

CSS

Lab 2

Positioning and Layout

1	CSS BOX MODEL	1
1.1	DIMENSIONS (WIDTH, HEIGHT).....	2
1.2	PADDING	2
1.2.1	<i>box-sizing</i>	2
1.3	BORDER.....	3
1.3.1	<i>Styling borders (border-radius, border-image)</i>	3
1.3.2	<i>List of all border properties</i>	3
1.4	MARGIN	3
1.4.1	<i>negative margins</i>	4
1.4.2	<i>collapsing margins</i>	4
2	CSS LAYOUTS	4
2.1	DISPLAY	4
2.2	POSITION	4
2.2.1	<i>Layering with z-index</i>	5
2.3	FLOAT	5
2.4	ALIGNMENT TECHNIQUES.....	5
3	FLEXBOX.....	6
3.1	FLEXBOX FOR RESPONSIVE WEB DESIGN (RWD).....	6
4	GRID	6
4.1	GRID FOR RESPONSIVE WEB DESIGN (RWD)	7

1 CSS Box Model

The Box Model is central in CSS when considering placement and layout of the various elements on a web page.

Every HTML element can be considered to have a box that wraps around it and the properties of these boxes and how they interact with each other determine how HTML elements are positioned relative to each other.

<https://www.tutorialrepublic.com/css-tutorial/css-box-model.php>

Note:

The margin (`margin-top`, `margin-bottom`, `margin-left`, `margin-right`) is the space between the borders of two neighboring elements

The padding (`padding-top`, `padding-bottom`, `padding-left`, `padding-right`) is the space between the content area of the element and its border

The border of the box is usually of uniform thickness, but you can customize the properties of each of the borders (including its thickness) with the `border-top`, `border-left`, `border-bottom` and `border-right` properties.

Note: The `<div>` element is often used to provide group one or more elements together because the various box modeling properties can be applied to it very clearly.

For the demo, you can create 2 successive `<div>` elements to understand the margin property better. Try playing around with the various properties and see how it changes the placement of the two boxes corresponding to the 2 `<div>`s

<https://www.tutorialrepublic.com/css-tutorial/css-visual-formatting.php>

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

1.1 dimensions (width, height)

<https://www.tutorialrepublic.com/css-tutorial/css-dimension.php>

All Topics: width, height, max-width, min-width, max-height, min-height

1.2 padding

<https://www.tutorialrepublic.com/css-tutorial/css-padding.php>

All Topics: padding-top, padding-right, padding-bottom, padding-left, padding

To prevent padding and border from changing element's box width and height, you can use the box-sizing property

1.2.1 box-sizing

<https://www.tutorialrepublic.com/css-tutorial/css3-box-sizing.php>

All topics: Redefining Box Width with Box-Sizing, Creating Layouts with Box-Sizing

https://www.w3schools.com/css/css3_box-sizing.asp

1.3 border

<https://www.tutorialrepublic.com/css-tutorial/css-border.php>

All topics: border-style, border-width, border-color, border

To specify border-style, border-width, border-color for individual sides

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

NOTE: None of the other CSS border properties will have ANY effect unless the border-style property is set

<https://www.tutorialrepublic.com/css-tutorial/css-color.php>

Affect of color Property on Borders and Outlines

1.3.1 Styling borders (border-radius, border-image)

<https://www.tutorialrepublic.com/css-tutorial/css3-border.php>

All topics: border-radius, border-image

Other examples of these 2 properties:

https://www.w3schools.com/css/css3_borders.asp

https://www.w3schools.com/css/css3_border_images.asp

For checking effects of various values for the border-radius property

<https://border-radius.com/>

1.3.2 List of all border properties

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

Topic: All CSS Border Properties

1.4 margin

<https://www.tutorialrepublic.com/css-tutorial/css-margin.php>

All topics: margin-top, margin-right, margin-bottom, margin-left, margin

`auto` property calculates the margin to center an element horizontally within a parent container

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

Topic: The inherit Value

1.4.1 negative margins

https://www.quirksmode.org/blog/archives/2020/02/negative_margin.html

Common use:

<https://css-tricks.com/negative-margins/>

For a <div>, the padding is from the border of the div into the child elements within the <div>

1.4.2 collapsing margins

<https://www.freecodecamp.org/news/what-is-margin-collapse-and-how-to-avoid-it/>

<https://css-tricks.com/what-you-should-know-about-collapsing-margins/>

2 CSS Layouts

https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Introduction

2.1 display

<https://www.tutorialrepublic.com/css-tutorial/css-display.php>

All topics: display block, display inline, display inline-block, none

<https://www.tutorialrepublic.com/css-tutorial/css-visibility.php#visibility-vs-display>

Another similar example to the previous

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

Detailed explanation of all possible values for the `display` property

<https://www.freecodecamp.org/news/the-css-display-property-display-none-display-table-inline-block-and-more/>

2.2 position

<https://www.tutorialrepublic.com/css-tutorial/css-position.php>

All topics: static positioning, relative positioning, absolute positioning, fixed positioning, (top, bottom, left and/or right)

Similar examples:

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

More detailed examples:

<https://css-tricks.com/almanac/properties/p/position/>

2.2.1 Layering with z-index

<https://www.tutorialrepublic.com/css-tutorial/css-layers.php>

<https://www.tutorialrepublic.com/css-reference/css-z-index-property.php>

More examples:

https://www.w3schools.com/css/css_z-index.asp

2.3 float

<https://www.tutorialrepublic.com/css-tutorial/css-float.php>

Topic: float, clear

More examples:

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

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

Demonstrate `overflow:auto` and `.clearfix::after`

<https://www.tutorialrepublic.com/css-tutorial/css-alignment.php>

Topic: Clearing floats

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

2.4 Alignment techniques

There various techniques to align text, images and other elements either horizontally or vertically

<https://www.tutorialrepublic.com/css-tutorial/css-alignment.php>

<https://www.freecodecamp.org/news/css-vertical-align-how-to-center-a-div-text-or-an-image-example-code/>

<https://blog.hubspot.com/website/center-an-image-in-html>

https://www.w3schools.com/cssref/pr_pos_vertical-align.php

vertical-align property

3 Flexbox

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

Key properties:

Display: flex, flex-direction, flex-wrap, flex-flow, justify-content, align-items, align-content, order, flex-grow, flex-shrink, flex-basis, flex, gap, row-gap, column-gap

More examples:

https://www.w3schools.com/css/css3_flexbox.asp

Gameplay demo (you will need to sign up to codepip first)

<https://codepip.com/games/flexbox-froggy/>

Direct demo

<https://demo.agektmr.com/flexbox/>

There is relevant functionality in Chrome Dev tools to work with the Flexbox layout.

<https://developer.chrome.com/docs/devtools/css/flexbox/>

3.1 Flexbox for Responsive Web Design (RWD)

Flexbox and Grid are two of the major layout techniques for RWD (covered in Lab 3)

https://www.w3schools.com/css/css3_flexbox_responsive.asp

Step by step project:

<https://www.freecodecamp.org/learn/2022/responsive-web-design>

Project: Learn CSS flexbox by building a Photo Gallery

<https://coder-coder.com/build-flexbox-website-layout/>

You can use some of the features for simulating mobile devices in Chrome Dev Tools to help you with RWD

<https://developer.chrome.com/docs/devtools/device-mode/>

4 Grid

Complete reference

<https://css-tricks.com/snippets/css/complete-guide-grid/>

Various example corresponding to various property values

<https://learncssgrid.com>

Gameplay demo (you will need to sign up to codepip first)

<https://cssgridgarden.com/>

Various examples:

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

<https://gridbyexample.com/examples/>

There is relevant functionality in Chrome Dev tools to work with the Grid layout.

<https://developer.chrome.com/docs/devtools/css/grid/>

4.1 Grid for Responsive Web Design (RWD)

Flexbox and Grid are two of the major layout techniques for RWD (covered in Lab 3)

<https://blog.logrocket.com/creating-responsive-web-layouts-with-css-grid>

Step by step project:

<https://www.freecodecamp.org/learn/2022/responsive-web-design>

Learn CCS Grid by Building a Magazine

You can use some of the features for simulating mobile devices in Chrome Dev Tools to help you with RWD

<https://developer.chrome.com/docs/devtools/device-mode/>