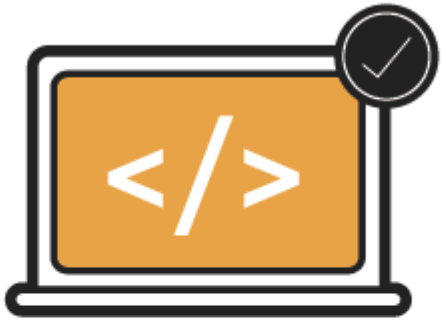
LEARN. DO. EARN

ACADGILD



FRONT END WEB DEVELOPMENT FUNDAMENTALS

Website : http://www.acadgild.com
LinkedIn : https://www.linkedin.com/company/acadgild
Facebook : https://www.facebook.com/acadgild

Session 3 – CSS3

Agenda – CSS3

Sl No	Agenda Title
1	About CSS3
2	Transitions
3	Applying Transitions
4	Animation
5	CSS3 Gradients
6	CSS3 Filters
7	Multi Column Layout
8	Multi Column Layout – Example
9	CSS3 Background

Sl No	Agenda Title
10	Box Model
11	Box Model with box-sizing
12	CSS3 flex Property
13	CSS3 flex Property – Example
14	Selectors
15	pseudo-class
16	pseudo classes
17	text-shadow

About CSS3

- **CSS3** is the latest standard for CSS
- **CSS3** is backward-compatible with earlier versions of CSS
- **CSS3** has been split into "modules“
- Some of the most important CSS3 modules are:
  - Selectors
  - Box Model
  - Backgrounds and Borders
  - Image Values and Replaced Content
  - Text Effects
  - 2D/3D Transformations
  - Animations
  - Multiple Column Layout
  - User Interface

Transitions

Transitions allow the values of CSS properties to **changeover time, essentially providing simple animations.**

How to Use CSS3 Transitions?

- To create a transition effect, you must specify two things:
- the CSS property you want to add an effect to
- the duration of the effect
- If the duration part is not specified, the transition will have no effect, because the default value is 0.

For example, if a link changes color on hover, you can have it gradually fade from one color to the other, instead of a sudden change.

Applying Transitions

Here are the steps to create a simple transition using only CSS3:

- Declare the original state of the element in the default style declaration.
- Declare the final state of your transitioned element (for example, in a hover state).
- Include the transition functions in your default style declaration, using a few different properties:
  - transition-property,
  - transition-duration,
  - transition-timing-function,
  - transition-delay

```
div {
        transition-property: color, left;
        transition-duration: 1s;
        transition-timing: ease-in;
}
```

Animation

- **CSS3 Animations** allow animation of most of the HTML elements without using JavaScript or Flash.
- **CSS3 Animations**, unlike transitions, allow you to control each step of an animation via **keyframes**.
- **CSS 3Animation** allow us to add any number of keyframes in between, to guide our animation in more complex ways.
- **Animation properties are as shown below:**

animation-name
animation-duration
animation-timing-function
animation-iteration-count
animation-direction
animation-delay
animation-fill-mode

CSS3 Gradients

- **CSS3 gradients** let you display smooth transitions between two or more specified colors.
- By using CSS3 gradients you can reduce download time and bandwidth usage.
- In addition, elements with gradients look better when zoomed, because the gradient is generated by the browser.
- **CSS3 defines two types of gradients:**
    1. **Linear Gradients** (goes down/up/left/right/diagonally)
    2. **Radial Gradients** (defined by their center)

CSS3 Filters

Allows creation of all the following effects on images:

- Greyscale
- Blur
- Saturate
- Sepia
- Hue Rotate
- Invert
- Brightness
- Contrast
- Opacity
- Drop-Shadow

```
img {
        filter: type(value);
        -webkit-filter: type(value);
        -moz-filter: type(value);
         -ms-filter: type(value);
        -o-filter: type(value);
    }
```

Multi Column Layout

- The **CSS3 multi-column layout** allows easy definition of multiple columns of text
- **Multi Column property** specifies the number of columns an element should be divided into
- Following table lists the multi-columns properties:

Column- count	Specifies the number of columns an element should be divided into
Column-fill	Specifies how to fill columns
Column-gap	Specifies the gap between the columns
Column-rule	A shorthand property for setting all the column-rule-* properties
Column-rule-color	Specifies the color of the rule between columns
Column-rule-style	Specifies the style of the rule between columns
Column-rule-width	Specifies the width of the rule between columns
Column-span	Specifies how many columns an element should span across
Column-width	Specifies a suggested, optimal width for the columns
Columns	A shorthand property for setting column-width and column-count

Multi Column Layout - Example

```
column-width: 15em;
column-gap: 2em; /* shown in yellow */
column-rule: 4px solid green;
padding: 5px; /* shown in blue */
```

CSS3 Background

- **CSS3 Background properties** are used to define the background effects of an element.
- **Background properties** include:
  - background size – using more than one background for an element.
  - background origin – which effects the position of a background.
- **Multiple Backgrounds**
  - The new ability of CSS3 to use multiple backgrounds is a great time saver, allowing you to achieve effects which previously required more than one div.

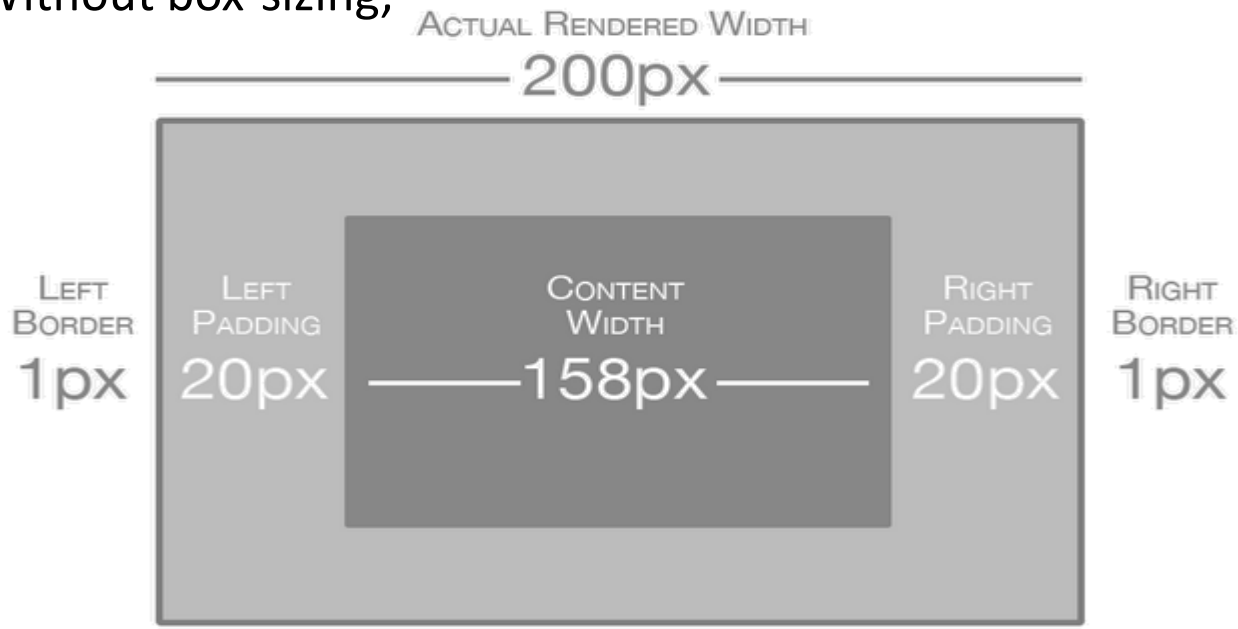
Example:

```
url(example.jpg) top left no-repeat,
url(example2.jpg) bottom left no-repeat,
url(example3.jpg) center center repeat-y;
```

Note: The first image will be the one “closest” to the user.

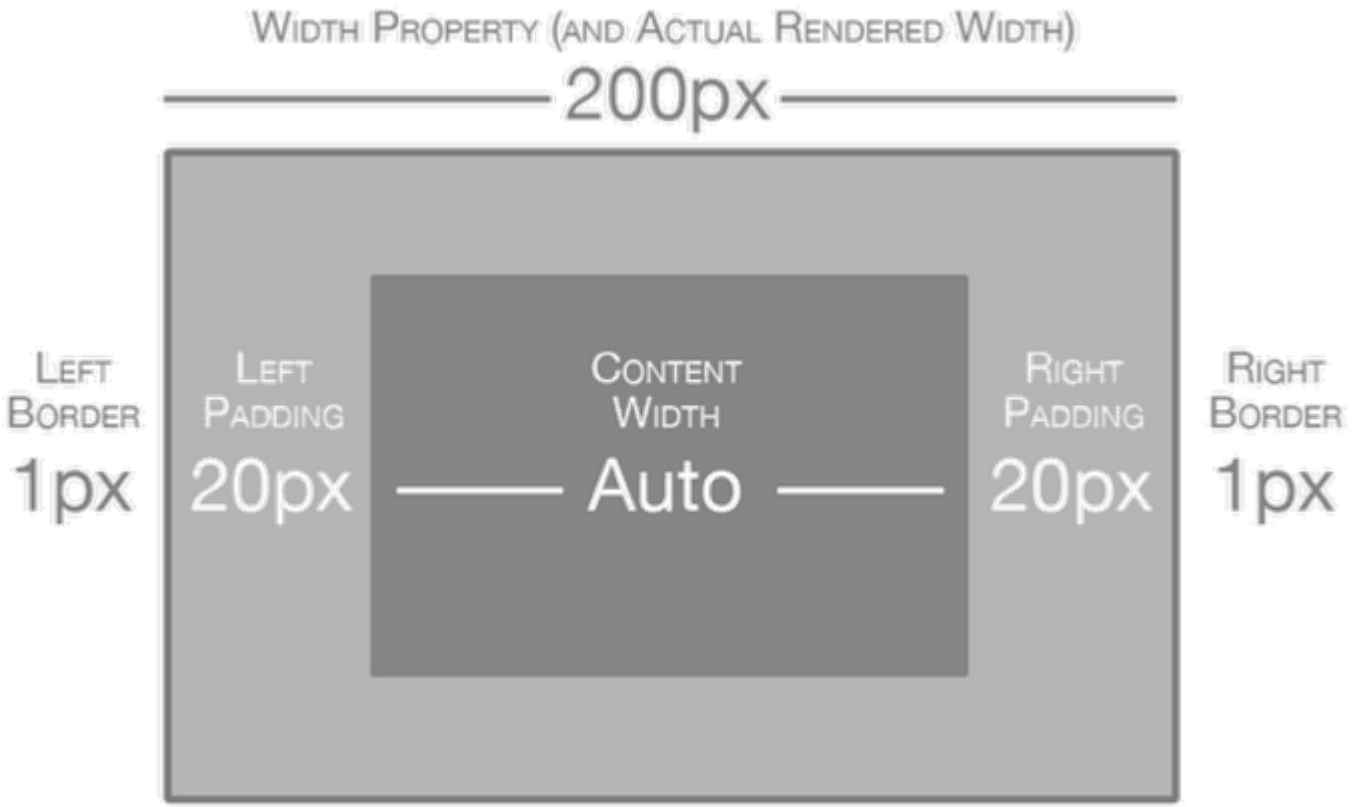
Box Model

- The **Box Model** is a box that wraps around HTML elements.
- The **Box model** consists of margins, borders, padding and the actual content.
- The **Box Model** allows us to add a border around elements and to define space between elements
- The width and height of any element on a webpage is governed by the CSS box model.
- Without box-sizing,



CSS Box Model with box-sizing

- * { box-sizing: border-box; }



CSS3 flex Property

- The **flex property** specifies the length of the item, relative to the rest of the flexible items inside the same container.
- The **flex property** is a shorthand for the flex-grow, flex-shrink, and the flex-basis properties.

Syntax:

```
flex: flex-grow flex-shrink flex-basis|auto|initial|inherit;
```

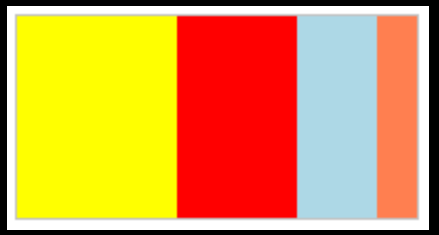
CSS3 flex Property – Example

```
#main {
width: 200px;          height: 100px;          border: 1px          solid #c3c3c3;

display: -webkit-flex; /* Safari */
 display: flex;

 -webkit-flex-direction: row-reverse; /* Safari 6.1+ */
 flex-direction: row-reverse;                          }

/* Safari 6.1+ */
#main div:nth-of-type(1) {-webkit-flex-grow: 5;}
#main div:nth-of-type(2) {-webkit-flex-grow:10;}
#main div:nth-of-type(3) {-webkit-flex-grow: 15;}
#main div:nth-of-type(4) {-webkit-flex-grow: 20;}
/* Standard syntax */
 #main div:nth-of-type(1) {flex-grow: 5;}
#main div:nth-of-type(2) {flex-grow: 10;}
#main div:nth-of-type(3) {flex-grow: 15;}
 #main div:nth-of-type(4) {flex-grow: 20;}
```



Selectors

- **Selectors** are at the heart of CSS.
- **Selectors** are patterns used to select the elements you want to style.
- **Selectors** allow you to select and manipulate HTML elements based on their id, class, type, attribute, etc.
- **Relational selectors**
  - Descendant (E F)
  - Child (E > F)
  - Adjacent Sibling (E + F)
  - General Sibling (E ~ F)
- **Attribute selectors**
  - E[attr$=val]
  - E[attr*=val]

pseudo-class

- 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 mouse moves over it
  - Style visited and unvisited links differently

Syntax:

```
selector:pseudo-class
{
property:value;
}
```

pseudo classes

pseudo classes	Description
:enabled	A user interface element that's enabled.
:disabled	Conversely, a user interface element that's disabled.
:checked	Radio buttons or checkboxes that are selected or ticked.
:valid	Applies to elements that are valid, based on the type or pattern attributes
:invalid	Applies to empty required elements, and elements failing to match the requirements defined by the type or pattern attributes.
:in-range	Applies to elements with range limitations, where the value is within those limitations. This applies, for example, to number and range input types with min and max attributes.

pseudo classes (contd.)

pseudo classes	Description
:out-of-range	The opposite of :in-range: elements whose value is outside the limitations of their range.
:required	Applies to form controls that have the required attribute set.
:optional	Applies to all form controls that do not have the required attribute.
:read-only	Applies to elements whose contents are unable to be altered by the user. This is usually most elements other than form fields.
:read-write	Applies to elements whose contents are user-alterable, such as text input fields.

text-shadow

Syntax:

text-shadow : (x-offset) (y-offset) (blur-radius) (color)

x-offset	to position the shadow along the x-axis
y-offset	to position the shadow along the y-axis
blur-radius	to set the amount of blur
color	to set the color of the shadow

```
#style{
    text-shadow : 0px  -15px 0 #fe2192
}
```

Lets Discuss Assignments

Assignment