

External/ Internal /Inline CSS

CSS rules

External

Internal

Inline

```
<link href="css/styles.css"  
type="text/css" rel="stylesheet" /> ...
```

Styles.css

```
<style type="text/css">
```

```
</style>
```

```
<span class="yellow">
```

...

```
</span>
```

CSS Text

Common default browser typefaces

Georgia

Times

Times New Roman

Arial

Verdana

Helvetica

CSS Text

Typefaces are subject to copyright

typefaces installed on the user's computer

Link to download the typeface

License fonts from font foundries

Convert font to images (logo, titles)

sIFR - Embed font into a Flash movie

Cufon- uses JavaScript to create SVG or VML version of the text

font-family:

```
<style type="text/css">
```

```
body {
```

```
    font-family: Georgia, Times, serif;}
```

```
h1, h2 {
```

```
    font-family: Arial, Verdana, sans-serif;}
```

```
.credits {
```

```
    font-family: "Courier New", Courier, monospace;}
```

```
</style>
```

font-size:

```
body {  
    font-family: Arial, Verdana, sans-serif;  
    font-size: 12px;}  
  
h1 {  
    font-size: 200%;}  
  
h2 {  
    font-size: 1.3em;}
```

font-size:

pixels – px

precise control over how much space the text takes

The default size of text in browsers is 16px

percentages - %

75% - the equivalent of 12px

200% - 32px

ems

An em is equivalent to the width of a letter m

points – pt

for printer friendly versions of web pages

font-size:

16 century – European typographers developed scale of text sizes

Changed little in the past 400 years

Photoshop, InDesign and Word offer the same sizes of text

8_{pt}
9_{pt}
10_{pt}
11_{pt}
12_{pt}
14_{pt}
18_{pt}
24_{pt}
36_{pt}
48_{pt}
60_{pt}
72_{pt}

font-size:

PIXELS

TWELVE PIXEL SCALE

h1	24px
h2	18px
h3	14px
body	12px

=

PERCENTAGES

h1	200%
h2	150%
h3	117%
body	75%

=

EMS

h1	1.5em
h2	1.3em
h3	1.17em
body	100%
p	0.75em

SIXTEEN PIXEL SCALE

h1	32px
h2	24px
h3	18px
body	16px

=

h1	200%
h2	150%
h3	133%
body	100%

=

h1	2em
h2	1.5em
h3	1.125em
body	100%
p	1em

font-size:

- Font size in pixels – true to size
- Pixels are relative to the resolution of the screen
800x600 vs 1280x800 screen resolution will appear different
- The default size of text in a web browser is 16 pixels.
Using percentages of this amount, create a scale where the default text size is different, and headings are sized in relation to this.
- Users can change the default size of text in their web browsers
- Preserve the scale but not the size

Typography options

- Are limited
- Replacements techniques

@font-face

sIFR

Canvas

@font-face

- allows to use a font not installed on the computer of the user
- by specifying a path to a copy of the font, which will be downloaded if it is not on the user's machine
- the license for the font must permit it to be used in this way

```
@font-face {  
    font-family: 'ChunkFiveRegular';  
    src: url('fonts/chunkfive.eot');  
h1, h2 {  
    font-family: ChunkFiveRegular, Georgia, serif;  
}
```

@font-face

- typeface designers do not allow to use their fonts as @font-face
- open source fonts:

www.fontsquirrel.com

www.fontex.org

www.openfontlibrary.org

- some fonts are only free for personal use (not in .com websites!)

@font-face

- commercial fonts for a fee:

www.typekit.com

www.kernest.com

www.fontspring.com

- Google open source font files on their servers:

www.google.com/webfonts

Replacement techniques

sIFR

- uses Flash to replace the text with a small vector representation of the desired glyphs
- requires a browser with the Adobe Flash Lite plugin

Canvas

- uses JavaScript and the HTML5 canvas element to render text to the device
- browsers must have good JavaScript and HTML5 support

Google Fonts - Free

Google Fonts

[More scripts](#) [About](#) [Analytics](#) [New to Google Fonts?](#)

1 font family shown

Word

Sentence

Paragraph

Poster

↺

↻

tauri

Preview Text: Grumpy wizards make toxic

Size: 28 px

Sorting: Trending

Filters:

All categories

Thickness

Slant

Width

★ Show Collection only

☒ Reset all filters/search

Script:

Latin

Styles:

☐ Show all styles

Normal 400 ☒ Use this style

Grumpy wizards make toxic brew for the evil Queen and Jack.

Tauri, 1 Style by [Yvonne Schüttler](#)



Remove from Collection



School of
Design

TypeKit

<https://typekit.com/>

install Creative Cloud Software
to sync the fonts to our desktop

At no extra fee for this sync, even if you do not subscribe to CC

The username and password are:

graphicdesignuic@gmail.com

Roeger3410

Font Formats

- different browsers support different font formats
- supply the font in several variations to reach all browsers

FontSquirrel can convert fonts into different formats:

www.fontsquirrel.com/fontface/generator

- Provides the CSS code for the @font-face rule

Font Formats

Font formats order:

1. eot
2. woff
3. ttf/otf
4. svg

Font Formats

```
@font-face {  
    font-family: 'ChunkFiveRegular';  
    src: url('fonts/chunkfive.eot');  
    src: url('fonts/chunkfive.eot?#iefix')  
    format('embedded-opentype'), url('fonts/chunkfive.woff')  
    format('woff'), url('fonts/chunkfive.ttf')  
    format('truetype'), url('fonts/  
    chunkfive.svg#ChunkFiveRegular')  
    format('svg');  
}
```

Font Formats

BROWSER	FORMAT			
	eot	woff	ttf / otf	svg
Chrome (all)				●
Chrome 6+		●	●	●
Firefox 3.5			●	
Firefox 3.6+		●	●	
IE 5 - 8	●			
IE 9+	●	●	●	
Opera 10+			●	●
Safari 3.1+			●	●
iOS <4.2				●
iOS 4.2+			●	●

font-weight:

```
.credits {  
    font-weight: bold;}
```

```
.maintext {  
    font-weight: normal;}
```

font-style:

- normal
- Italic
- oblique

```
.credits {  
    font-style: italic; }
```

text-transform:

- uppercase
- lowercase
- capitalize

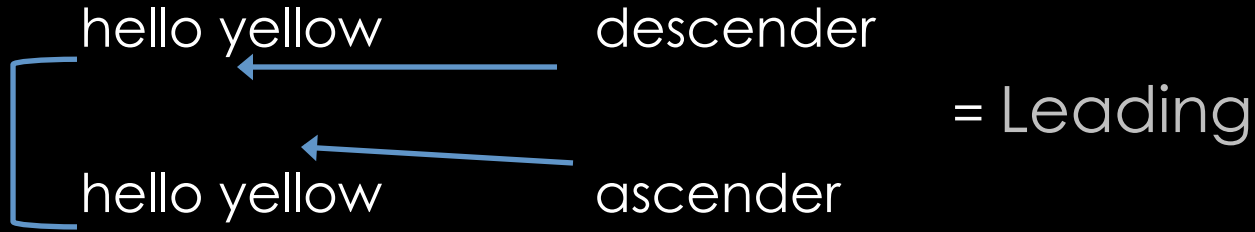
```
h1{  
    text-transform: uppercase; }  
h2{  
    text-transform: lowercase; }  
.credits{  
    text-transform: capitalize; }
```

text-decoration:

- none
- underline
- line-through
- blink

```
.credits{  
    text-decoration: underline; }  
A      {  
    text-decoration: none; }
```


line-height:



Font-size minus line-height = leading

```
p {  
    line-height: 1.4 em; }
```

letter-spacing: word-spacing:

letter-spacing: = kerning

```
h1, h2 {  
    text-transform: uppercase;  
    letter-spacing: 0.2em;}  
  
.credits {  
    font-weight: bold;  
    word-spacing: 1em;}
```

text-align:

- left
- right
- center
- Justify

```
h1 {  
    text-align: left;}  
  
p {  
    text-align: justify;}
```

vertical-align:

- Baseline
- Sub
- Super
- Top
- Text-top
- Middle
- Bottom
- Test-bottom

```
#six-months {  
    vertical-align: text-top;}  
#one-year {  
    vertical-align: baseline;}  
#two-years {  
    vertical-align: text-bottom;}
```

RESULT

Briard Life Stages



Six months



One year



Two years

text-indent:

Allows to indent the first line of text

```
h1 {  
  background-image: url("images/logo.gif");  
  background-repeat: no-repeat;  
  text-indent: -9999px;}  
.credits {  
  text-indent: 20px;}
```

RESULT

Briards

by Ivy Duckett

The briard, or berger de brie, is a large breed of dog traditionally used as a herder and guardian of sheep.

Breed History

The briard, which is believed to have originated in France, has been bred for centuries to herd and to protect sheep. The breed was used by the French Army as sentries, messengers and to search for wounded soldiers because of its fine sense of hearing. Briards were used in the First World War almost to the point of extinction. Currently the population of briards is slowly recovering. Charlemagne, Napoleon, Thomas Jefferson and Lafayette all owned

CSS3 – text-shadow:

- 1 - how far to the left or right the shadow should fall
- 2 - the distance to the top or bottom that the shadow should fall
- 3 - specifies the amount of blur that should be applied to the drop shadow
- 4 - the color of the drop shadow

```
p.three {  
    background-color: #cccccc;  
    color: #ffffff;  
    text-shadow: 2px 2px 7px #111111;}
```

Pseudo elements - :first-letter :first-line

```
p.intro:first-letter {  
    font-size: 200%;}
```

```
p.intro:first-line {  
    font-weight: bold;}
```

Pseudo classes - :link, :visited, :hover, :active, :focus

Pseudo classes allow to change appearance of elements upon interaction

:link

styles for links that have not yet been visited

:visited

styles for links that have been visited

:hover

a user hovers over an element with a mouse cursor

:active

when an element is activated by a user (button is pressed)

:focus

when an element has focus

Pseudo classes - :link, :visited, :hover, :active, :focus

```
a:link {  
    color: deeppink;  
    text-decoration: none;}  
a:visited {  
    color: black;}  
a:hover {  
    color: deeppink;  
input {  
    padding: 6px 12px 6px 12px; border: 1px solid #665544; color: #ffffff;}  
input.submit:hover {  
    background-color: #665544;}
```

Attribute Selectors

allow to create rules that apply to elements that have an attribute with a specific value

SELECTOR	MEANING	EXAMPLE
EXISTENCE	<code>[]</code> Matches a specific attribute (whatever its value)	<code>p[class]</code> Targets any <code><p></code> element with an attribute called <code>class</code>
EQUALITY	<code>[=]</code> Matches a specific attribute with a specific value	<code>p[class="dog"]</code> Targets any <code><p></code> element with an attribute called <code>class</code> whose value is <code>dog</code>
SPACE	<code>[~=]</code> Matches a specific attribute whose value appears in a space-separated list of words	<code>p[class~="dog"]</code> Targets any <code><p></code> element with an attribute called <code>class</code> whose value is a list of space-separated words, one of which is <code>dog</code>

Attribute Selectors

PREFIX

[^=]

Matches a specific attribute whose value begins with a specific string

p[attr^="d"]

Targets any <p> element with an attribute whose value begins with the letter "d"

SUBSTRING

[*=]

Matches a specific attribute whose value contains a specific substring

p[attr*"do"]

Targets any <p> element with an attribute whose value contains the letters "do"

SUFFIX

[\$=]

Matches a specific attribute whose value ends with a specific string

p[attr\$"g"]

Targets any <p> element with an attribute whose value ends with the letter "g"

CSS Selectors

Simple attribute selector

selects all elements with the matching value

```
p[class] {color: blue}
```

CSS Selectors

Advanced attribute selector

selects elements with matching substrings in the value of the attribute. (existing attributes or created custom attributes)

caret (^) - begins with the value "iphone":

```
p[device^="iphone"] {background: green;}
```

dollar sign or string (\$) - ends with the value "iphone":

```
p[device$="iphone"] {background: blue;}
```

star (*) - contains at least one instance of "lcd":

```
p[device*="lcd"] {background: red;}
```

CSS Selectors

Selector selects markup elements it should apply rules to

- Universal
- Type or element selectors
- Contextual selectors (descendant, child, and adjacent sibling)
- Class
- ID
- Pseudoclasses
- Pseudoelements

CSS Selectors

Universal selector

selects all elements useful for defining the default typeface or font size (body, html)

```
{font-family:sans-serif;}
```

Type or element selector

selects specific element

```
h1 {color: red;}
```

CSS Selectors

Descendent selector

selects elements that are descendants of another element as indicated by a single space separation

```
li a {color: black;}
```

Child selector

similar to the descendent selector

selects only the child of the defined element, not all instances like the descendent selector

```
p > em {background-color: yellow;}
```


CSS Selectors

Pseudoselectors

select elements that might not exist in the markup

:link

:visited

:hover

:active

:before

:after

:first-child

:last-child

CSS Selectors

Adjacent sibling selector

Selects an element that comes immediately after another element with the same parent.

```
h1 + p { margin-bottom: 1em;}
```

CSS Selectors Commonly used in Mobile Dev.

Class selector

The class selector targets any element with the matching class

```
.red {background-color: red;}
```

ID selector

selects any element with the matching ID

```
#alert {background-color: yellow;}
```

CSS Text

- properties to control the choice of font, size, weight, style, and spacing
- limited choice of fonts that most people will have installed
- wider range of typefaces involves copyright and the right license issues
- can control the space between lines of text, individual letters, and words
- text can be aligned to the left, right, center, or justified, or indented
- pseudo-classes can change the style of an interactive element

Text / Color / Selectors / exercise3.html

Ivory

```
<span class="yellow"> Yellow </span>
```

Lavender

Orange

Golden

Ivory

Yellow

Lavender

Orange

Golden

<http://tomastranstromer.net/poetry/poetry-3/>

Alternate:

RGB vs # hex values

Class vs ID selectors

Text / Color / Selectors / exercise3.html

font-family

font-size

@font-face

font-weight

font-style

font-transform

font-decoration

line-height

letter-spacing

word-spacing

text-align

vertical-align

text-indent

text-shadow

CSS selectors

class

ID

universal

type / element

pseudoclasses

pseudoelements

Colors

RGB and #hex

RGBA

CSS Boxes

CSS treats each HTML element as if it is in its own box

Control the appearance of each box:

- Dimensions

- Borders

- Margins & padding

- Show & hide

CSS Boxes

Border

Margins – outside the border

Padding – space between the border and the content



CSS Boxes

height: 300px;

width: 300px;

min-width: 450px;

max-width: 650px;

min-height: 10px;

max-height: 30px;

overflow: hidden;

overflow: scroll;

CSS Boxes

border-width:

thin

medium

thick

p.one {

border-width: 2px;}

p.two {

border-width: thick;}

p.three {

border-width: 1px 4px 12px 4px;}

border-top-width

border-right-width

border-bottom-width

border-left-width

border-style:

```
p.one {border-style: solid;}  
p.two {border-style: dotted;}  
p.three {border-style: dashed;}  
p.four {border-style: double;}  
p.five {border-style: groove;}  
p.six {border-style: ridge;}  
p.seven {border-style: inset;}  
p.eight {border-style: outset;}
```

border-color:

border-top-color

border-right-color

border-bottom-color

border-left-color

```
p.one {  
    border-color: #0088dd;}
```

```
p.two {  
    border-color: #bbbbbaa #111111 #ee3e80 #0088dd;}
```

border:

width, style and color in that specific order

```
p {  
  width: 250px;  
  border: 3px dotted #0088dd;}
```

padding:

Space between the content of an element and its border

Most often - px

padding-top

padding-right

padding-bottom

padding-left

```
p.example {  
    padding: 10px;}
```

margin:

Space between the boxes

Most often - px

margin-top

margin-right

margin-bottom

margin-left

```
p.example {  
    margin: 20px;}
```

margin:

Space between the boxes (most often – px)

margin-top

margin-right

margin-bottom

margin-left

clockwise order: top, right, bottom, left

```
p.example {  
    margin: 20px;}
```

```
p.example2 {  
    margin: 1px 2px 3px 4px; }
```


Centering boxes

- set a width for the box (otherwise it will take up the full width of the page)
- setting the left and right margins to auto will make the browser put an equal gap on each side of the box
- for older browsers the element that the box sits inside should have a text-align property with its value set to center

display:

converts inline elements into a block-level elements or vice versa

inline

block

inline-block

causes a block-level element to flow like an inline element, while retaining other features of a block-level element

none

hides element from the page

display:

```
<ul>
<li>Home</li>
<li>Products</li>
<li class="coming-soon">Services</li>
<li>About</li>
<li>Contact</li>
</ul>
```

```
li {
    display: inline;
    margin-right: 10px;}
li.coming-soon {
    display: none;}
```

About Schools News Contact

display:

hide boxes but leaves a space where the elements would have been

a blank space will appear in place of an element

hidden

visible

visibility:

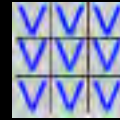
```
<ul>
<li>Home</li>
<li>Products</li>
<li class="coming-soon">Services</li>
<li>About</li>
<li>Contact</li>
</ul>
```

```
li {
    display: inline;
    margin-right: 10px;}

li.coming-soon {
    visibility: hidden;}
```

CSS3 border-image:

applies an image to the border of any box
background image is sliced it into 9 pieces



1: The URL of the image

2: Where to slice the image

3: What to do with the straight edges;

the possible values are:

stretch stretches the image repeat repeats the image round like
repeat but if the tiles do not fit exactly, scales the tile image

CSS3 box-shadow:

Drops shadow around a box

inset – crates inner shadow

horizontal offset - Negative values position the shadow to the left

vertical offset - Negative values position the shadow to the top

blur distance - If omitted, the shadow is a solid line like a border

spread of shadow -positive value will cause the shadow to expand in all directions, and a negative value will make it contract

CSS3 border-radius:

Creates rounded corners on any box

The value indicates the size of the radius in pixels

border-top-right-radius

border-bottom-right-radius

border-bottom-left-radius

border-top-left-radius

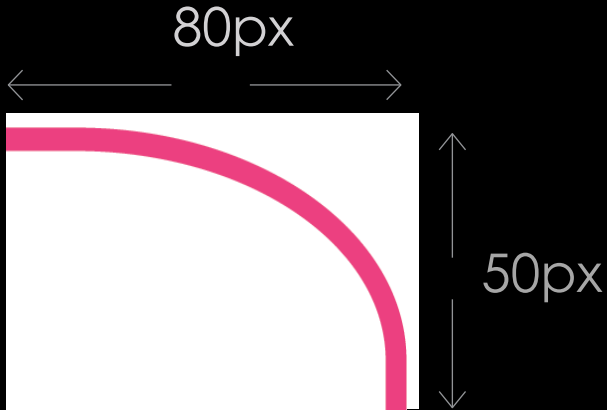
```
p {  
  border: 5px solid #cccccc;  
  padding: 20px;  
  width: 275px;  
  text-align: center;  
  border-radius: 10px;  
  -moz-border-radius: 10px;  
  -webkit-border-radius: 10px;}
```


CSS3 elliptical shapes

more complex elliptical shapes

specify different distances for the horizontal and the vertical parts of the rounded corners

`border-radius: 80px 50px;`



CSS list properties

list-style-type: - controls the shape/style of a bullet

none

disc

circle

square

CSS list properties

decimal

1 2 3

upper-alpha

A B C

decimal-leading-zero

01 02 03

lower-roman

i. ii. iii.

lower-alpha

a b c

upper-roman

I II III

CSS list properties

list-style-image: - specifies an image to act as a bullet point

```
ul {  
    list-style-image: url("images/star.png");  
li {  
    margin: 10px 0px 0px 0px;}
```

CSS list properties

list-style-position: - specifies if the marker appears on the inside or the outside of the box containing the main points

outside inside

```
ul {  
    width: 150px;}  
  
li {  
    margin: 10px;}  
  
ul.illuminations {  
    list-style-position: outside;}  
  
ul.season {  
    list-style-position: inside;}
```

CSS list properties

list-style: - addresses all the above (shorthand)

```
ul {  
    list-style: inside circle;  
    width: 300px;}  
  
li {  
    margin: 10px 0px 0px 0px;}
```

CSS table properties

width:

padding:

text-transform: converts the content of the table headers to uppercase

letter-spacing, font-size: adds additional styling to the content of the table headers

border-top, border-bottom: sets borders above and below the table headers

text-align:

background-color:

:hover - highlights a table row when a user's mouse goes over it

CSS table styling tips

padding to add space between cells

distinguish headings (<th> or bold, etc.)

shade alternate rows for legibility

align numerals

CSS table styling tips

empty-cells: specifies if empty table cell borders are shown

show

hide

inherit - the nested table cells will obey the rules of the container

```
td {  
    border: 1px solid #0088dd;  
    padding: 15px;}  
table.one {  
    empty-cells: show;}  
table.two {  
    empty-cells: hide;}
```

CSS table styling tips

border-spacing: controls the distance between adjacent cells

border-collapse: collapses adjacent borders of inside cells to prevent the width of lines twice that of the outside edges

collapse

separate

CSS layout

Layout

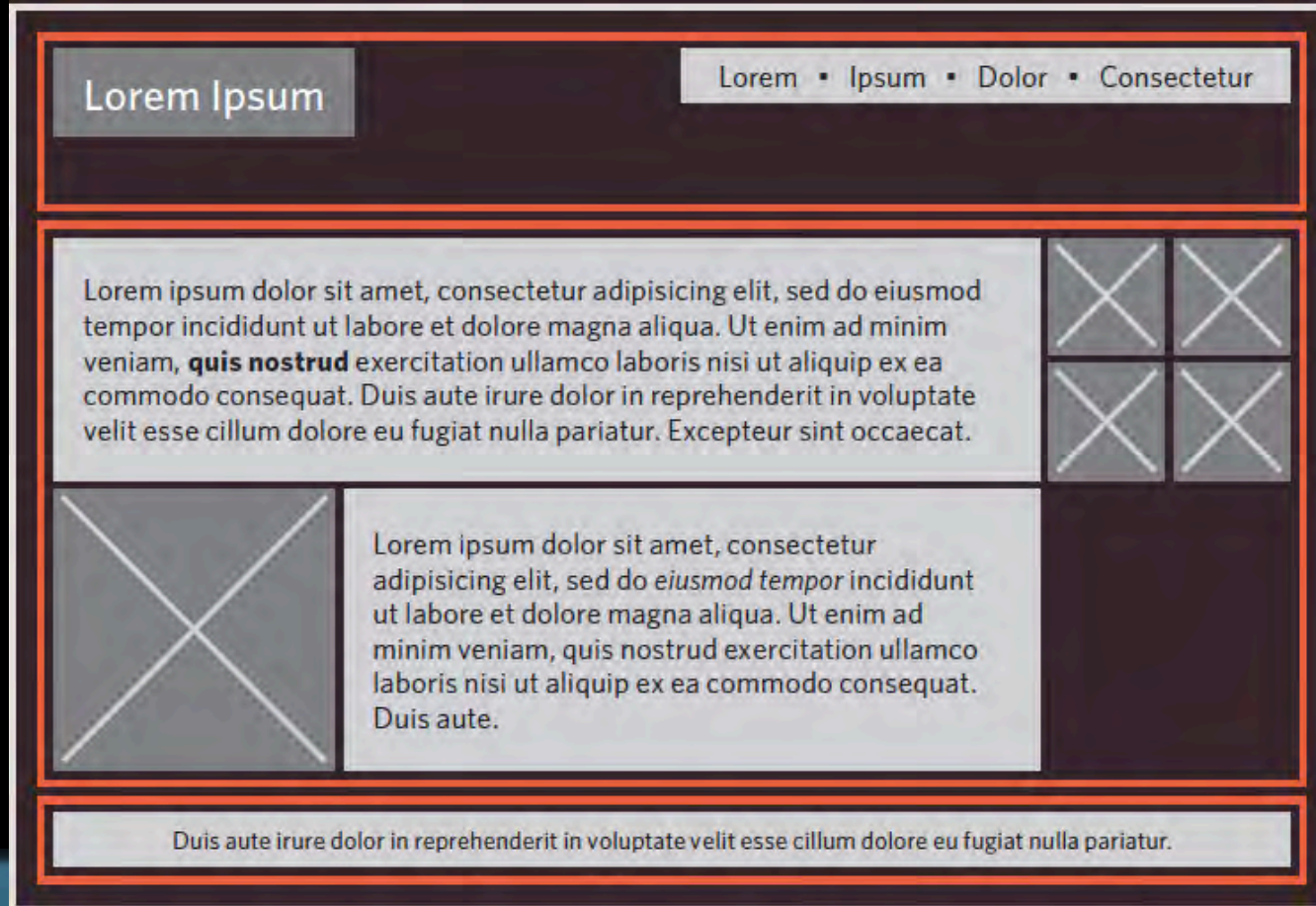
Composition

Positioning

CSS layout

grouping
a number
of elements
together inside
a block - `<div>`

parent
child



CSS positioning

normal flow

Relative

Absolute

Fixed

Floating

Normal Flow

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.

Relative

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea. Duis aute irure dolor in reprehenderit in voluptate velit.

Absolute
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.

Fixed
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.

Floating
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.

CSS positioning

normal flow (static)

- Every block-level element appears on a new line, causing each item to appear lower down the page than the previous one - default

relative positioning

- A relative positioned element is positioned relative to its normal position
- does not affect the position of surrounding elements; they stay in the position they would be in in normal flow

CSS positioning

absolute positioning

- An absolute position element is positioned relative to the first parent element that has a position other than static. If no such element is found, the containing block is `<html>`
- Removed from the normal flow
- The document and other elements behave like the absolutely positioned element does not exist
- can overlap other elements
- Due to the way different mobile browsers treat the viewport, fixed positioning can be somewhat unreliable

CSS positioning

fixed positioning

- An element with fixed position is positioned relative to the browser window
- will not move even if the window is scrolled

floating elements

- Floated element is taken out of normal flow and positioned
- it to the far left or right of a containing box
- other content can flow around

CSS positioning

fixed positioning

- An element with fixed position is positioned relative to the browser window
- will not move even if the window is scrolled

floating elements

- Floated element is taken out of normal flow and positioned
- it to the far left or right of a containing box
- other content can flow around
- not encouraged in mobile development

CSS stacking order

Z- index

used to determine the stacking order of positioned elements

`z-index: 10;`

used to overlay elements on top of each other to create a specific visual effect