

Web Engineering LAB



Lab # 04

**CSS: Selector, Color, Text,
Boxes, & Layout**

Instructor: Hurmat Hidayat

Email: hurmathidayat@nu.edu.pk

Course Code: SL3003

Semester Spring 2022

**Department of Computer Science,
National University of Computer and Emerging Sciences
FAST Peshawar Campus**

Content

CSS SELECTOR	3
HOW CSS RULES CASCADE	4
INHERITANCE	5
COLORS	6
RGB VALUES	6
HEX CODES	6
COLOR NAMES	6
BACKGROUND COLOR	7
CONTRAST	7
CSS 3: OPACITY	8
CSS 3: HSL COLORS	9
TEXT	9
FONT VS TYPEFACE	10
TYPEFACE TERMINOLOGY	10
SPECIFYING TYPEFACES	11
SIZE OF TYPE	12
MORE FONT CHOICE	12
RESPONDING TO USERS	13
BOXES	15
BOX DIMENSION	15
BORDER, MARGIN & PADDING	16
BORDER WIDTH	16
BORDER STYLE	18
PADDING	18
MARGIN	18
CHANGE INLINE/BLOCK	19
LAYOUT	20
A FIXED WIDTH LAYOUT	20
LAB TASKS:	22

CSS Selector

There are many different types of CSS selector that allow you to target rules to specific elements in an HTML document. CSS selectors are case sensitive, so they must match element names and attribute values exactly.

The following table introduces the most commonly used CSS selectors.

SELECTOR	MEANING	EXAMPLE
UNIVERSAL SELECTOR	Applies to all elements in the document	* {} Targets all elements on the page
TYPE SELECTOR	Matches element names	h1, h2, h3 {} Targets the <h1>, <h2> and <h3> elements
CLASS SELECTOR	Matches an element whose class attribute has a value that matches the one specified after the period (or full stop) symbol	.note {} Targets any element whose class attribute has a value of note .p.note {} Targets only <p> elements whose class attribute has a value of note
ID SELECTOR	Matches an element whose id attribute has a value that matches the one specified after the pound or hash symbol	#introduction {} Targets the element whose id attribute has a value of introduction
CHILD SELECTOR	Matches an element that is a direct child of another	li>a {} Targets any <a> elements that are children of an element (but not other <a> elements in the page)
DESCENDANT SELECTOR	Matches an element that is a descendent of another specified element (not just a direct child of that element)	p a {} Targets any <a> elements that sit inside a <p> element, even if there are other elements nested between them
ADJACENT SIBLING SELECTOR	Matches an element that is the next sibling of another	h1+p {} Targets the first <p> element after any <h1> element (but not other <p> elements)
GENERAL SIBLING SELECTOR	Matches an element that is a sibling of another, although it does not have to be the directly preceding element	h1~p {} If you had two <p> elements that are siblings of an <h1> element, this rule would apply to both

How Css Rules Cascade

If there are two or more rules that apply to the same element, it is important to understand which will take precedence.

- If the two selectors are identical, the latter of the two will take precedence.
- If one selector is more specific than the others, the more specific rule will take precedence over more general ones. In the example:
 - h1 is more specific than *
 - p b is more specific than p
 - p#intro is more specific than p
- You can add !important after any property value to indicate that it should be considered more important than other rules that apply to the same element.

Example

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>CSS Selectors</title>
5      <link href="css_selectors.css" type="text/css" rel="stylesheet" />
6  </head>
7  <body>
8  <h1>
9      Potatoes
10 </h1>
11 <p id="intro">
12     There are <i>dozens</i> of different <b>potato</b> varieties.
13 </p>
14 <p>
15     They are usually described as early, second early
16 and maincrop potatoes.</p>
17 </body>
18 </html>
19
20 *
21 font-family: Arial, Verdana, sans-serif; }
22
23 h1 {
24 font-family: "Courier New", monospace; }
25
26 i {
27 color: green; }
28
29 i {
30 color: red; }
31
32 b {
33 color: pink; }
34
35 p b {
36 color: blue !important; }
37
38 p b {
39 color: violet; }
40
41 p#intro {
42 font-size: 100%; }
43
44 p {
45 font-size: 75%; }
```

Potatoes

There are *dozens* of different **potato** varieties.

They are usually described as early, second early and maincrop potatoes.

Inheritance

If you specify the font-family or color properties on the <body> element, they will apply to most child elements. This is because the value of the font-family property is inherited by child elements. You can compare this with the background-color or border properties; they are not inherited by child elements. If these were inherited by all child elements then the page could look quite messy.

You can force a lot of properties to inherit values from their parent elements by using inherit for the value of the properties.

Example

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>CSS Selectors</title>
5      <link href="css_selectors.css" type="text/css" rel="stylesheet" />
6  </head>
7  <body>
8  <div class="page">
9      <h1>
10         Potatoes
11     </h1>
12     <p>
13         There are dozens of different potato varieties.
14     </p>
15     <p>
16         They are usually described as early, second early and maincrop potatoes.
17     </p>
18  </div>
19  </body>
20  </html>
```

```
1  body
2  {
3      font-family: Arial, Verdana, sans-serif;
4      color: #665544;
5      padding: 10px;
6  }
7
8  .page
9  {
10      border: 1px solid #665544;
11      background-color: #efefef;
12      padding: inherit;
13 }
```

Potatoes

There are dozens of different potato varieties.

They are usually described as early, second early and maincrop potatoes.

Colors

Color not only brings yours site to life, but also helps convey the mood and evokes reactions. The color property allows you to specify the color of text inside an element. You can specify any color in CSS in one of three ways:

rgb values

These express colors in terms of how much red, green and blue are used to make it up.

For example: `rgb(100,100,90)`

hex codes

These are six-digit codes that represent the amount of red, green and blue in a color, preceded by a pound or hash # sign.

For example: `#ee3e80`

color names

There are 147 predefined color names that are recognized by browsers.

For example: `DarkCyan`

```
1  /* color name */
2  h1 {
3      color: DarkCyan;
4  }
5
6  /* hex code */
7  h2 {
8      color: #ee3e80;
9  }
10
11 /* rgb value */
12 p {
13     color: rgb(100,100,90);
14 }
```

RESULT

Marine Biology

The Composition of Seawater

Almost anything can be found in seawater. This includes dissolved materials from Earth's crust as well as materials released from organisms. The most important components of seawater that influence life forms are salinity, temperature, dissolved gases (mostly oxygen and carbon dioxide), nutrients, and pH. These elements vary in their composition as well as in their influence on marine life.

Note: CSS3 has also introduced another way to specify colors called HSLA.

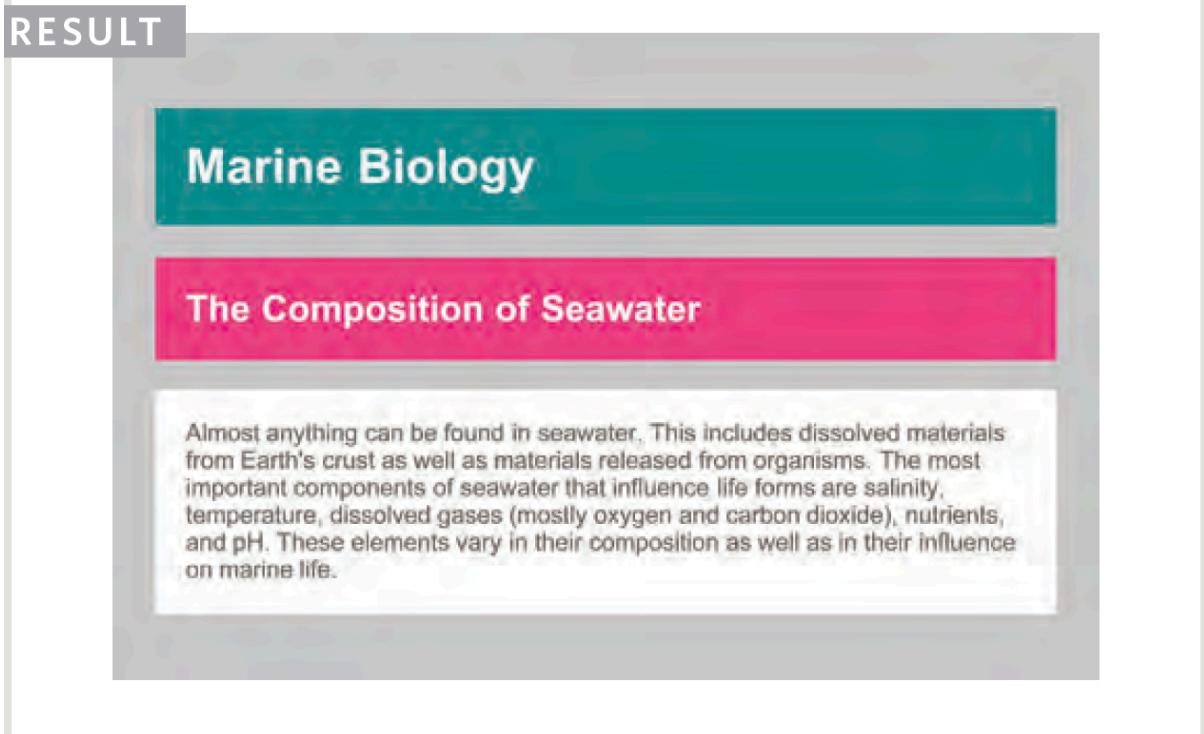
Background Color

CSS treats each HTML element as if it appears in a box, and the background-color property sets the color of the background for that box.

You can specify your choice of background color in the same three ways. If you do not specify a background color, then the background is transparent.

```
1 body { background-color: rgb(200,200,200); }
2 h1 { background-color: DarkCyan; }
3 h2 { background-color: #ee3e80; }
4 p { background-color: white; }
5
```

RESULT



Contrast



- Text is harder to read when there is low contrast between background and foreground colors.
- Text is easier to read when there is higher contrast between background and foreground colors.
- For long spans of text, reducing the contrast a little bit improves readability.
- A lack of contrast is particularly a problem for those with visual impairments and color blindness.

Note: To check contrast there is a handy online tool at:
www.snook.ca/technical/colour_contrast/colour.html

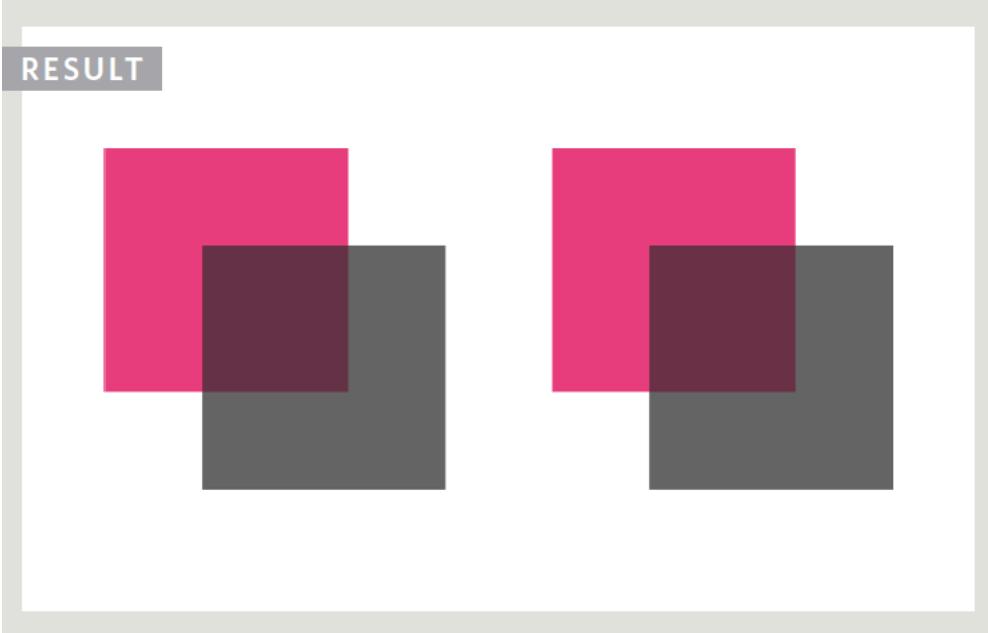
CSS 3: Opacity

CSS3 introduces the opacity property which allows you to specify the opacity of an element and any of its child elements. The value is a number between 0.0 and 1.0 (so a value of 0.5 is 50% opacity and 0.15 is 15% opacity)

The CSS3 rgba property allows you to specify a color, just like you would with an RGB value, but adds a fourth value to indicate opacity. This value is known as an alpha value and is a number between 0.0 and 1.0 (so a value of 0.5 is 50% opacity and 0.15 is 15% opacity). The rgba value will only affect the element on which it is applied (not child elements).

Example

```
1  p.one
2  {
3      background-color: rgb(0,0,0);
4      opacity: 0.5;
5  }
6
7  p.two
8  {
9      background-color: rgba(0,0,0,0.5);
10 }
11 |
```



CSS 3: HSL Colors

The `hsl` color property has been introduced in CSS3 as an alternative way to specify colors. The value of the property starts with the letters `hsl`, followed by individual values inside parentheses for:

hue

This is expressed as an angle (between 0 and 360 degrees).

saturation

This is expressed as a percentage.

lightness

This is expressed as a percentage with 0% being white, 50% being normal, and 100% being black.

The `hsla` color property allows you to specify color properties using hue, saturation, and lightness as above, and adds a fourth value which represents transparency (just like the `rgba` property). The `a` stands for:

alpha

This is expressed as a number between 0 and 1.0. For example, 0.5 represents 50% transparency, and 0.75 represents 75% transparency.

```
1  body
2  {
3      background-color: hsl(0,0%,78%);
4  }
5
6  p {
7      background-color: hsla(0,100%,100%,0.5);
8  }
```

RESULT

The screenshot shows a web page with a teal header containing the text "Marine Biology". Below the header is a pink rectangular area containing the text "The Composition of Seawater". Underneath this is a larger text box with a gray border, containing the following text:
Almost anything can be found in seawater. This includes dissolved materials from Earth's crust as well as materials released from organisms. The most important components of seawater that influence life forms are salinity, temperature, dissolved gases (mostly oxygen and carbon dioxide), nutrients, and pH. These elements vary in their composition as well as in their influence on marine life.

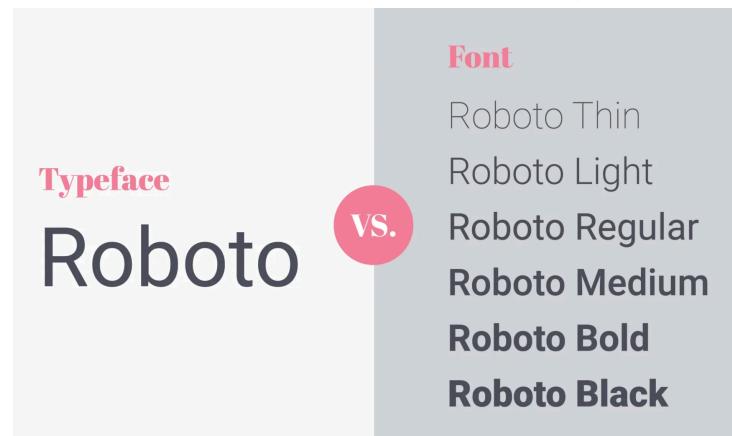
Text

The properties that allow you to control the appearance of text can be split into two groups:

- Those that directly affect the font and its appearance (including the typeface, whether it is regular, bold or italic, and the size of the text)
- Those that would have the same effect on text no matter what font you were using (including the color of text and the spacing between words and letters)

Font vs Typeface

The main difference between these two terms is that a typeface (or type family) is the name of a specific collection of related fonts. In comparison, font refers to a particular weight, width, and style within that typeface. To put it in simple terms, each variation of a typeface is a font.



Typeface Terminology

SERIF	SANS-SERIF	MONOSPACE
Serif fonts have extra details on the ends of the main strokes of the letters. These details are known as serifs.	Sans-serif fonts have straight ends to letters, and therefore have a much cleaner design.	Every letter in a monospace (or fixed-width) font is the same width. (Non-monospace fonts have different widths.)
In print, serif fonts were traditionally used for long passages of text because they were considered easier to read.	Screens have a lower resolution than print. So, if the text is small, sans-serif fonts can be clearer to read.	Monospace fonts are commonly used for code because they align nicely, making the text easier to follow.

Specifying Typefaces

The font-family property allows you to specify the typeface that should be used for any text inside the element(s) to which a CSS rule applies.

The value of this property is the name of the typeface you want to use. The people who are visiting your site need the typeface you have specified installed on their computer in order for it to be displayed.

You can specify a list of fonts separated by commas so that, if the user does not have your first choice of typeface installed, the browser can try to use an alternative font from the list.

If a font name is made up of more than one word, it should be put in double quotes.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Font Family</title>
5      <style type="text/css">    body {
6          font-family: Georgia, Times, serif;
7          h1, h2 {
8              font-family: Arial, Verdana, sans-serif;
9          .credits {
10              font-family: "Courier New", Courier,
11                  monospace;
12          }
13      </style>
14  </head>
15  <body>
16      <h1>
17          Briards
18      </h1>
19      <p class="credits">
20          by Ivy Duckett
21      </p>
22      <p class="intro">
23          The <a class="breed" href="http://en.wikipedia.org/wiki/
24          Briard">briard</a>, or berger de brie, is a large breed of dog
25          traditionally used as a herder and guardian of sheep...
26      </p>
27  </body>
28  </html>
```

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...

Size of Type

The font-size property enables you to specify a size for the font. There are several ways to specify the size of a font. The most common are:

- Pixels are commonly used because they allow web designers very precise control over how much space their text takes up. The number of pixels is followed by the letters px.
- The default size of text in browsers is 16px. So a size of 75% would be the equivalent of 12px, and 200% would be 32px.
- An em is equivalent to the width of a letter m.

```
1  body
2  {
3      font-family: Arial, Verdana, sans-serif;
4      font-size: 12px;
5  }
6
7  h1 { font-size: 200%; }
8  h2 { font-size: 1.3em; }
```

More Font Choice

The font-weight property allows you to create bold text. There are two values that this property commonly takes:

- normal
- bold

If you want to create italic text, you can use the font-style property. There are three values this property can take:

- normal
- italic
- oblique

The text-transform property is used to change the case of text giving it one of the following values:

- uppercase
- lowercase
- capitalize

The text-decoration property allows you to specify the following values:

- none
- underline
- overline
- line-through
- blink

The text-align property allows you to control the alignment of text. The property can take one of four values:

- left

- right
- center
- justify

Responding to Users

There are three pseudo-classes that allow you to change the appearance of elements when a user is interacting with them.

:hover, :active, :focus

:hover

This is applied when a user hovers over an element with a pointing device such as a mouse. This has commonly been used to change the appearance of links and buttons when a user places their cursor over them. It is worth noting that such events do not work on devices that use touch screens (such as the iPad) because the screen is not able to tell when someone is hovering their finger over an element.

:active

This is applied when an element is being activated by a user; for example, when a button is being pressed or a link being clicked. Sometimes this is used to make a button or link feel more like it is being pressed by changing the style or position of the element slightly.

:focus

This is applied when an element has focus. Any element that you can interact with, such as a link you can click on or any form control can have focus.

Example

```
<!DOCTYPE html>
<html>
<head>
    <title>Text</title>
    <style type="text/css">
        body {
            padding: 20px;
        }
        h1, h2, h3, a {
            font-weight: normal;
            color: #0088dd;
            margin: 0px;
        }
        h1 {
            font-family: Georgia, Times, serif;
            font-size: 250%;
            text-shadow: 2px 2px 3px #666666;
            padding-bottom: 10px;
        }
        h2 {
            font-family: "Gill Sans", Arial, sans-serif;
            font-size: 90%;
        }
    </style>
</head>
<body>
    <h1>Hello World!</h1>
    <h2>This is a heading</h2>
    <a href="#">Click me!</a>
</body>
</html>
```

```
        text-transform: uppercase;
        letter-spacing: 0.2em; }
h3 {
    font-size: 150%; }
p {
    font-family: Arial, Verdana, sans-serif;
    line-height: 1.4em;
    color: #665544; }
p.intro:first-line {
    font-weight: bold; }
.credits {
    font-style: italic;
    text-align: right; }
a {
    text-decoration: none; }
a:hover {
    text-decoration: underline; }
</style>
</head>
<body>
<h1>
    Briards
</h1>
<h2>
    A Heart wrapped in fur
</h2>
<p class="intro">
    The <a class="breed"
    href="http://en.wikipedia.org/wikiBriard"> briard</a>, or berger de
    brie, is a large breed of dog traditionally used as a herder and
    guardian of sheep.
</p>
<h3>
    Breed History
</h3>
<p>
    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
    briards.
</p>
<p class="credits">
    by Ivy Duckett
</p>
</body>
</html>
```

Briards

A HEART WRAPPED IN FUR

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 briards.

by Ivy Duckett

Boxes

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

Box Dimension

By default a box is sized just big enough to hold its contents. To set your own dimensions for a box you can use the height and width properties. The most popular ways to specify the size of a box are to use pixels, percentages, or ems.

```
1 <div>
2   <p>
3     The Moog company pioneered the commercial manufacture
4       of modular voltage-controlled analog synthesizer systems
5         in the early 1950s.
6   </p>
7 </div>
8
9
10 div.box
11 {
12   height: 300px;
13   width: 300px;
14   background-color: #bbbbbaa;
15 }
16
17 p
18 {
19   height: 75%;
20   width: 75%;
21   background-color: #0088dd;
22 }
```

Note: Create HTML document and see the results.

- Some page designs expand and shrink to fit the size of the user's screen. In such designs, the min-width property specifies the smallest size a box can be displayed at when the browser window is narrow, and the max-width property indicates the maximum width a box can stretch to when the browser window is wide.
- you may also want to limit the height of it. This is achieved using the min-height and max-height properties.
- The overflow property tells the browser what to do if the content contained within a box is larger than the box itself. It can have one of two values:
 - hidden: This property simply hides any extra content that does not fit in the box.
 - scroll: This property adds a scrollbar to the box so that users can scroll to see the missing content.

Border, Margin & Padding

Every box has three available properties that can be adjusted to control its appearance:



Border Width

The border-width property is used to control the width of a border. The value of this property can either be given in pixels or using one of the following values:

- thin
- medium

- thick

You can control the individual size of borders using four separate properties:

- border-top-width
- border-right-width
- border-bottom-width
- border-left-width

```
<p class="one">Hohner's "Clavinet" is essentially an  
electric clavichord.</p>  
<p class="two">Hohner's "Clavinet" is essentially an  
electric clavichord.</p>  
<p class="three">Hohner's "Clavinet" is essentially  
an electric clavichord.</p>
```

CSS

```
p.one {  
    border-width: 2px;}  
p.two {  
    border-width: thick;}  
p.three {  
    border-width: 1px 4px 12px 4px;}
```

Border Style

You can control the style of a border using the border-style property. This property can take the values as shown in the following example:

```
9  p.one {border-style: solid;}  
10 p.two {border-style: dotted;}  
11 p.three {border-style: dashed;}  
12 p.four {border-style: double;}  
13 p.five {border-style: groove;}  
14 p.six {border-style: ridge;}  
15 p.seven {border-style: inset;}  
16 p.eight {border-style: outset;}
```

RESULT

The screenshot displays a vertical stack of eight rectangular boxes, each containing the text "Wurlitzer Electric Piano". The boxes are styled with different border properties:

- Box 1: Solid border (black)
- Box 2: Dotted border (black dots)
- Box 3: Dashed border (black dashes)
- Box 4: Double border (two black borders)
- Box 5: Groove border (outer border is groove-like)
- Box 6: Ridge border (outer border is ridge-like)
- Box 7: Inset border (inner border is thin black)
- Box 8: Outset border (outer border is thick black)

Padding

The padding property allows you to specify how much space should appear between the content of an element and its border. The value of this property is most often specified in pixels. You can specify different values for each side of a box using:

- padding-top
- padding-right
- padding-bottom
- padding-left

Margin

The margin property controls the gap between boxes. Its value is commonly given in pixels. You can specify values for each side of a box using:

- margin-top
- margin-right
- margin-bottom
- margin-left

If you want to center a box on the page (or center it inside the element that it sits in), you can set the left-margin and right-margin to auto.

Example:

```

1 <body>
2 <p>
3     Analog synthesizers are often said to have a "warmer" sound than their digital counterparts.
4 </p>
5 <p class="example">
6     Analog synthesizers are often said to have a "warmer" sound than their digital counterparts.
7 </p>
8 </body>
9
10 body { text-align: center; }
11 p
12 {
13     width: 300px;
14     padding: 50px;
15     border: 20px solid #0088dd;
16 }
17 p.example
18 {
19     margin: 10px auto 10px auto;
20     text-align: left;
21 }
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
159

```

Analog synthesizers are often said to have a "warmer" sound than their digital counterparts.

Analog synthesizers are often said to have a "warmer" sound than their digital counterparts.

Change Inline/block

The display property allows you to turn an inline element into a block-level element or vice versa, and can also be used to hide an element from the page. The values this property can take are:

- inline: This causes a block-level element to act like an inline element.
- Block: This causes an inline element to act like a block-level element.
- inline-block: This causes a block-level element to flow like an inline element, while retaining other features of a block-level element.
- none: This hides an element from the page.

Layout

A website is often divided into headers, menus, content and a footer:



There are tons of different layout designs to choose from. However, the structure above, is one of the most common, and we will take a closer look at it in this tutorial.

A Fixed Width Layout

To create a fixed width layout, the width of the main boxes on a page will usually be specified in pixels (and sometimes their height, too). The fixed width layout will stay the same width no matter what size the browser window is, whereas the liquid layout will stretch (or shrink) to fill the screen.

Consider the following example:

```
<body>
  <div id="header">
    <h1>Logo</h1>
    <div id="nav">
      <ul>
        <li><a href="">Home</a></li>
        <li><a href="">Products</a></li>
        <li><a href="">Services</a></li>
        <li><a href="">About</a></li>
        <li><a href="">Contact</a></li>
      </ul>
    </div>
  </div>
  <div id="content">
    <div id="feature">
      <p>Feature</p>
    </div>
    <div class="article column1">
      <p>Column One</p>
    </div>
    <div class="article column2">
      <p>Column Two</p>
    </div>
    <div class="article column3">
      <p>Column Three</p>
    </div>
  </div>
  <div id="footer">
    <p>© Copyright 2011</p>
  </div>
</body>
```

CSS

```
body {
  width: 960px;
  margin: 0 auto;}
#content {
  overflow: auto;
  height: 100%;}
#nav, #feature, #footer {
  background-color: #efefef;
  padding: 10px;
  margin: 10px;}
.column1, .column2, .column3 {
  background-color: #efefef;
  width: 300px;
  float: left;
  margin: 10px;}
li {
  display: inline;
  padding: 5px;}
```

RESULT



The rule for the `<body>` element is used to fix the width of the page at 960 pixels, and it is centered by setting the left and right margins to auto.

The main boxes on the page have a margin of 10 pixels to create a gap between them.

The navigation, feature, and footer panels stretch to the width of the containing element (which in this instance is the `<body>` element), so we do not need to specify a width for them.

The three columns are each 300 pixels wide and use the float property, which allows them to sit next to each other.

Lab Tasks:

1. Recreate the following PH scale using HTML and CSS.



2. Apply CSS to the HTML content to get the following expected output:

HTML:

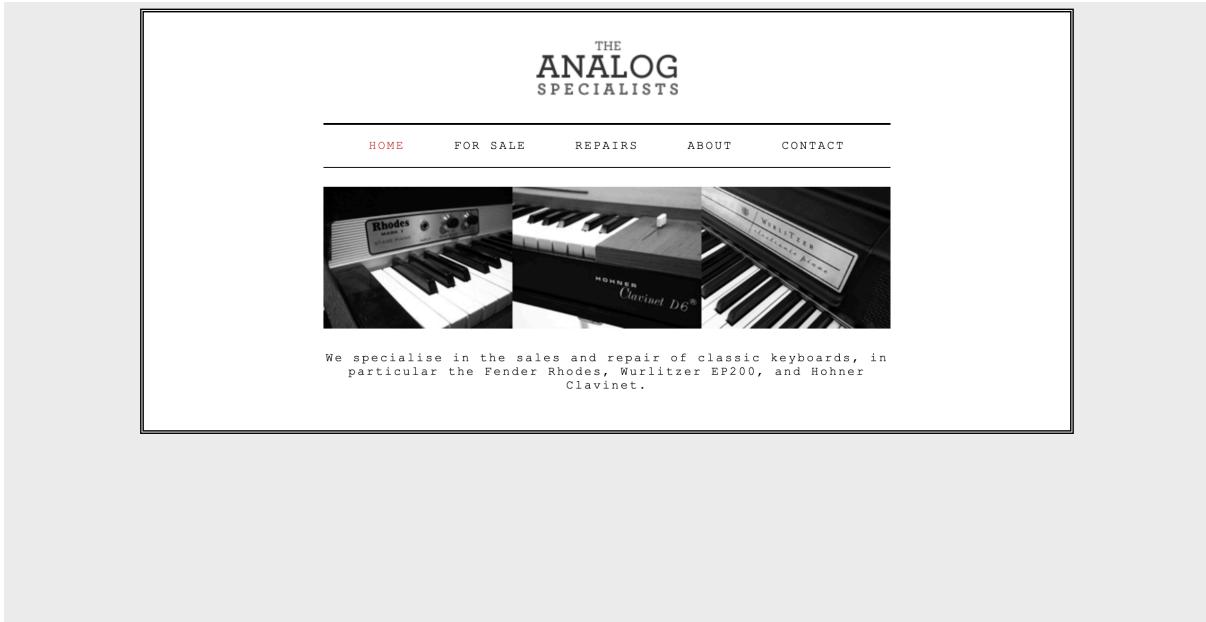
```
<!DOCTYPE html>
<html>
<head>
    <title>Boxes</title>
</head>
<body>
    <div id="page">
        <div id="logo">
            
        </div>
        <ul id="navigation">
            <li><a href="#" class="on">Home</a></li>
            <li><a href="#">For Sale</a></li>
            <li><a href="#">Repairs</a></li>
            <li><a href="#">About</a></li>
            <li><a href="#">Contact</a></li>
        </ul>
        <p>
            
        </p>
        <p>
            We specialise in the sales and repair of classic keyboards, in particular the Fender Rhodes, Wurlitzer EP200, and Hohner Clavinet.
        </p>
    </div>
</body>
</html>
```

We specialise in the sales and repair of classic keyboards, in particular the Fender Rhodes, Wurlitzer EP200, and Hohner Clavinet.

```

        </p>
</div>
</body>
</html>

```



3. Apply CSS to the following form to make it look better:

Registration form

Note: Form is to be completed ...

—Personal Details—

Name: Enter name here
 Address: Enter address here
 Email: Enter email here
 Phone Number: Enter phone number here
 IQ: Enter IQ here

Gender

Male
 Female
 Other

Date of Birth: dd/mm/yyyy

—Check List—

Check All That Apply

check_01
 check_02
 check_03
 check_04

—Education—

Education Level Completed: University

—Essay Section—

In 50 words or more explain why you want to register

Enter Text Here

Please upload contact details for 2 references

Enter Text Here

Upload Police Clearance Certificate, Bank Statement and Medical Certificates here: