts.

EXERCISE

Grant's Dog Walking Company has been doing great since you built that impressive navigation.

However, now, it has gotten so popular that people on their phones are trying to access it.

No one can use it because we did not develop it mobile-first!

Our client wants us to re-do our navigation (still using flexbox) but to work responsively.

EXERCISE

Here is the design.

They also want the name of the company to go on the right side on bigger screen sizes without changing the HTML structure.

Plan for them to change at the 768px breakpoint.

Grant's Dog Walking!
[Home](#)
[Prices](#)
[Contact Us](#)

[Home](#) [Prices](#) [Contact Us](#)

Grant's Dog Walking!

ADDITIONAL RESOURCES

Here are some additional readings to check out:

Resource	Description
flex-flow property	flex-flow property
flexbox cheat sheet	Flexbox cheat sheet
mobile-first importance	mobile first importance
more media-query resources	more media query resources

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

HTML/CSS LAB PART 2



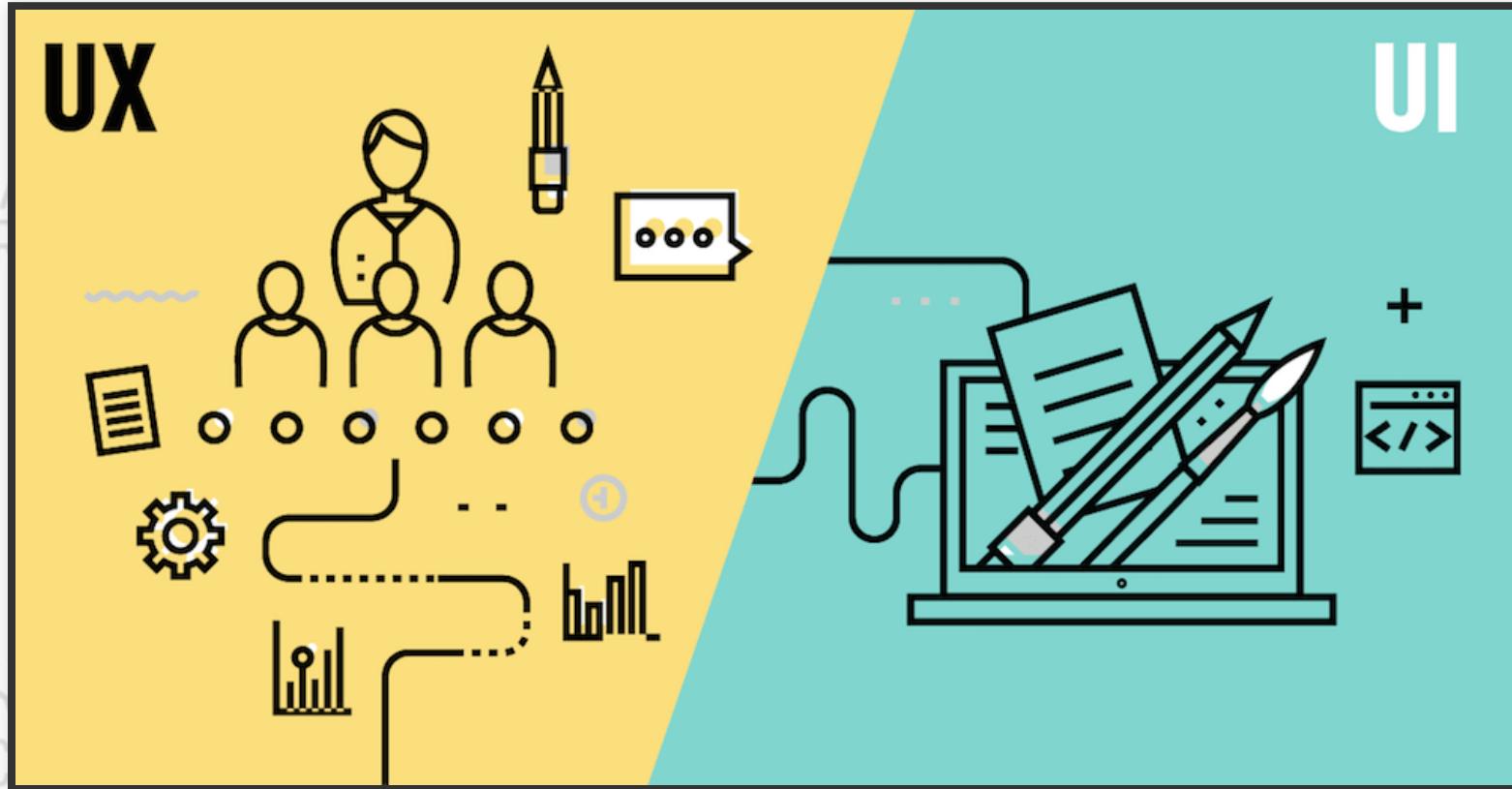
GRAND
CIRCUS
DETROIT

WIREFRAMING & UX/UI

GRAND
CIRCUS
DETROIT

UX/UI

GRAND
CIRCUS
DETROIT



source

GRAND
CIRCUS
DETROIT

GRAND

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

TERMINOLOGY

GRAND
CIRCUS
DETROIT

GRAND

GRAND



WHAT IS UX?

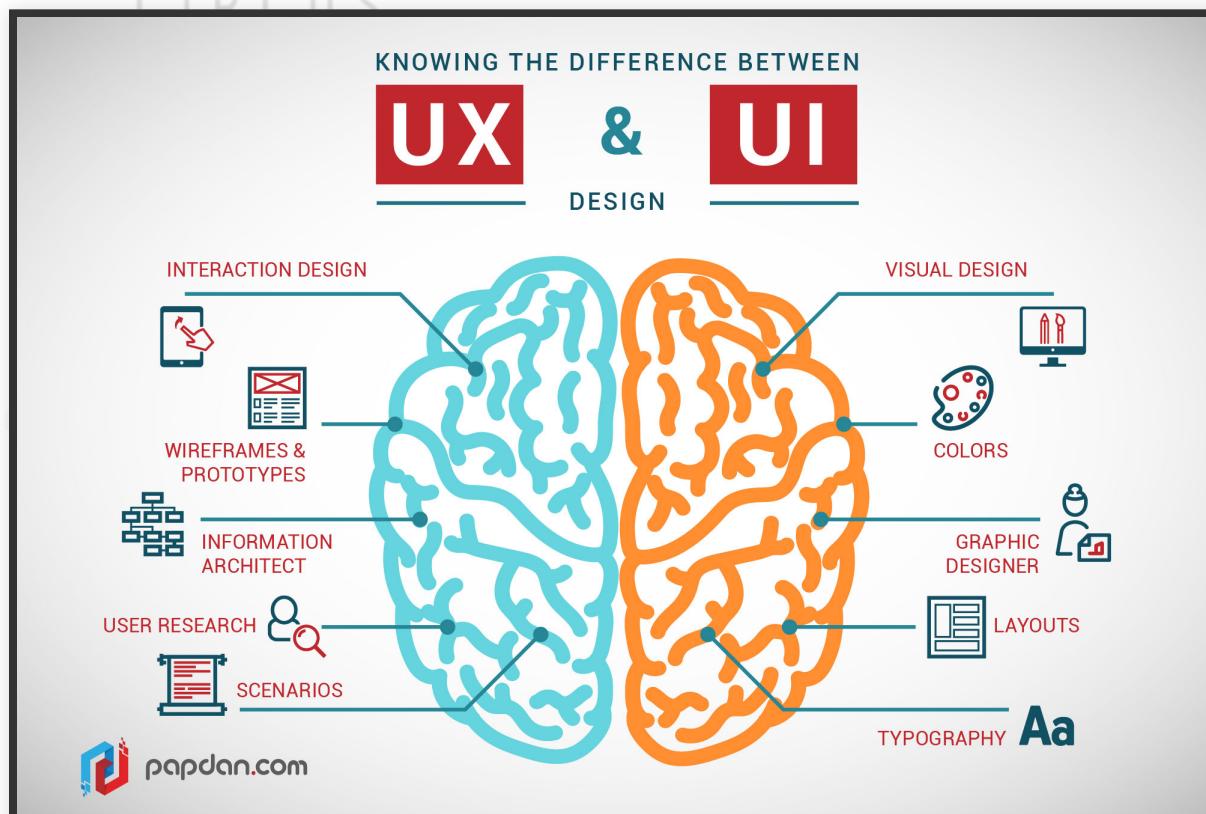
UX means user experience, it relates to the process of creating products that will provide personal and meaningful experiences.

WHAT IS THE UI?

UI refers to the user interface, it relates to the process of designing user interfaces for software or machines, such as the look of a mobile app, with a focus on the ease and enjoyment of the user.

It usually refers to the graphical interface and how the product is laid out.

WHAT'S THE DIFFERENCE?



WHAT'S THE DIFFERENCE?

- UI is how things look, UX is how things work.
- UI makes interfaces beautiful, UX makes interfaces useful.

USER INTERFACE

GRAND
CIRCUS
DETROIT

UI EXAMPLES

Inspiration UI

GRAND
CIRCUS
DETROIT

UI CORE PRINCIPLES

Principle Description

Clarity Things need to be apparent to the user like what happened, what they are, what they can do and what will happen if they do it.

Flexibility Designing something that looks good in all situations.

Familiarity Using familiar items and patterns in your app.

UI CORE PRINCIPLES

Principle

Efficiency

Description

User finishes main task in the most efficient way possible.

Consistency
and Structure

Examples are alignment and reusing shareable elements .

UI DESIGN TOOLS

- Sketch
- Adobe XD CC

UI FRAMEWORKS/KITS

- Bootstrap
- UI Kit
- Bulma

UI ACTIVITY

Make a contact form using Bootstrap styles!

Contact form should include the following:

1. 2 text inputs
2. two radio buttons
3. A date picker
4. A text area
5. A submit button

Be ready to demo!

GRAND
CIRCUS
DETROIT

USER EXPERIENCE

GRAND
CIRCUS
DETROIT

GRAND

GRAND
CIRCUS
DETROIT

HISTORY OF UX

"When we go into a new project, we believe in it all the way. We have the confidence in our ability to do it right." -Walt Disney (One of the original UXers)

HISTORY OF UX

You could say UX started during the machine age.

Easier it was to use the machines, quicker the work would get done.



HISTORY OF UX

A good example of experience design is Disneyland
(opened in 1955)

Walt Disney was a pioneer connecting experience
and emotion.



UX EXAMPLES

Pendar Yousefi Portfolio

Kathy Li



UX CORE PRINCIPLES

Principle Description

Useful Should be original and fulfill a need.

Usable Must be easy for everyone to use.

Desirable Image, identity, brand and other design elements are used to evoke emotion and appreciation.

UX CORE PRINCIPLES

Principle	Description
Findable	Content needs to be navigable and locatable on and offsite.
Accessible	Should be accessible to people with disabilities.
Credible	Users need to trust and believe what you tell them.

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GR
CIR
DET

UX TOOLS

Moqups

Invision

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS

UX PROCESS

UX PROCESS

User Centered Design

Design based on understanding users, tasks, and environments.

Driven by user-centered evaluation.

Addresses the whole user experience.

UX PROCESS

Step	Description
User Research	Know and understand users.
Analysis	Identifying design opportunities.
Design	Conceptualization and exploration.
Prototype	Implementation of ideas and feedback.
User Testing	Evaluation.

UX PROCESS: USER RESEARCH

What do your users want to accomplish?

What are their goals?

Who are the users?

TYPES OF USER RESEARCH

- Online Surveys
- Interviews
- Focus Groups
- Analytics



CIRCUS
DETROIT



ANALYSIS

Analyzing user research to create user personas

“The purpose of personas is to create reliable and realistic representations of your key audience segments for reference”. usability.gov

USER PERSONA EXAMPLE

GRAND
CIRCUS

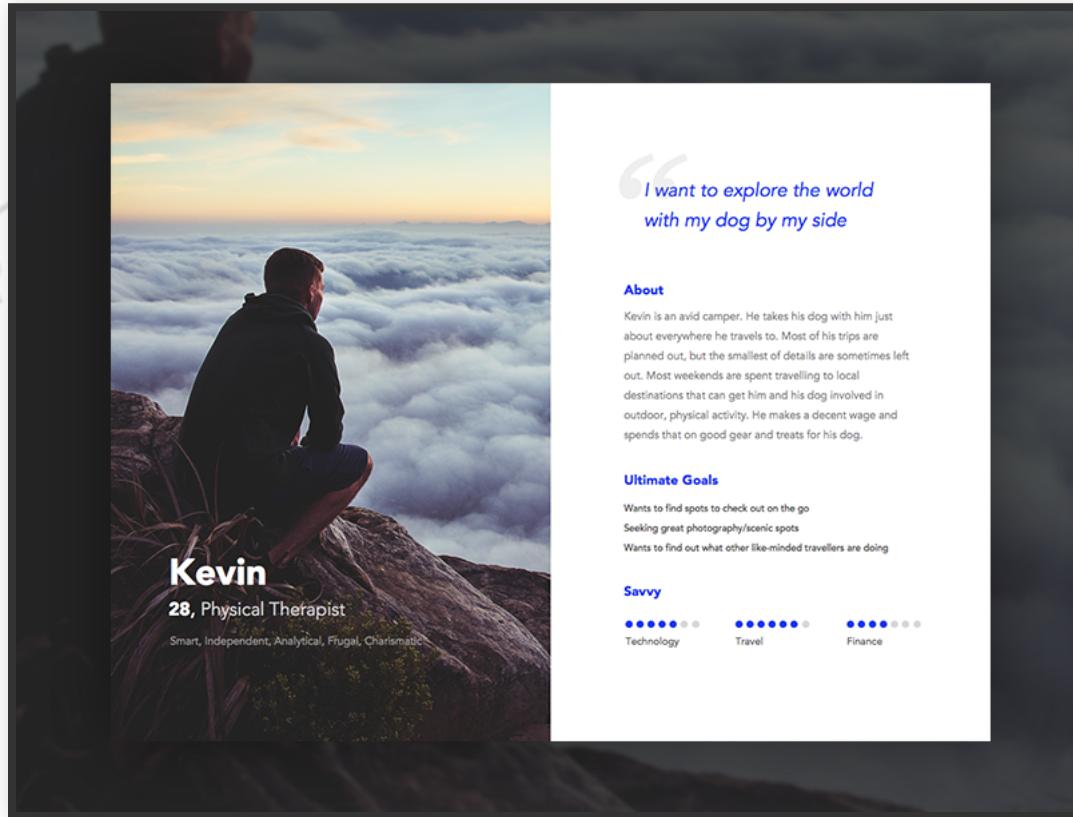
DETROIT

GRAND
CIRCUS

DETROIT

GRAND
CIRCUS

DETROIT



Kevin
28, Physical Therapist
Smart, Independent, Analytical, Frugal, Charismatic

I want to explore the world with my dog by my side

About
Kevin is an avid camper. He takes his dog with him just about everywhere he travels to. Most of his trips are planned out, but the smallest of details are sometimes left out. Most weekends are spent travelling to local destinations that can get him and his dog involved in outdoor, physical activity. He makes a decent wage and spends that on good gear and treats for his dog.

Ultimate Goals
Wants to find spots to check out on the go
Seeking great photography/scenic spots
Wants to find out what other like-minded travellers are doing

Savvy

Technology	Travel	Finance
•••••	•••••	•••••

GRAND
CIRCUS

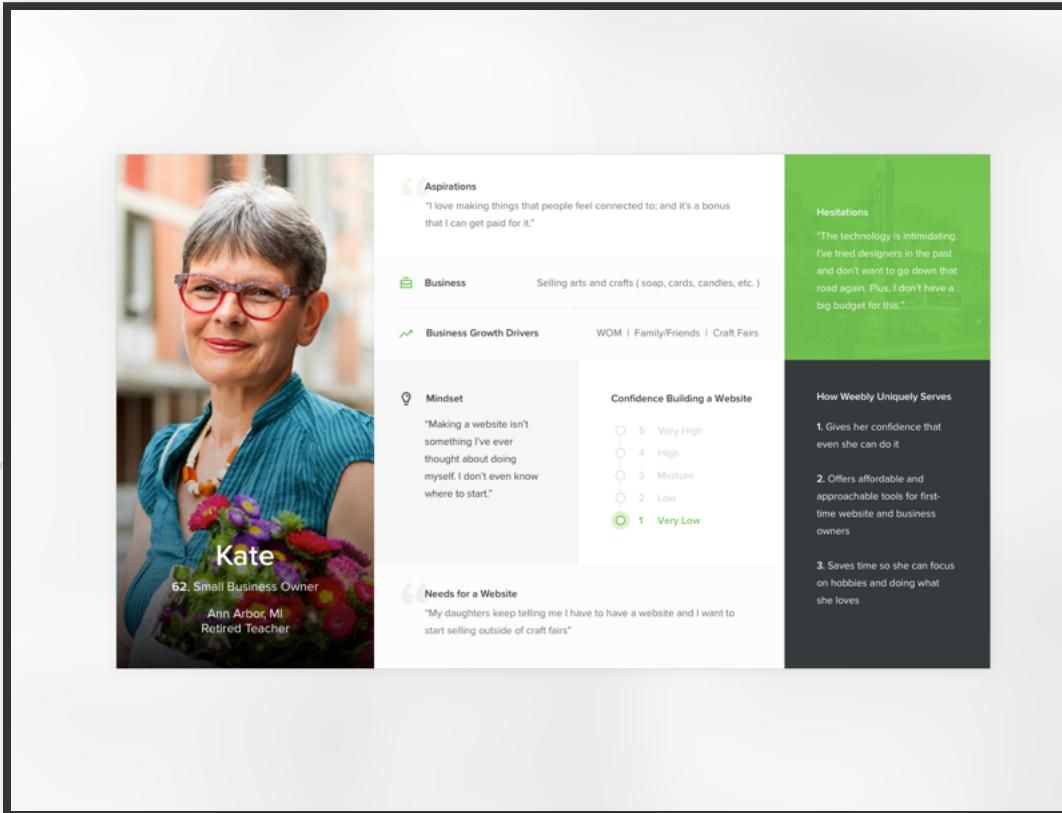
DETROIT

GRAND
CIRCUSGRAND
CIRCUS

DETROIT

GRAND
CIRCUS

USER PERSONA EXAMPLE



Kate
62, Small Business Owner
Ann Arbor, MI
Retired Teacher

Aspirations
"I love making things that people feel connected to; and it's a bonus that I can get paid for it."

Business
Selling arts and crafts (soap, cards, candles, etc.)

Business Growth Drivers
WOM | Family/Friends | Craft Fairs

Mindset
"Making a website isn't something I've ever thought about doing myself. I don't even know where to start."

Confidence Building a Website

<input type="radio"/>	5 Very High
<input type="radio"/>	4 High
<input type="radio"/>	3 Medium
<input type="radio"/>	2 Low
<input checked="" type="radio"/>	1 Very Low

Needs for a Website
"My daughters keep telling me I have to have a website and I want to start selling outside of craft fairs"

Hesitations
"The technology is intimidating. I've tried designers in the past, and don't want to go down that road again. Plus, I don't have a big budget for this."

How Weebly Uniquely Serves

1. Gives her confidence that even she can do it
2. Offers affordable and approachable tools for first-time website and business owners
3. Saves time so she can focus on hobbies and doing what she loves

DESIGN

Sketching

Creating wireframes

User Flow
Feedback



PROTOTYPING

- Paper Prototyping (Lo-Fi)
- Digital Prototyping (Hi-Fi)

Resource on Prototyping

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GR
CIR

PROTOTYPING

Digital Prototyping

GRAND
CIRCUS
DETROIT

GR
CIR

GRAND
CIRCUS

GRAND
CIRCUS
DETROIT



USER TESTING

Analytics (web traffic and sales analytics)

Test design

Offer solutions

UX ACTIVITY

A client wants to create a product to help dog owners find playmates for their pet.

What are some possible user needs for this product the client has not thought of?

Split into groups and brainstorm at least six possible user needs on post-it notes.



UX ACTIVITY

Prioritize the eight most essential user needs that should be addressed throughout the product.



CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

GRAND
CIRCUS
DETROIT

WIREFRAMING

GRAND
CIRCUS
DETROIT

WIREFRAME

A wireframe is a skeleton of what we want our site to look like.

Think of it as the blueprint or sketch of the layout the site should include.

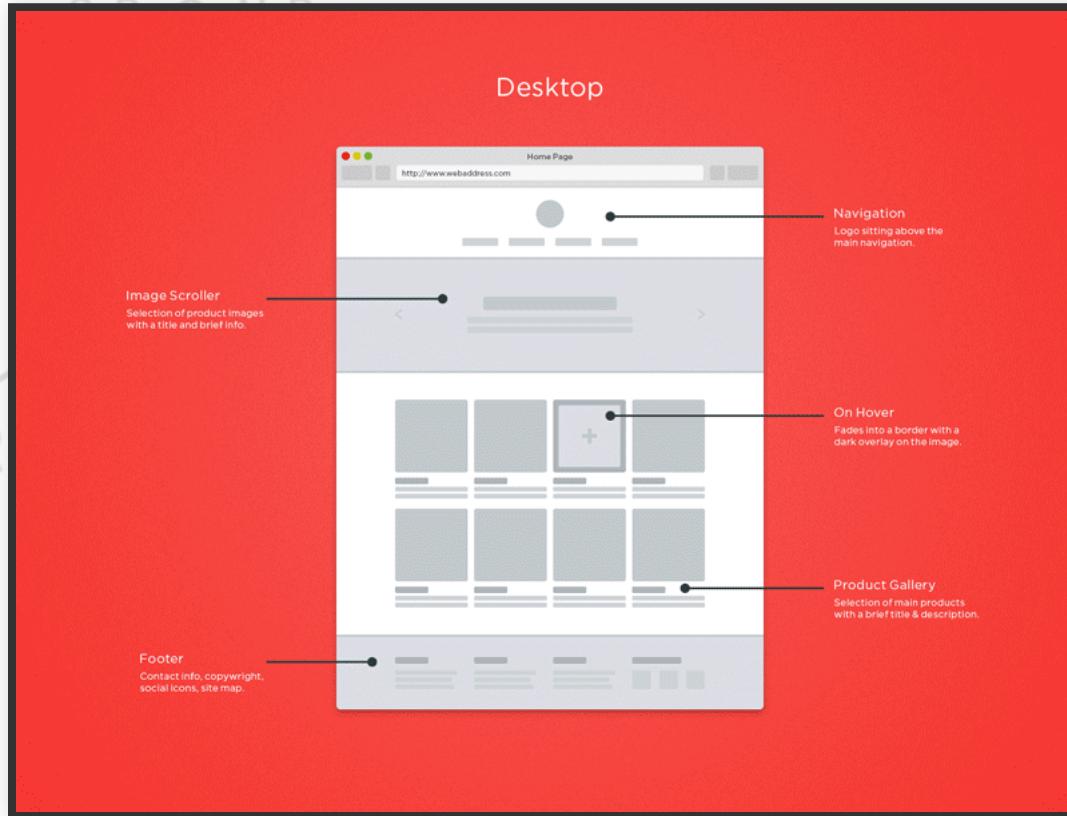
These are handy for us developers to use so we can visualize how sites are going to look before we start to code them.

WIREFRAME

We use wireframes so we can plan and architect our code before we even touch the keyboard.

Strategizing this way helps to eliminate future problems by thinking through possible issues in the beginning.

WIREFRAME EXAMPLE



WIREFRAME ACTIVITY

We are responsible for the creation of an event planning website.

Remember, we need a wireframe for mobile, tablet, and desktop.

What are some interactive features we can include?

WIREFRAME ACTIVITY

Let's go back to our dog matching site.

Based on our priorities, let's come up with a mobile and desktop wireframe.

Each group should take 10 minutes to sketch their interpretation of the home page structure based on user needs.

APPENDIX

Lessons in this section may be added when there is additional buffer time in the schedule.

GRAND
CIRCUS
DETROIT

HTML FORMS

GRAND
CIRCUS
DETROIT

FORMS

Forms are used to send data back to a server from user inputs.



FORM TAGS

```
<form action="some-url" method="get">  
  <!-- Form inputs and other html -->  
</form>
```



METHODS AND ACTIONS

Action is the URL you want the data to be sent to.

Method is the HTTP methods used to transfer data.

- GET
- POST
- And others

-DETROIT-

GRAND
CIRCUS
-DETROIT-

-DETROIT-

GRAND
CIRCUS
-DETROIT-

FORM TAGS

```
<input type="text" name="firstname" placeholder="Name?">
```

CIRCUS
-DETROIT-

GRAND
CIRCUS
-DETROIT-

CIRCUS
-DETROIT-

GRAND
CIRCUS
-DETROIT-

GRAND
CIRCUS
-DETROIT-

GRAND
CIRCUS
-DETROIT-

TYPES OF INPUTS

- Text (type="text")
- Radio Button (type="radio")
- Checkbox (type="checkbox")
- Password (type="password")
- Date (type="date")
- Time (type="time")
- Date & Time (type="datetime")
- Email (type="email")
- Submit
- `<input type="submit"/>`
- `<button type="submit"/>`



DEMO



TEXT

```
<input type="text"/> Starts out empty.  
<input type="text" value="Hello"/> Starts out with 'Hello' typed in.
```

The 'value' attribute determines what text it starts out with.



CHECKBOX

```
<input type="checkbox" checked/> I'm checked.  
<input type="checkbox"/> I'm NOT checked.
```

The presence or absence of a 'checked' attribute determines whether it starts out checked.



RADIO BUTTONS

```
<input type="radio" name="group-name" checked/> I'm checked.  
<input type="radio" name="group-name"/> I'm NOT checked.
```

Radio buttons that have the same 'name' are grouped together. Only one of the buttons in a group can be checked at a time.

SELECT (DROP-DOWN)

```
<select>
<option>Red</option>
<option selected="">White</option>
<option>Blue</option>
</select>
```

One of the options can have the 'selected' attribute, which pre-selects that option.

LABELS

```
<!-- input nested inside label -->  
<label><input type="text"/> Describe the input</label>  
  
<!-- input id specified in label 'for' attribute -->  
<input type="text" id="color"/><label for="color">Color</label>
```

It's good to associate a label with an input. Here are two ways.

LABELS

```
<!-- input nested inside label -->
<label><input type="checkbox" checked/> Click Me!</label>

<!-- input id specified in label 'for' attribute -->
<input type="checkbox" id="hasDog"/><label for="hasDog">Has Dog</label>
```

It's especially important to give checkboxes and radio buttons a label. Then the user can click either the label or the checkbox.

FORMS ACTIVITY

Open a new JS Bin

- Add a form.
- Add a checkbox. Make sure it has a clickable label.
- Add a group of two or more radio buttons. Make sure they have labels.
- Add a dropdown (select) with at least two options.