



# Css3 Part2

**BY : HESHAM MOHAMED**

# Css Fonts

```
body{
    font-family: cursive;
}
p{
    font-weight: bold;
    font-size: 1.2rem; /* 1rem => 16px */
    font-style: italic;
    font-variant: small-caps;
    font: italic small-caps bold 20px fantasy;
    /*
        font shorthand property
        font-style font-variant

    */
}
```

# Working With Google Fonts

► <https://fonts.google.com/>

```
@import url('https://fonts.googleapis.com/css2?family=Open+Sans:ital,wght@1,600&display=swap');  
body{  
    font-family: 'Open Sans', sans-serif;  
}
```

# CSS Links

```
/*  
The four links states are:  
a:link - a normal, unvisited link  
a:visited - a link the user has visited  
a:hover - a link when the user mouses over it  
a:active - a link the moment it is clicked  
*/  
a{  
    text-decoration: none;  
}  
a:hover{  
    color: #00f;  
}
```

# Css Lists

```
ul{
  /* style-type => disc circle square none */
  list-style-type: circle;
  list-style-image: url('image.jpg');
  list-style-position: inside; /* inside or outside */
  /* Shorthand Property => type position image */
  list-style: circle inside url('image.jpg');
}
ol{
  /* lower-alpha upper-alpha upper-roman lower-roman*/
  list-style-type: lower-alpha;
}
```

# Display

```
h1{
    display: inline;
}
span{
    display: block;
}
li{
    display: inline-block;
}
p{
    /* display none to show and hide elements*/
    display: none;
}
div{
    /*visibility:hidden; also hides an element.However,
    the element will still take up the same space as before.
    The element will be hidden, but still affect the layout*/
    visibility: hidden;
}
```

# Float

- ▶ The float property is used for positioning and formatting content e.g. let an image float left to the text in a container.
- ▶ The float property can have one of the following values:
  - ▶ left - The element floats to the left of its container
  - ▶ right - The element floats to the right of its container
  - ▶ none - The element does not float (will be displayed just where it occurs in the text). This is default
  - ▶ inherit - The element inherits the float value of its parent

# Float

```
.div1{
    float: left;
}
.div2{
    float: right;
}
.div3{
    float: none;
}
.clearfix{
    /* The clear property specifies what elements can float beside
    the cleared element and on which side.
    Values => right , left , none , both
    */
    clear: both;
}
```



# Pseudo-classes

- ▶ A pseudo-class is used to define a special state of an element.
- ▶ For example, it can be used to:
- ▶ Style an element when a user mouses over it
- ▶ Style visited and unvisited links differently
- ▶ Style an element when it gets focus

```
button:hover{  
    color: orange;  
}  
a:active{  
    color: red;  
}
```

# Pseudo-elements

- ▶ What are Pseudo-Elements?
- ▶ A CSS pseudo-element is used to style specified parts of an element.
- ▶ For example, it can be used to:
  - ▶ Style the first letter, or line, of an element
  - ▶ Insert content before, or after, the content of an element

# Pseudo-elements

```
p::first-line{
    color: red;
}
h1::first-letter{
    color: #00f;
}
a::before{
    content: url(image.jpg);
}
a::after{
    content: " |";
}
```

# Opacity

- ▶ The opacity property specifies the opacity/transparency of an element.

```
img{  
    opacity: 1;  
}  
img:hover{  
    opacity: .8;  
}
```

# border-radius Property

- ▶ With the CSS border-radius property, you can give any element "rounded corners".

```
button{  
    background: #2f2f2f;  
    color: #fff;  
    border: none;  
    padding: 10px;  
    border-radius: 20px;  
}
```

# Box Sizing

- ▶ Without the CSS box-sizing Property
- ▶ By default, the width and height of an element is calculated like this:
  - ▶  $\text{width} + \text{padding} + \text{border} = \text{actual width of an element}$
  - ▶  $\text{height} + \text{padding} + \text{border} = \text{actual height of an element}$
- ▶ This means: When you set the width/height of an element, the element often appears bigger than you have set (because the element's border and padding are added to the element's specified width/height).

# Box Sizing

- ▶ The box-sizing property allows us to include the padding and border in an element's total width and height.
- ▶ If you set `box-sizing: border-box;` on an element, padding and border are included in the width and height:
- ▶ Defines how the width and height of an element are calculated: should they include padding and borders, or not

```
*{  
    /* box-sizing: content-box ; or border-box */  
    box-sizing: border-box;  
}
```

# Flexbox Layout

- ▶ To start using the Flexbox model, you need to first define a flex container.
- ▶ The flex container properties are:
  - [flex-direction](#)
  - [flex-wrap](#)
  - [flex-flow](#)
  - [justify-content](#)
  - [align-items](#)
  - [align-content](#)



# Flexbox Layout

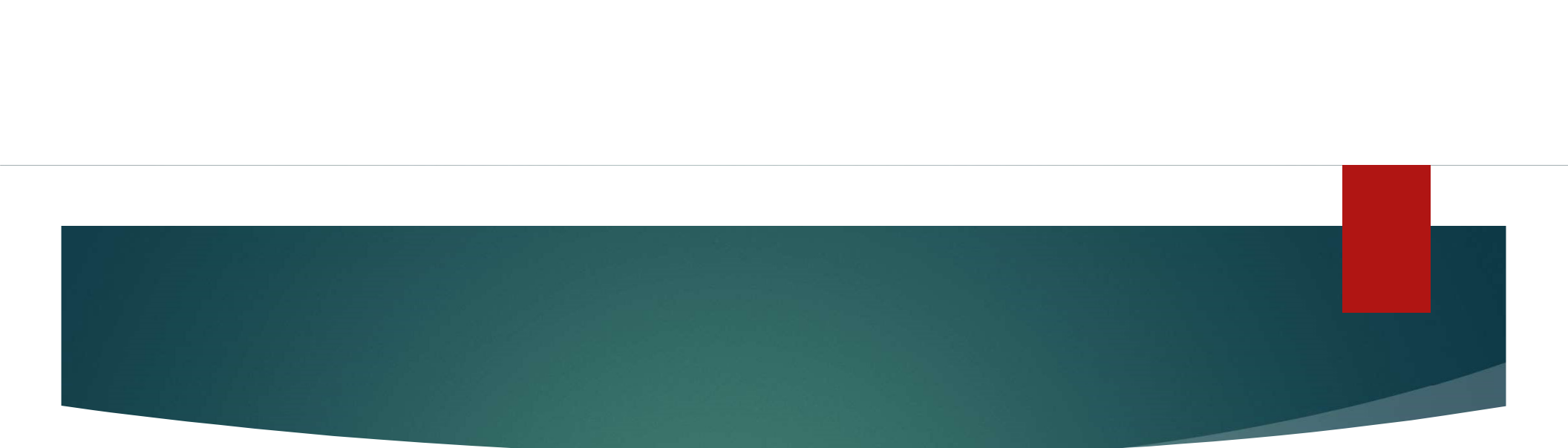
```
/* Parent Div */
.flex-container{
  display: flex;
  flex-direction: row; /* row row-reverse column column-reverse */
  flex-wrap: wrap; /* warp warp-reverse or nowrap*/
  flex-flow: row wrap-reverse;
  justify-content: center;
  /* flex-start flex-end center space-between space-around */
  align-items: center;
  /*flex-start flex-end center stretch baseline*/
  align-content: center;
}
```

# Flexbox Layout

- ▶ Child Elements (Items)
- ▶ • The flex item properties are:
- ▶ • order • flex-grow • flex-shrink • flex-basis • flex • align-self

# Flexbox Layout

```
<div class="flex-container">  
  <div style="order: 3; flex-grow: 1">1</div>  
  <div style="order: 2; flex-grow: 1">2</div>  
  <div style="order: 4; flex-grow: 2">3</div>  
  <div style="order: 1; flex-grow: 6">4</div>  
</div>
```

- 
- ▶ The **align-self** Property • The align-self property specifies the alignment for the selected item inside the flexible container.
  - ▶ • The **align-self** property overrides the default alignment set by the container's align-items property.
  - ▶ The **flex-shrink** property specifies how much a flex item will shrink relative to the rest of the flex items.
  - ▶ The **flex-basis** property specifies the initial length of a flex item.

Thank You 😊