CSS3

Banu Prakash

banuprakashc@yahoo.co.in

Introduction to CSS3

- **CSS (Cascading Style Sheets)** is a language which describes how a document written in a markup language looks and how it is formatted.

- Giving layout, colors and fonts to a document and its elements can change how your website looks and how people think about it.

- CSS has evolved over time from CSS1 to CSS2 (which is the current standard) to **CSS3** (which is still in development).

- Modern browser support CSS1 and CSS2 but for CSS3 the support is still limited.

Layout Engine Support

- There are a number of Layout Engines available, each rendering the marked up content and formatting information in its own way.
  - Leading in support is the Webkit Engine, used by Google Chrome
  - Safari and the Gecko Webkit Engine for Mozilla Firefox
- Some of the CSS3 properties which are still in experimental phase are lead by a prefix:
  - Trident (Microsoft Internet Explorer) : -ms-
  - Gecko (Google Chrome & Safari) : -webkit-
  - WebKit (Mozilla Firefox): -moz-
- Over time these prefixes will diminish when they are no longer in experimental phase.

CSS3 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

The CSS Box Model

- All HTML elements can be considered as boxes.

- In CSS, the term "box model" is used when talking about design and layout.

- The CSS box model is essentially a box that wraps around HTML elements, and it consists of: margins, borders, padding, and the actual content.
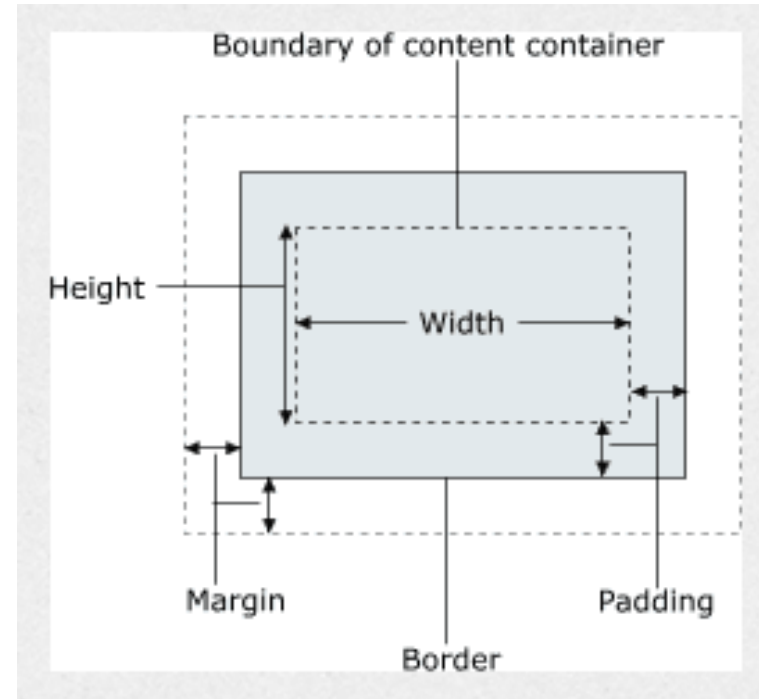
CSS3 Box Model

- CSS3 Flexible Box Module
  - The Flexbox Layout (Flexible Box) module aims at providing a more efficient way to lay out, align and distribute space among items in a container, even when their size is unknown and/or dynamic (thus the word "flex").
  - The main idea behind the flex layout is to give the container the ability to alter its items' width/height (and order) to best fill the available space (mostly to accommodate to all kind of display devices and screen sizes).
  - A flex container expands items to fill available free space, or shrinks them to prevent overflow.
  - Most importantly, the flexbox layout is direction-agnostic as opposed to the regular layouts (block which is vertically-based and inline which is horizontally-based).

CSS3 Flexible Box Module

- main axis - The main axis of a flex container is the primary axis along which flex items are laid out.; it depends on the flex-direction property

- main-start | main-end - The flex items are placed within the container starting from main-start and going to main-end.

- main size - A flex item's width or height, whichever is in the main dimension, is the item's main size.

- cross axis - The axis perpendicular to the main axis is called the cross axis.
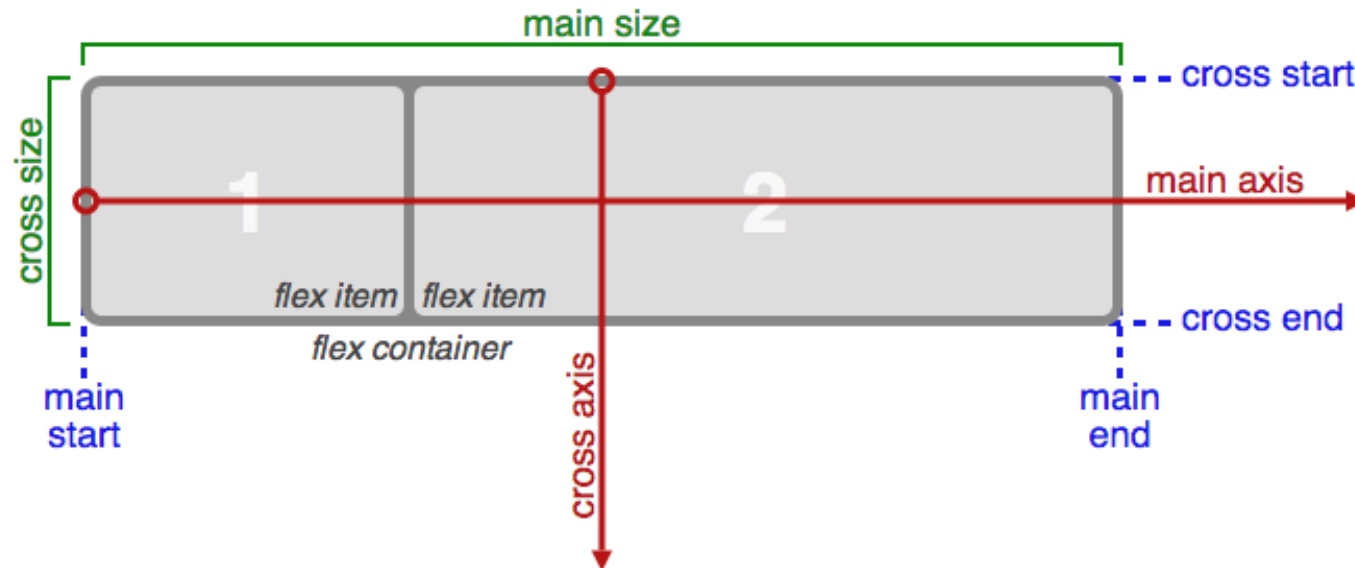
CSS3 Flexible Box Module
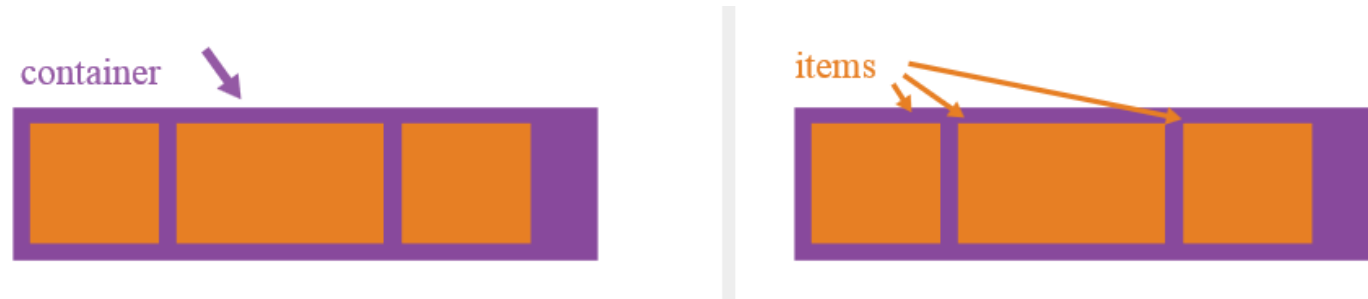
- Properties for the Parent (flex container)
- **display**
  - This defines a flex container; inline or block depending on the given value. It enables a flex context for all its direct children.

    .container {

    display: flex; /* or inline-flex */
    }

CSS3 Flexible Box Module

- **flex-direction**
  - This establishes the main-axis, thus defining the direction flex items are placed in the flex container.
  - Think of flex items as primarily laying out either in horizontal rows or vertical columns.

    .container {

      flex-direction: row | row-reverse | column | column-reverse;

    }

    - row (default): left to right
    - row-reverse: right to left
    - column: same as row but top to bottom
    - column-reverse: same as row-reverse but bottom to top

CSS3 Flexible Box Module

- **flex-wrap**
  - By default, flex items will all try to fit onto one line.
  - You can change that and allow the items to wrap as needed with this property.
  - Direction also plays a role here, determining the direction new lines are stacked in.

    .container{

    flex-wrap: nowrap | wrap | wrap-reverse;

    }
  - nowrap (default): single-line / left to right
  - wrap: multi-line / left to right
  - wrap-reverse: multi-line / right to left
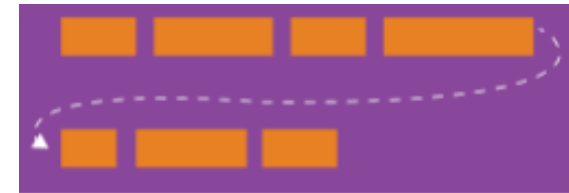
CSS3 Flexible Box Module

- **justify-content**

  .container {

   justify-content: flex-start |

  flex-end |

  center |

  space-between | space-around;

  }
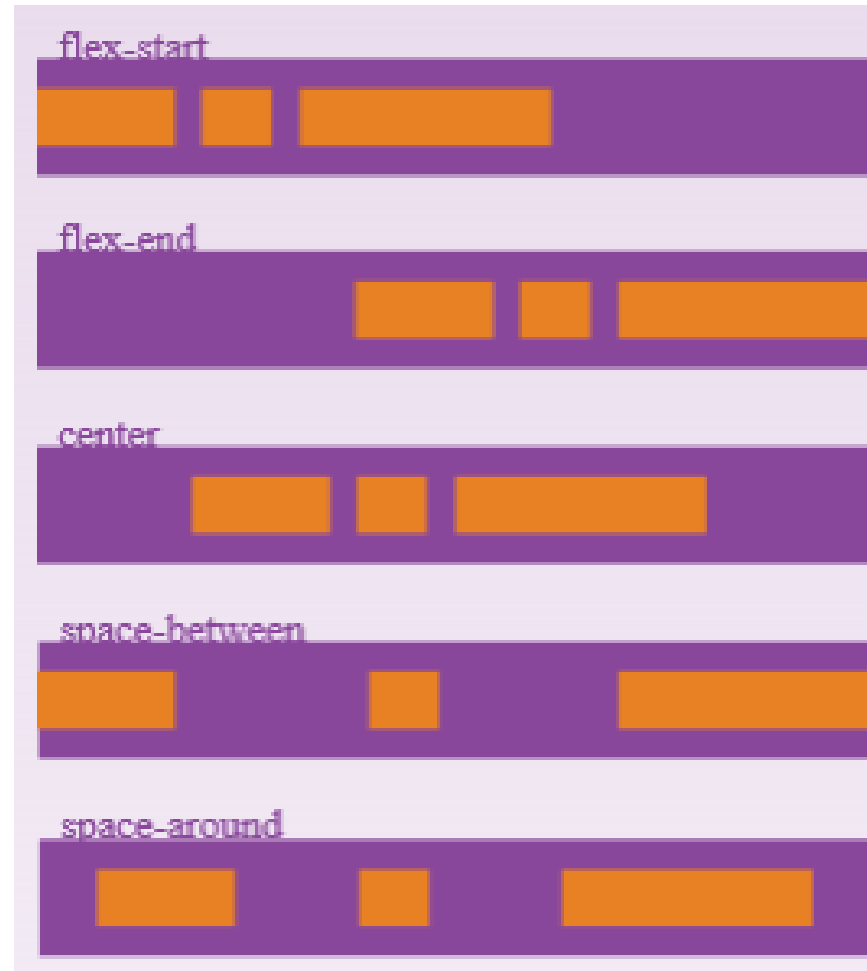
CSS3 Flexible Box Module

- **Properties for the Children**

- **Order**
  - By default, flex items are laid out in the source order. However, the order property controls the order in which they appear in the flex container.

  .item {

    order: <integer>;

  }
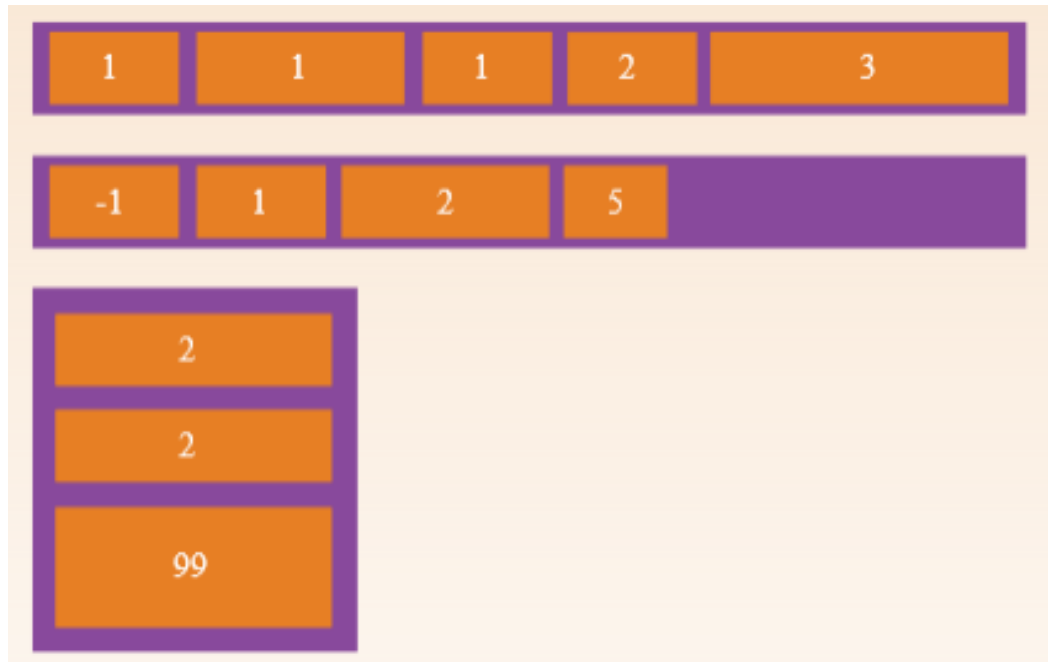
CSS3 Flexible Box Module

- **Properties for the Children**
  - **flex-grow**
  - This defines the ability for a flex item to grow if necessary. It accepts a unit-less value that serves as a proportion. It dictates what amount of the available space inside the flex container the item should take up.
  - If all items have flex-grow set to 1, every child will set to an equal size inside the container. If you were to give one of the children a value of 2, that child would take up twice as much space as the others.

    .item {
      flex-grow: <number>; /* default 0 */
    }
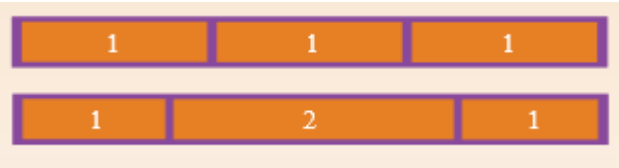
CSS3 Flexible Box Module example -1
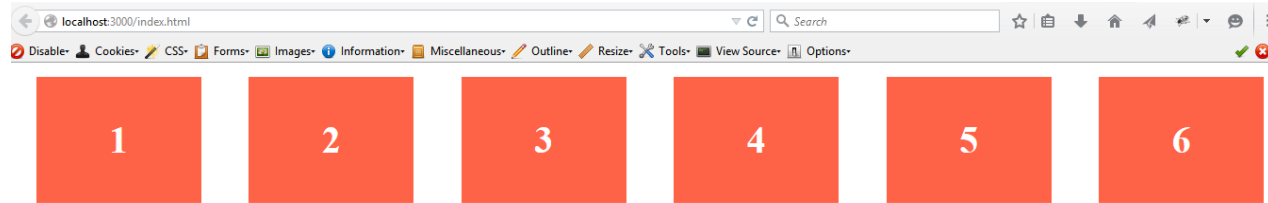

```css
.flex-container {
  padding: 0;
  margin: 0;
  list-style: none;
  display: -webkit-box;
  display: -moz-box;
  display: -ms-flexbox;
  display: -webkit-flex;
  display: flex;
  -webkit-flex-flow: row wrap;
  justify-content: space-around;
}
```

```css
.flex-item {
  background: tomato;
  padding: 5px;
  width: 200px;
  height: 150px;
  margin-top: 10px;
  line-height: 150px;
  color: white;
  font-weight: bold;
  font-size: 3em;
  text-align: center;
}
```

```html
<ul class="flex-container">
    <li class="flex-item">1</li>
    <li class="flex-item">2</li>
    <li class="flex-item">3</li>
    <li class="flex-item">4</li>
    <li class="flex-item">5</li>
    <li class="flex-item">6</li>
</ul>
```

CSS3 @media Rule

- The @media rule is used to define different style rules for different media types/devices.
- In CSS2 this was called media types, while in CSS3 it is called media queries.
- Media queries look at the capability of the device, and can be used to check many things, such as:
- width and height of the browser window
- width and height of the device
- orientation (is the tablet/phone in landscape or portrait mode?)
- resolution
- and much more

CSS3 @media Rule

- **Media Types**
  - all   Used for all media type devices
  - print   Used for printers
  - screen   Used for computer screens, tablets, smart-phones etc.
  - speech   Used for screenreaders that "reads" the page out loud

```
@media screen and (max-width: 300px) {
    body {
        background-color: lightblue;
    }
}
```

CSS3 Flexible Box Module example -2

- Navigation Menu

  - When on medium sized screens, we center it by evenly distributing empty space around items
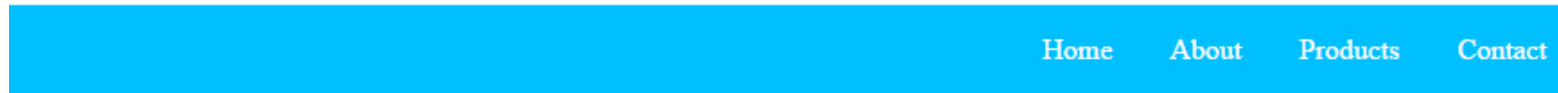

```
.navigation {
  list-style: none;
  margin: 0;
  background: deepskyblue;
  display: -webkit-box;
  display: -moz-box;
  display: -ms-flexbox;
  display: -webkit-flex;
  display: flex;
  -webkit-flex-flow: row wrap;
  justify-content: flex-end;
}
```

```
@media all and (max-width: 800px) {
  .navigation {
    justify-content: space-around;
  }
}
```

CSS3 Flexible Box Module example -2

- On small screens, we are no longer using row direction but column

```
@media all and (max-width: 600px) {
  .navigation {
    -webkit-flex-flow: column wrap;
    flex-flow: column wrap;
    padding: 0;
  }

  .navigation a {
    text-align: center;
    padding: 10px;
    border-top: 1px solid rgba(255, 255, 255, 0.3);
    border-bottom: 1px solid rgba(0, 0, 0, 0.1);
  }

  .navigation li:last-of-type a {
    border-bottom: none;
  }
}
```

Home

About

Products

Contact

CSS3 Flexible Box Module example - 3

- Mobile-first 3-columns layout with full-width header and footer.

- Page Content

```html
<div class="wrapper">
    <header class="header">Header</header>
    <article class="main">
        <p>Mindtree delivers technology services and accelerates growth
            for Global 1000 companies by solving complex business challenges
            with breakthrough technical innovations. Mindtree specializes in
            e-commerce, mobility, cloud enablement, digital transformation,
            business intelligence, data analytics, testing, infrastructure, EAI
            and ERP solutions</p>
    </article>
    <aside class="aside aside-1">Aside 1</aside>
    <aside class="aside aside-2">Aside 2</aside>
    <footer class="footer">Footer</footer>
</div>
```

CSS3 Flexible Box Module example - 3

- Page output on wide screen
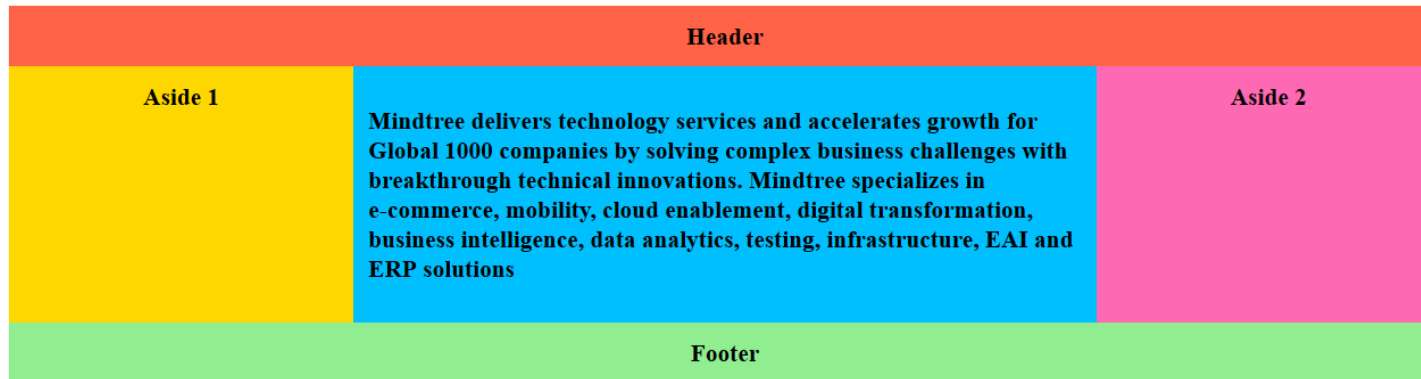
- Page output in Mobile screen
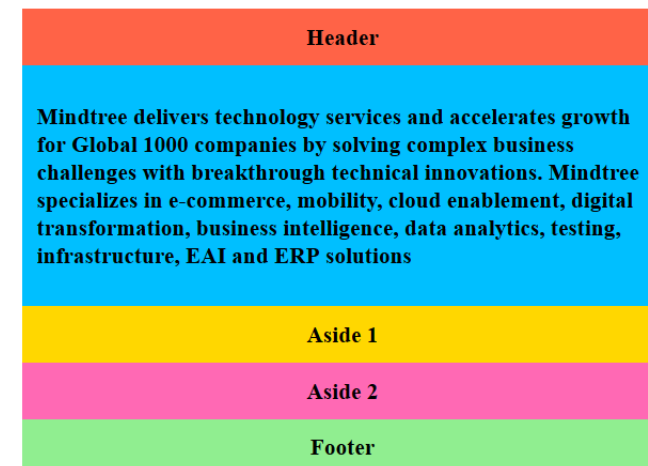
CSS3 Flexible Box Module example - 3

- Medium screens
  - We tell both sidebars to share a row
    .aside { flex: 1 auto; }

- Large screens

- the main element to take thrice as much width as the other two sidebars

```css
.wrapper {
    display: -webkit-box;
    display: -moz-box;
    display: -ms-flexbox;
    display: -webkit-flex;
    display: flex;
    -webkit-flex-flow: row wrap;
    flex-flow: row wrap;
    font-weight: bold;
    text-align: center;
}
```

```css
@media all and (min-width: 600px) {
    .aside {
        flex: 1 auto;
    }
}

@media all and (min-width: 800px) {
    .main {
        flex: 3 0px;
    }
    .aside-1 {
        order: 1;
    }
    .main {
        order: 2;
    }
    .aside-2 {
        order: 3;
    }
    .footer {
        order: 4;
    }
}
```

CSS3 Multiple Columns

- CSS3 Create Multiple Columns
- Multiple column properties:
  - column-count
  - column-gap
  - column-rule

```
{
        -webkit-column-count: 3;
        column-count: 3; /*IE*/
        -webkit-column-gap: 40px;
        column-gap: 40px; /*IE*/
        -webkit-column-rule: 4px outset #ff00ff;
        column-rule: 4px outset #ff00ff; /*IE*/
}
```

CSS3 Selectors

1. element1~element2
   Selects an element that is a general next sibling of another element
   - *Example:*

```css
section ~ footer {
    color: #CB000F;
}
```

   - *All footer elements declared after section will have color of red. the <footer> element doesn't have to be the direct preceding element*

CSS3 Selectors

[attribute^=value]
*Represents an element with the attribute whose value begins with the prefix **value**.*

[attribute$=value]
*Represents an element with the attribute whose value ends with the prefix **value**.*

[attribute*=value]
*Represents an element with the attribute whose value contains the prefix **value**.*

```css
p[clazzName^=red] {
    color: red;
}

p[testAttr$=MakeBold] {
    font-weight: bold;
}

p[clazz*=Underline] {
    text-decoration: underline;
}
```

```html
<p clazzName="redClass">
    This paragraph is red
</p>
<p testAttr="ParaMakeBold">
    This paragraph is bold
</p>

<p clazz="ParaUnderline">
    This paragraph is underlined
</p>
```

CSS3 Selectors

- Pseudo elements/classes
  - :first-of-type
  - :last-of-type
  - :only-of-type
    - *Select elements that are first, last, or only child of its type of its parent*

```
table tr:first-of-type {
    background-color: blue;
    color: white
}

table tr:last-of-type {
    background-color: black;
    color: yellow
}
```

Name	Occupation
Banu Prakash	Corporate Trainer
Kavitha	HR
Rahul	Student
Bharath	Network Eng
Swetha	House Wife
Karthik	Maint. Eng
Sample List of People	

CSS3 Selectors

- Pseudo elements/classes
  - :first-child
  - :only-child
  - :last-child
    - *Selects an element that is the first, only, or last child of its parent*

```css
table tr:first-child {
    background-color: green;
    color: black
}

table tr:last-child {
    background-color: black;
    color: yellow
}
```

CSS3 Selectors

- nth-child(n)
- :nth-of-type(n)
  - *Allows you to select elements with a formula.*
  - *The syntax is :nth-child(an+b), where you replace a and b are numbers.*
    - *For instance, :nth-child(3n+1) selects the 1st, 4th, 7th etc. child.*
  - *:nth-of-type() works the same, except that it only considers element of the given type*

```css
table tr:nth-child(2n+1) {
    background-color: green;
    color: white
}
```

```css
table tr:nth-of-type(2n+1) {
    background-color: red;
    color: white
}
```

- :nth-last-child(n)
- :nth-last-of-type(n)
  - *Work same as nth-child, except that they start counting at the last element instead of the first.*

CSS3 Gradients

- **CSS3 Linear Gradients**
  - background: linear-gradient(*direction*, *color-stop1*, *color-stop2, …*);
  - A linear gradient from top to bottom:
    - background: -webkit-linear-gradient(red, blue);
  - A linear gradient from left to right:
    - background: -webkit-linear-gradient(left, red , blue);
  - A linear gradient that starts at top left (and goes to bottom right):
    - background: -webkit-linear-gradient(left top, red , blue);

CSS3 Gradients

- A linear gradient from left to right, with transparency
  - background: -webkit-linear-gradient(left,rgba(255,0,0,0),rgba(255,0,0,1));
- repeating linear gradient
  - background: -webkit-repeating-linear-gradient(red, yellow 10%, green 20%);
- A radial gradient with evenly spaced color stops
  - background: -webkit-radial-gradient(red, green, blue);
- A radial gradient with differently spaced color stops
  - background: -webkit-radial-gradient(red 5%, green 15%, blue 60%);

Responsive design & font sizes

- Change your font-size depending on screen size

- Base font size is set to 16px.

- Remember: 1em = 16px.

```
@media all and (min-width: 960px) {
        body{
                font-size: 18px;
        }
}

@media all and (max-width: 959px) and (min-width: 600px) {
        body{
                font-size: 16px;
        }
}

@media all and (max-width: 599px) and (min-width: 320px) {
        body{
                font-size: 12px;
        }

}
```

```
body{
        font-size: 16px;
        font-family: sans-serif;
}

h1{
        font-size: 4.236em;
}

h2{
        font-size: 2.618em;
}

h3{
        font-size: 1em;
        line-height: 1.618em;
        text-transform: uppercase;
}

p{
        font-size: 1em;
        line-height: 1.618em;
}
```

Responsive design & font sizes

- <meta name="viewport" content="width=device-width" />

- Not adding this meta tag will cause the site to still load zoomed out rather than zoomed in.

- Remember that web design is 95% typography! It is important that you get that right.

Web fonts

- Allows a page to use fonts not installed on a users computer
- @font-face rule is not part of CSS3.
  - @font-face first spec'd in CSS2
  - It was a part of CSS2, but it was basically *Dead on arrival* (*DOA*) thanks to licensing issues around fonts.
  - Understandably, font designers did not want their work available to be freely downloaded.

- Now a push by the web design community for open-source fonts, and the arrival of services like typekit and Font Squirrel have changed the picture a lot.

Web fonts

- When to use web fonts?
  - Great for headlines, pull quotes, etc., especially when limiting font styles to limit file size
  - Be careful with body copy. Not all fonts are readable at body copy size
  - Keep in mind that different browsers will render differently, so avoid line-breaks and accommodate re-flow of text
  - They draw attention and can help guide the reader through the page.
  - They add spice, basically.
- Not every font works for extended reading, and some of them render poorly, especially at small sizes, in certain browsers (Internet Explorer).

Web Fonts

- Different Font Formats:
- **TrueType Fonts (TTF) :** TrueType is a font standard developed by Apple and Microsoft. TrueType is the most common font format for both the Mac OS and Microsoft Windows operating systems.
- **OpenType Fonts (OTF) :** OpenType is a format for scalable computer fonts. It was built on TrueType, and is a registered trademark of Microsoft. OpenType fonts are used commonly today on the major computer platforms.
- **The Web Open Font Format (WOFF) :** WOFF is a font format for use in web pages. WOFF is essentially OpenType or TrueType with compression and additional metadata. The goal is to support font distribution from a server to a client over a network with bandwidth constraints.
- **Embedded OpenType Fonts (EOT) :** EOT fonts are a compact form of OpenType fonts designed by Microsoft for use as embedded fonts on web pages

Web Fonts

- Browser support

Font format	![IE]	![Chrome]	![Firefox]	![Safari]	![Opera]
TTF/OTF	9.0*	4.0	3.5	3.1	10.0
WOFF	9.0	5.0	3.6	5.1	11.1
WOFF2	Not supported	36.0	35.0*	Not supported	26.0
SVG	Not supported	4.0	Not supported	3.2	9.0
EOT	6.0	Not supported	Not supported	Not supported	Not supported

Web Fonts

- How to use @font-face
  - Hosted services require a script tag and configuration via their website
    - Local code (generated by Font Squirrel):
    - Visit
      - **http://www.fontsquirrel.com/tools/webfont-generator**

```
@font-face  {
          font-family: myFirstFont;
          src: url(sansation_light.woff);
}

div {     font-family:myFirstFont;  }
```

CSS3 transform

- **CSS3 transform Property**
  - The transform property applies a 2D or 3D transformation to an element. This property allows you to rotate, scale, move, skew, etc., elements
  - **Syntax**
    - transform: property_value;

CSS3 transform- **CSS3 transform Property**

Value	Description
translate(x,y) / translate3d(x,y,z)	Defines a 2D/3D translation
translateX(x)/ translateY(y)/ translateZ(z)	Defines a translation, using value for the X-axis. Y-axis and/or Z-axis
scale(x,y)	Defines a 2D scale transformation
scale3d(x,y,z)	Defines a 3D scale transformation
scaleX(x) /scaleY(y) / scaleZ(z)	Defines a scale transformation by giving a value for the axis
rotate(angle)	Defines a 2D rotation, the angle is specified in the parameter
rotate3d(x,y,z,angle)	Defines a 3D rotation
rotateX(angle) / rotateY(angle) / rotateZ(angle)	Defines a 3D rotation
skew(x-angle,y-angle)	Defines a 2D skew transformation along the X- and the Y-axis

CSS3 transform- **CSS3 transform Property**

- Usage
  - -webkit-transform: rotate(30deg);
  - -webkit-transform: translate(50px, 100px);
  - -webkit-transform: scale(2, 4);
  - -webkit-transform: skew(30deg, 20deg);

CSS3 transitions

- The transition property is a shorthand property for the four transition properties:
  - transition-property, transition-duration, transition-timing-function, and transition-delay.

Value	Description
transition-property	Specifies the name of the CSS property the transition effect is for
transition-duration	Specifies how many seconds or milliseconds the transition effect takes to complete
transition-timing-function	Specifies the speed curve of the transition effect
transition-delay	Defines when the transition effect will start

CSS3 transitions

- Example
  - Below example shows a 100px * 100px red <div> element. The <div> element has also specified a transition effect for the width property, with a duration of 2 seconds because of below specified transition

    div {

         width:100px; height:100px;

         background:red;

         -webkit-transition:width 2s;

    }
  - When a user moves mouse over the <div> element, div width changes to 300px in 2s

    div: hover

    {

    width:300px;

    }

CSS3 animations

- **CSS animations** make it possible to animate transitions from one CSS style configuration to another.
- Animations consist of two components, a style describing the CSS animation and a set of keyframes that indicate the start and end states of the animation's style, as well as possible intermediate waypoints along the way.
- There are three key advantages to CSS animations over traditional script-driven animation techniques:
  - They're easy to use for simple animations; you can create them without even having to know JavaScript.
  - The animations run well, even under moderate system load. Simple animations can often perform poorly in JavaScript (unless they're well made).

CSS3 animations

- The @keyframes Rule
  - When you specify CSS styles inside the @keyframes rule, the animation will gradually change **from** the current style **to** the new style.

```css
/* The animation code */
/* Chrome, Safari, Opera */
@-webkit-keyframes example {
    from {background-color: red;}
    to {background-color: yellow;}
}


/* Standard syntax */
@keyframes example {
    from {background-color: red;}
    to {background-color: yellow;}
}
```

```css
/* apply to div */
div {
    width: 100px;
    height: 100px;
    background-color: red;
    /* Chrome, Safari, Opera */
    -webkit-animation-name: example;
    -webkit-animation-duration: 4s;
    animation-name: example;
    animation-duration: 4s;
}
```

CSS3 animations

- Change the background color when the animation is 25%, and 50%, and again when the animation is 100% complete

```
@-webkit-keyframes myfirst
{
0%    {background: red;}
25%   {background: yellow;}
50%   {background: blue;}
100%  {background: green;}
}
```

Using animation events

- You can get additional control over animations as well as useful information about them by making use of animation events.

- These events, represented by the **AnimationEvent** object, can be used to detect when animations start, finish, and begin a new iteration.

- Each event includes the time at which it occurred as well as the name of the animation that triggered the event.

```javascript
// set the var here
var watchme = document.querySelector("#watchme");

// listen for animation start
watchme.addEventListener("animationstart",function(e){
    console.log("log at beginning of watchme animation");
},false);

// listen for animation iteration
watchme.addEventListener("animationiteration",function(e){
    console.log("log at beginning of each subsequent iteration");
},false);

// listen for animation end
watchme.addEventListener("animationend",function(e){
    console.log("log at end of watchme animation");
},false);
```

CSS3 User Interface

- CSS3 has new user interface features such as resizing elements, outlines, and box sizing
  - **CSS3 Resizing**
    - The resize property specifies whether or not an element should be resizable by the user.

      div {

        resize: vertical;

        overflow: auto;

      }
  - **CSS3 Outline Offset**
    - Outlines differ from borders in two ways:
    - An outline is a line drawn around elements, outside the border edge
    - A outline do not take up space

    div {
    ** border: 1px solid black;**
    ** outline: 1px solid red;**
    ** outline-offset: 15px;**
    }

CSS3 User Interface

- **Without the CSS3 box-sizing Property**
  - By default, the width and height of an element is calculated like this:
  - width + padding + border = actual width of an element
    height + padding + border = actual height of an element

- **CSS3 box-sizing**
  - The CSS3 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

    ```
    .div2 {
        width: 300px;
        height: 100px;
        padding: 50px;
        border: 1px solid red;
        box-sizing: border-box;
    }
    ```

Bootstrap

banuprakashc@yahoo.co.in

Objectives

- Understand Grid System,
- Understand container, rows
- Understand fixed and fluid layouts
- Understand typography, buttons, glyphicons
- Understand bootstrap tables, list, forms, images, media objects,
- Understand navigation components, navbar, panels
- Understand Accordion, breadcrumbs, Carousel

Mobile First

- Since Bootstrap 3, Bootstrap has become mobile first.
  - It means mobile first styles can be found throughout the entire library instead of in separate files.
  - You need to add the viewport meta tag to the<head> element, to ensure proper rendering and touch zooming on mobile devices
  - <meta name="viewport" content="width=device-width, initial-scale=1.0">
    - width property controls the width of the device. Setting it to device-width will make sure that it is rendered across various devices (mobiles,desktops,tablets...) properly.
    - initial-scale=1.0 ensures that when loaded, your web page will be rendered at a 1:1 scale, and no zooming will be applied out of the box
    - Add maximum-scale=1.0 with user-scalable=no to the content attribute to disable zooming capabilities

Containers

- Use class .container to wrap a page's content and easily center the content

```
<div class="container">
  ...
</div>
```

```
.container {
    padding-right: 15px;
    padding-left: 15px;
    margin-right: auto;
    margin-left: auto;
}
```

Grid System

- Twitter Bootstrap grid system provides the fastest and easy way to create layouts of web pages

- Twitter Bootstrap 3 introduces the responsive mobile first fluid grid system that appropriately scales up to 12 columns as the device or viewport size increases.

- Twitter Bootstrap 3 includes predefined grid classes for quickly making grid layouts for different types of devices like cell phones, tablets, desktops, etc.

Grid System

- Bootstrap grid system is 12 columns

Grid Options

| Features | Extra small devices | Small devices | Medium devices | Large devices |
Bootstrap 3 Grid System	Phones (<768px)	Tablets (≥768px)	Desktops (≥992px)	Desktops (≥1200px)
Max container width	None (auto)	750px	970px	1170px
Class prefix	.col-xs-	.col-sm-	.col-md-	.col-lg-
Max column width	Auto	~62px	~81px	~97px
Gutter width	15px on each side of a column (i.e. 30px)			

- Applying any .col-sm- class to an element will not only affect its styling on small devices like tablets, but also on medium and large devices having screen size greater than or equal to 768px (i.e. ≥768px) if .col-md- and .col-lg- class is not present.
- Similarly the .col-md- class will not only affect the styling of elements on medium devices, but also on large devices if a .col-lg- class is not present.

Grid options

- In a laptop or desktop having screen or viewport width greater than or equal to 992px and less than 1200px you will find it has 4 rows where each row has 3 equal columns resulting in 3x4 grid layout

```html
<div class="container">
    <div class="row">
        <div class="col-md-4"><p>Box 1</p></div>
        <div class="col-md-4"><p>Box 2</p></div>
        <div class="col-md-4"><p>Box 3</p></div>
        <div class="col-md-4"><p>Box 4</p></div>
        <div class="col-md-4"><p>Box 5</p></div>
        <div class="col-md-4"><p>Box 6</p></div>
        <div class="col-md-4"><p>Box 7</p></div>
        <div class="col-md-4"><p>Box 8</p></div>
        <div class="col-md-4"><p>Box 9</p></div>
        <div class="col-md-4"><p>Box 10</p></div>
        <div class="col-md-4"><p>Box 11</p></div>
        <div class="col-md-4"><p>Box 12</p></div>
    </div>
</div>
```

Box 1	Box 2	Box 3
Box 4	Box 5	Box 6
Box 7	Box 8	Box 9
Box 10	Box 11	Box 12

- Actual content like text, images, videos, etc. should be placed within columns, and only columns may be the immediate children of rows.

Grid options

- Customize our layout for all devices.

```html
<div class="container">
    <div class="row">
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 1</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 2</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 3</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 4</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 5</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 6</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 7</p></div>
        <div class="col-sm-6 col-md-4 col-lg-3"><p>Box 8</p></div>
    </div>
</div>
```

Large device			
Box 1	Box 2	Box 3	Box 4
Box 5	Box 6	Box 7	Box 8

Medium device		
Box 1	Box 2	Box 3
Box 4	Box 5	Box 6
Box 7	Box 8	

Small device	
Box 1	Box 2
Box 3	Box 4
Box 5	Box 6
Box 7	Box 8

Columns on extra small devices are always horizontal and rendered as one column grid layout by default

Jumbotron

- A jumbotron indicates a big box for calling extra attention to some special content or information.

- A jumbotron is displayed as a grey box with rounded corners. It also enlarges the font sizes of the text inside it.

- Use a <div> element with class **.jumbotron** to create a jumbotron.

Web Development with Banu Prakash

In today's world internet is the most popular way of connecting with the people. Learn the essential of web development technologies, so that you can create your own website to connect with the people around the world.

Creating Fixed Layout

- Rows must be placed within a .container for proper alignment and padding.

- Columns can be created inside the rows using the predefined grid classes like

  - .col-xs-*, .col-sm-*, .col-md-* and .col-lg-* where * represent grid number and should be from 1 to 12.

```
<div class="container">
    <div class="row">
        <div class="col-xs-4">
            <h2>HTML and CSS</h2>
            ....
        </div>
        <div class="col-xs-4">
            <h2>Angular JS</h2>
            ....
        </div>
        <div class="col-xs-4">
            <h2>Bootstrap</h2>
            ....
        </div>
    </div>
</div>
```

HTML and CSS

HTML is a markup language that is used for creating web pages. The HTML tutorial section will help you understand the basics of HTML, so that you can create your own web pages or website.CSS is used for describing the presentation of web pages. The CSS tutorial section will help you learn the essentials of CSS, so that you can fine control the style and layout of your HTML document.

Angular JS

A client-side JavaScript Framework for adding interactivity to HTML.Angular helps you organize your JavaScript Angular helps create responsive (as in fast) websites. Angular plays well with jQuery Angular is easy to test

Bootstrap

Bootstrap is a powerful front-end framework for faster and easier web development. The Bootstrap tutorial section will help you learn the techniques of Bootstrap so that you can quickly create your own website.

Creating Fixed Layout

- The illustrated code in previous slide creates a fixed width responsive layout that is:
  - 750px pixels wide on small devices like tablet having screen width ≥768px
  - 970px wide on medium devices like desktop and laptop having screen width ≥992px
  - 1170px wide on large devices like large desktops having screen width ≥1200px.
  - However the layout width will be automatically calculated for devices that has screen width <768px like cell phones.

Creating Fluid Layout

- In Twitter Bootstrap (version 3.2+), you can use the class **.container-fluid** to create the fluid layouts in order to utilize the 100% width of the viewport.

- The class .container-fluid simply applies the horizontal margin with the value auto and left and right padding of 15px on element to offset the left and right margin of -15px (i.e. margin: 0 -15px;) used on the .row.

- Use <div class="container-fluid"> instead of <div class="container">

Creating Responsive Layout

- <div class="col-sm-6 col-md-4 col-lg-2">
  - Its four tiers grids classes provides better control over the layout as well as how it will be rendered on different types of devices like cell phones, tablets, desktop and laptops, large screen devices etc.
  - the orientation of content boxes changes when viewport width crossing or approaching the certain limit

Web Development with Banu Prakash

In today's world internet is the most popular way of connecting with the people. Learn the essential of web development technologies, so that you can create your own website to connect with the people around the world.

HTML and CSS	Angular JS	Bootstrap	Java	Restful services	jQuery
HTML is a markup language that is used for creating web pages. The HTML tutorial section will help you understand the basics of	A client-side JavaScript Framework for adding interactivity to HTML.Angular helps you organize your JavaScript Angular helps create responsive (as in fast) websites. Angular plays well	Bootstrap is a powerful front-end framework for faster and easier web development. The Bootstrap tutorial section will help you learn the techniques of Bootstrap so that you can quickly create your own	Java is a technology	Note on restful services	JavaScript framework.

Bootstrap Text/Typography

- Bootstrap's global default font-size is 14px, with a line-height of 1.428.
- HTML elements will be styled a little bit different by Bootstrap than browser defaults.

h1 Bootstrap heading (36px)

h2 Bootstrap heading (30px)

h3 Bootstrap heading (24px)

h4 Bootstrap heading (18px)

h5 Bootstrap heading (14px)

h6 Bootstrap heading (12px)

Bootstrap Text/Typography

- \<mark\>
  - Bootstrap will style the HTML \<mark\> element in the following way:

  Use the mark element to highlight text.

- \<blockquote\>
  - Bootstrap will style the HTML \<blockquote\> element in the following way:

  For 50 years, WWF has been protecting the future of nature. The world's
  leading conservation organization, WWF works in 100 countries and is
  supported by 1.2 million members in the United States and close to 5
  million globally.

  — From WWF's website

Bootstrap Text/Typography

- \<kbd>
  - Bootstrap will style the HTML \<kbd> element in the following way:

Use `ctrl + p` to open the Print dialog box.

Bootstrap Text/Typography

- Contextual Colors and Backgrounds
  - The classes for text colors are:
  - **.text-muted, .text-primary, .text-success, .text-info, .text-warning,**
  - **and .text-danger:**

```
This text is muted.

This text is important.

This text indicates success.

This text represents some information.

This text represents a warning.

This text represents danger.
```

Bootstrap Text/Typography

- The classes for background colors are:
- **.bg-primary, .bg-success, bg-info, bg-warning, and .bg-danger:**
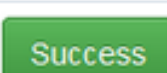
```
This text is important.

This text indicates success.

This text represents some information.

This text represents a warning.

This text represents danger.
```

Buttons

- <button type="button" class="btn btn-default">Default</button>

Button	Class	Description
Default	btn btn-default	Default gray button with gradient.
Primary	btn btn-primary	Provides extra visual weight to indicate primary action button in a set of buttons.
Info	btn btn-info	Can be used as an alternative to the default button.
Success	btn btn-success	Indicates a successful or positive action.
Warning	btn btn-warning	Indicates caution should be taken with this action.
Danger	btn btn-danger	Indicates a dangerous or potentially negative action.
Link	btn btn-link	Deemphasize a button by making it look like a link while maintaining button behavior.

Glyphicons

- What are Glyphicons?
- Glyphicons are icon fonts which you can use in your web projects.
- Glyphicons are not free and require licensing, however their creator has made them available for Bootstrap projects free of cost.
  - It is recommended, as a thank you, we ask you to include an optional link back to GLYPHICONS whenever practical.
- Where to find Glyphicons?
  - glyphicons-halflings-regular.eot
  - glyphicons-halflings-regular.svg
  - glyphicons-halflings-regular.ttf
  - glyphicons-halflings-regular.woff
- Associated CSS rules are present within bootstrap.css and bootstrap-min.css files within css folder of dist folder.

Glyphicons

- **Usage**
  - To use the icons, simply use the following code just about anywhere in your code.
  - Leave a space between the icon and text for proper padding.

    <button type="button" class="btn btn-default btn-lg">
       User
    </button>

Glyphicons [260 glyphs in font format]

- Available Glyphicons

✻	➕	€	€	—	☁	✉	✎
glyphicon glyphicon-asterisk	glyphicon glyphicon-plus	glyphicon glyphicon-euro	glyphicon glyphicon-eur	glyphicon glyphicon-minus	glyphicon glyphicon-cloud	glyphicon glyphicon-envelope	glyphicon glyphicon-pencil
▼	♪	🔍	♥	★	☆	👤	🎞
glyphicon glyphicon-glass	glyphicon glyphicon-music	glyphicon glyphicon-search	glyphicon glyphicon-heart	glyphicon glyphicon-star	glyphicon glyphicon-star-empty	glyphicon glyphicon-user	glyphicon glyphicon-film
▦	⊞	☰	✔	✖	🔍	🔍	⏻
glyphicon glyphicon-th-large	glyphicon glyphicon-th	glyphicon glyphicon-th-list	glyphicon glyphicon-ok	glyphicon glyphicon-remove	glyphicon glyphicon-zoom-in	glyphicon glyphicon-zoom-out	glyphicon glyphicon-off
📶	⚙	🗑	🏠	📄	🕐	Ⓐ	⬇
glyphicon glyphicon-signal	glyphicon glyphicon-cog	glyphicon glyphicon-trash	glyphicon glyphicon-home	glyphicon glyphicon-file	glyphicon glyphicon-time	glyphicon glyphicon-road	glyphicon glyphicon-download-alt

Glyph icons [260 glyphs in font format]

- Available Glyphicons

Glyph icons [260 glyphs in font format]

- Available Glyphicons

Glyph icons [260 glyphs in font format]

- Available Glyphicons

Bootstrap tables

- Bootstrap's class .table <table class="table">

Product ID	Name	Price
PRD1234	HP Laptop 15" screen, Core i5	45000.00
PRD9883	Logitech Mouse	350.00
PRD9800	Track Pad	1234.44

- Tables with Alternate Background
  - Bootstrap's class **.table-striped** to base class **.table**

Product ID	Name	Price
PRD1234	HP Laptop 15" screen, Core i5	45000.00
PRD9883	Logitech Mouse	350.00
PRD9800	Track Pad	1234.44

Bootstrap tables

- Optional Emphasis Classes for Table Rows
  - There are some contextual classes to emphasize the row or individual cells data like success, warning, danger, etc. through colouring its background.
  - Example: <tr class="success">

Row	Bill	Payment Date	Payment Status
1	Credit Card	04/07/2014	Call in to confirm
2	Water	01/07/2014	Paid
3	Internet	05/07/2014	Change plan
4	Electricity	03/07/2014	Pending
5	Telephone	06/07/2014	Due

Bootstrap tables

- Table with Borders
  - <table class="table table-bordered">

- Enable Hover State on Table Rows
  - <table class="table table-hover">

- Responsive tables
  - Responsive tables enable scrolling on small devices (screen width under 768px).
  - However viewing responsive tables on other devices having screen width larger than 768px, you will not see any difference
  - <div class="table-responsive"> <table class="table">

Bootstrap List

```
<ul class="list-unstyled">
    <li>Home</li>
    <li>Products</li>
        <ul>
            <li>Gadgets</li>
            <li>Accessories</li>
        </ul>
    </li>
    <li>About Us</li>
    <li>Contact</li>
</ul>


<ul class="list-group">
    <li class="list-group-item">Pictures</li>
    <li class="list-group-item">Documents</li>
    <li class="list-group-item">Music</li>
    <li class="list-group-item">Videos</li>
</ul>
```

Home
Products
    ○ Gadgets
    ○ Accessories
About Us
Contact

Pictures

Documents

Music

Videos

Bootstrap List

```html
<ul class="list-inline">
    <li>Home</li>
    <li>Products</li>
    <li>About Us</li>
    <li>Contact</li>
</ul>
```

Home   Products   About Us   Contact

Bootstrap List using <div>

```html
<div class="list-group">
    <a href="#" class="list-group-item active">
        <span class="glyphicon glyphicon-camera"></span> Pictures
        <span class="badge">25</span>
    </a>
    <a href="#" class="list-group-item">
        <span class="glyphicon glyphicon-file"></span> Documents
        <span class="badge">145</span>
    </a>
    <a href="#" class="list-group-item">
        <span class="glyphicon glyphicon-music"></span> Music
        <span class="badge">50</span>
    </a>
    <a href="#" class="list-group-item">
        <span class="glyphicon glyphicon-film"></span> Videos
        <span class="badge">8</span>
    </a>
</div>
```

📷 Pictures		25
📄 Documents		145
🎵 Music		50
🎞 Videos		8

Bootstrap forms

- Bootstrap provides three types of form layouts:
  - Vertical form (this is default)
  - Horizontal form
  - Inline form

- Standard rules for all three form layouts:
  - Always use <form **role="form">**
  - Wrap labels and form controls in **<div class="form-group">** (needed for optimum spacing)
  - Add class **.form-control** to all textual <input>, <textarea>, and <select> elements

Bootstrap forms

- Vertical form

```html
<h2>Vertical (basic) form</h2>
<form role="form" >
    <div class="form-group">
        <label for="email" >Email:</label>
        <input type="email" class="form-control"
            id="email" placeholder="Enter email">
    </div>
    <div class="form-group">
        <label for="pwd" >Password:</label>
        <input type="password" class="form-control"
            id="pwd" placeholder="Enter password">
    </div>
    <div class="checkbox">
        <label ><input type="checkbox"> Remember me</label>
    </div>
    <button type="submit" class="btn btn-default">Submit</button>
</form>
```

- If you don't include a label for every input, screen readers will have trouble with your forms.
- You can hide the labels for all devices, except screen readers, by using the .sr-only class
    - [<label class="sr-only" for="pwd">Password:</label>]

Bootstrap forms

- Bootstrap Horizontal Form
  - Add class .form-horizontal to the <form> element
  - Add class .control-label to all <label> elements

```html
<form class="form-horizontal" role="form">
  <div class="form-group">
    <label class="control-label col-sm-2" for="email">Email:</label>
    <div class="col-sm-10">
      <input type="email" class="form-control"
          id="email" placeholder="Enter email">
    </div>
  </div>
  <div class="form-group">
    <label class="control-label col-sm-2" for="pwd">Password:</label>
    <div class="col-sm-10">
      <input type="password" class="form-control"
          id="pwd" placeholder="Enter password">
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <button type="submit" class="btn btn-default">Submit</button>
    </div>
  </div>
</form>
```

Email: Enter email

Password: Enter password

Submit

Styling Images

- Using the Bootstrap built-in classes you can easily style images such as making **the round cornered or circular images**, or give them effect like thumbnails.

```html
<img src="images/banu.jpg" class="img-rounded" alt="Rounded Profile picture">
<img src="images/banu.jpg" class="img-circle" alt="Circular Profile picture">
<img src="images/banu.jpg" class="img-thumbnail" alt="Thumbnail picture">
```

Media Objects

- To create a layout that contains left- or right-aligned image along with the textual content like blog comments

Amitabh Bachchan *Posted on Feb 6, 2015*
Back from the first trial public show of #Shamitabh .. the applause at Interval and twice at end of film says it all .. !!.

```html
<div class="media">
    <a href="#" class="pull-left">
        <img src="images/amitabh.jpeg" class="media-object" alt="Amitabh Image">
    </a>
    <div class="media-body">
        <h4 class="media-heading">Amitabh Bachchan <small><i>Posted on Feb 6,
        2015</i></small></h4>
        <p>Back from the first trial public show of #Shamitabh .. the applause
        at
            Interval and twice at end of film says it all .. !!.</p>
    </div>
</div>
```

Navigation Components

- class="nav nav-tabs"


```html
<ul class="nav nav-tabs">
    <li class="active">
        <a href="#"><span class="glyphicon glyphicon-home"></span> Home</a>
    </li>
    <li>
        <a href="#"><span class="glyphicon glyphicon-user"></span> Profile</a>
    </li>
    <li>
        <a href="#"><span class="glyphicon glyphicon-envelope"></span> Messages</a>
    </li>
</ul>
```

Navigation Components

- class="nav nav-pills"

- class="nav nav-pills nav-stacked"

Navigation Components

- class="dropdown" and

- class="dropdown-menu"


```html
<ul class="nav nav-tabs">
    <li class="active"><a href="#">Home</a></li>
    <li><a href="#">Profile</a></li>
    <li class="dropdown">
        <a href="#" data-toggle="dropdown" class="dropdown-toggle">
            Messages <b class="caret"></b>
        </a>
        <ul class="dropdown-menu">
            <li><a href="#">Inbox</a></li>
            <li><a href="#">Drafts</a></li>
            <li><a href="#">Sent Items</a></li>
            <li class="divider"></li>
            <li><a href="#">Trash</a></li>
        </ul>
    </li>
</ul>
```

Navigation Bars

- A navigation bar is a navigation header that is placed at the top of the page.

- With Bootstrap, a navigation bar can extend or collapse, depending on the screen size.

Navigation Bars


```
<nav class="navbar navbar-inverse navbar-fixed-top">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">mindtree.com</a>
        <button type="button" data-target="#navbarCollapse"
                data-toggle="collapse" class="navbar-toggle">
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
    </div>

    <div id="navbarCollapse" class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">Home</a></li>
        <li><a href="#">Page 1</a></li>
        <li><a href="#">Page 2</a></li>
      </ul>
    </div>
  </div>
</nav>
```
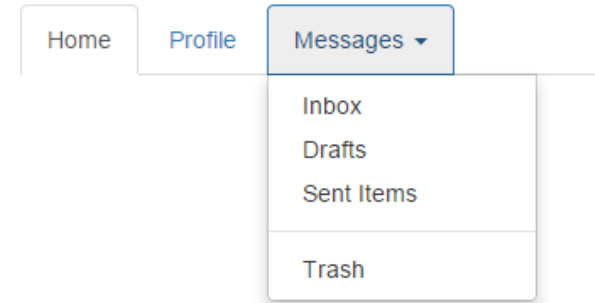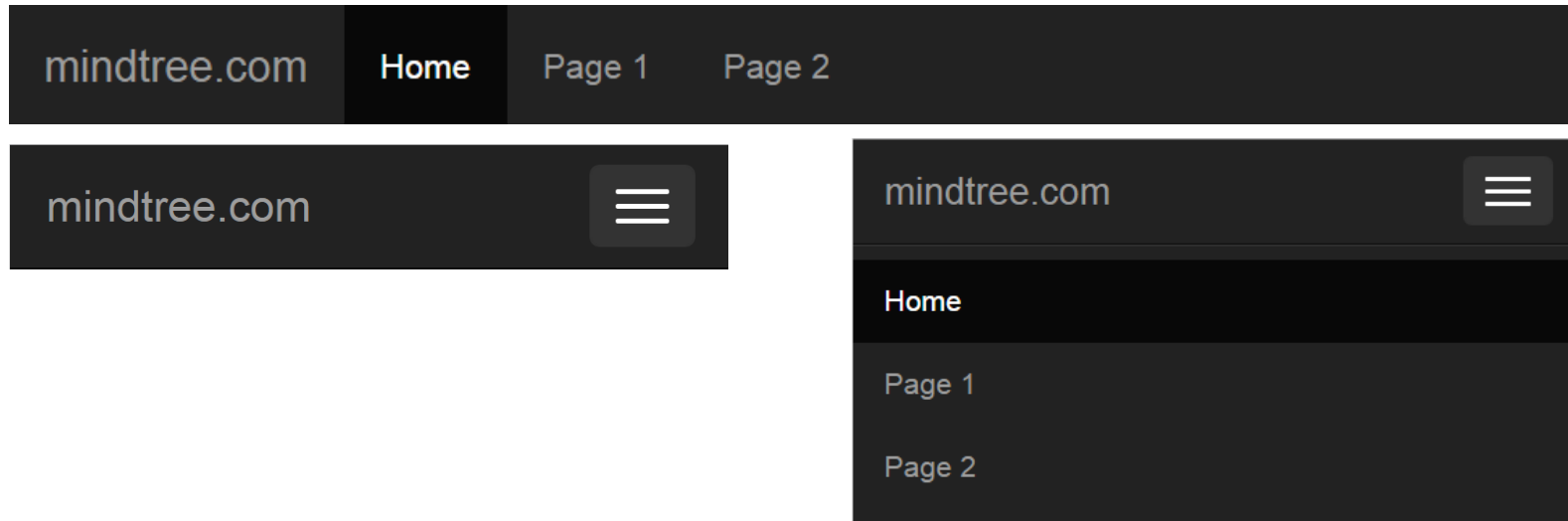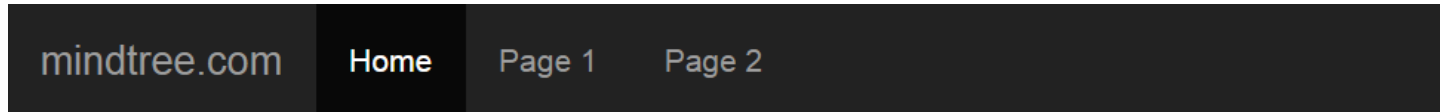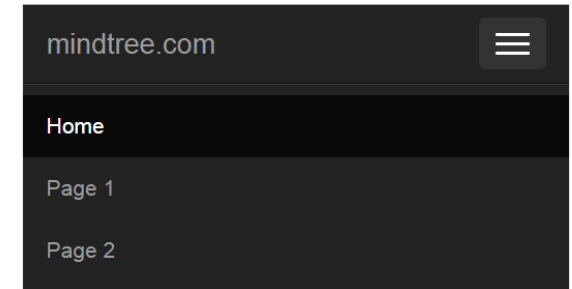
Panels

- Panel component are used when you want to put your DOM component in a box. To get a basic panel, just add **class .panel** to the <div> element.

```html
<div class="panel panel-default">
    <div class="panel-heading">This Page is Disabled</div>
    <div class="panel-body">This page is temporarily disabled by the site administrator for some reason.<br>
    <a href="#">Click here</a> to enable the page.</div>
</div>

<div class="panel panel-primary">
    <div class="panel-heading">
        <h3 class="panel-title">301 Moved Permanently</h3>
    </div>
    <div class="panel-body">The requested page has been permanently moved to a new location.</div>
</div>
```

This Page is Disabled

This page is temporarily disabled by the site administrator for some reason.
Click here to enable the page.

301 Moved Permanently

The requested page has been permanently moved to a new location.

Accordion

- Accordion menus or navigation are widely used on the website to manage large content and navigation lists.

1. What is HTML?

2. What is Twitter Bootstrap?

Twitter Bootstrap is a powerful front-end framework for faster and easier web development. It is a collection of CSS and HTML conventions. Learn more.

3. What is CSS?

Accordion

```html
<div id="accordion" class="panel-group">
    <div class="panel panel-default">
        <div class="panel-heading">
            <h4 class="panel-title">
                <a data-toggle="collapse" data-parent="#accordion" href="#collapseOne">1. What is HTML?</a>
            </h4>
        </div>
        <div id="collapseOne" class="panel-collapse collapse">
            <div class="panel-body">
                <p>HTML stands for...</p>
            </div>
        </div>
    </div>
    <div class="panel panel-default">
        <div class="panel-heading">
            <h4 class="panel-title">
                <a data-toggle="collapse" data-parent="#accordion" href="#collapseTwo">2. What is Twitter
                Bootstrap?</a>
            </h4>
        </div>
        <div id="collapseTwo" class="panel-collapse collapse in">
            <div class="panel-body">
                <p>Twitter Bootstrap is a....</a></p>
            </div>
        </div>
    </div>
</div>
```

Breadcrumbs

- A breadcrumb is a navigation scheme that indicates the user's location in a website or web application. Breadcrumb navigation can greatly enhance the accessibility of websites having large number of pages.

```html
<ul class="breadcrumb">
    <li><a href="#">Home</a></li>
    <li><a href="#" >Products</a></li>
    <li class="active">Accessories</li>
</ul>
```

Home / Products / Accessories

Carousel

- Carousels popularly known as slide shows are some of the best ways of showcasing huge amount of contents within a small space on the web pages

Carousel

```html
<div id="myCarousel" class="carousel slide" data-interval="3000" data-ride="carousel">
<!-- Carousel indicators -->
<ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
</ol>
<!-- Carousel items -->
<div class="carousel-inner">
    <div class="active item">
        <h2>Slide 1</h2>
        <div class="carousel-caption">
          <h3>First slide label</h3>
        </div>
    </div>
    <div class="item">
        <h2>Slide 2</h2>
        <div class="carousel-caption">
          <h3>Second slide label</h3>
        </div>
    </div>
</div>
<!-- Carousel nav -->
<a class="carousel-control left" href="#myCarousel" data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left"></span>
</a>
<a class="carousel-control right" href="#myCarousel" data-slide="next">
    <span class="glyphicon glyphicon-chevron-right"></span>
</a>
</div>
```

```css
h2{
    color: #666;
    padding-top: 90px;
    font-size: 52px;
}
.item{
    background: #333;
    text-align: center;
    height: 300px ;
}
```