1 CSS for Styling

The good, the bad and the… ugly!

```html
<p>
<font face="Arial">Shashdot.</font>
News for <b>nerds!!</b> You will <i>never</i>, <u>EVER</u>
be
<font size="+4" color="red">BORED</font> here!
</p>
```
HTML

Slashdot. News for **nerds!!** You will never, <u>EVER</u> be BORED here!

output

- ☐ Tags such as b, i, u, and font are discouraged in strict XHTML

- ☐ Why is this bad?

CS380

Cascading Style Sheets (CSS)

- Describes the appearance, layout, and presentation of information on a web page
  - HTML describes **the content** of the page
- Describes *how* information is to be displayed, not *what* is being displayed
- Can be embedded in HTML document or placed into separate .css file

CS380

Basic CSS rule syntax

```css
selector {
property: value;
property: value;
...
property: value;
}                                                    CSS
```

```css
p {
font-family: sans-serif;
color: red;
}                                                    CSS
```

□ A CSS file consists of one or more **rules**

□ Each rule starts with a **selector**

□ A selector specifies an HTML element(s) and then applies style **properties** to them

  ◻ a selector of * selects all elements

Attaching a CSS file <link>

```
<head>
...
<link href="filename" type="text/css" rel="stylesheet" />
...
</head>                                                    HTML
```

```
<link href="style.css" type="text/css" rel="stylesheet" />
<link href="http://www.google.com/uds/css/gsearch.css"
rel="stylesheet" type="text/css" />
                                                          HTML
```

□ A page can link to multiple style sheet files

  ▪ In case of a conflict (two sheets define a style for the same HTML element), the latter sheet's properties will be used

CS380

Embedding style sheets: <style>

```
<head>
<style type="text/css">
p { font-family: sans-serif; color: red; }
h2 { background-color: yellow; }
</style>
</head>
                                                          HTML
```

- ☐ CSS code can be embedded within the head of an HTML page
- ☐ B*ad style* and should be avoided when possible (why?)

CS380

Inline styles: the style attribute

```
<p style="font-family: sans-serif; color: red;">
This is a paragraph</p>
```
HTML

This is a paragraph

output

- ☐ Higher precedence than embedded or linked styles

- ☐ Used for one-time overrides and styling a particular element

- ☐ B*ad style* and should be avoided when possible (why?)

CS380

For more study materials>www.ktustudents.in

CSS properties for colors

```
p {
color: red;
background-color: yellow;
}
```
CSS

This paragraph uses the style above   *output*

property	description
color	color of the element's text
background-color	color that will appear behind the element

For more study materials>www.ktustudents.in

Specifying colors

```
p { color: red; }
h2 { color: rgb(128, 0, 196); }
h4 { color: #FF8800; }
                                            CSS
```

This paragraph uses the first style above

This h2 uses the second style above.

This h4 uses the third style above.

output

- color names: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white (white), yellow
- RGB codes: red, green, and blue values from 0 (none) to 255 (full)
- hex codes: RGB values in base 16 from 00 (0, none) to FF (255, full)

Grouping styles

```css
p, h1, h2 {
color: green;
}
h2 {
background-color: yellow;
}                                                           CSS
```

This paragraph uses the above style.

This h2 uses the above styles.

output

- ☐ A style can select multiple elements separated by commas
- ☐ The individual elements can also have their own styles

CS380

CSS comments /*...*/

```css
/* This is a comment.
It can span many lines in the CSS file. */
p {
color: red; background-color: aqua;
}                                                    CSS
```

- CSS (like HTML) is usually not commented as rigorously as programming languages such as Java

- The // single-line comment style is NOT supported in CSS

- The <!-- ... --> HTML comment style is also NOT supported in CSS

CS380

CSS properties for fonts

property	description
font-family	which font will be used
font-size	how large the letters will be drawn
font-style	used to enable/disable italic style
font-weight	used to enable/disable bold style

Complete list of font properties (http://www.w3schools.com/css/css_reference.asp#font)

CS380

font-family

```css
p {
font-family: Georgia;
}
h2 {
font-family: "Courier New";
}                                              CSS
```

This paragraph uses the first style above.

This h2 uses the second style above.

output

☐ Enclose multi-word font names in quotes

CS380

More about font-family

```css
p {
font-family: Garamond, "Times New Roman", serif;
}                                                       CSS
```

This paragraph uses the above style.

output

- We can specify multiple fonts from highest to lowest priority

- Generic font names:

  - serif, sans-serif, cursive, FANTASY, monospace

- If the first font is not found on the user's computer, the next is tried

- Placing a generic font name at the end of your font-family value, ensures that every computer will use a valid font

CS380

font-size

```css
p {
      font-size: 24pt;
}
                                                    CSS
```

This paragraph uses the style above.

output

- units: pixels (px) vs. point (pt) vs. m-size (em)

  16px, 16pt, 1.16em

- vague font sizes: xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger

- percentage font sizes, e.g.: 90%, 120%

CS380

font-size

```css
p {
        font-size: 24pt;
}
                                                    CSS
```

This paragraph uses the style above.

output

- **pt** specifies number of point, where a point is 1/72 of an inch onscreen

- **px** specifies a number of pixels on the screen

- **em** specifies number of m-widths, where 1 em is equal to the font's current size

font-weight, font-style

```css
p {
font-weight: bold;
font-style: italic;
}                                                      CSS
```

This paragraph uses the style above.

output

☐ Either of the above can be set to normal to turn them off (e.g. headings)

CS380

CSS properties for text

property	description
text-align	alignment of text within its element
text-decoration	decorations such as underlining
line-height, word-spacing, letter-spacing	gaps between the various portions of the text
text-indent	indents the first letter of each paragraph

Complete list of text properties (http://www.w3schools.com/css/css_reference.asp#text)

CS380

text-align

```
blockquote { text-align: justify; }
h2 { text-align: center; }
```
CSS

The Gollum's Quote

We wants it, we needs it. Must have the precious. They stole it from us. Sneaky little hobbitses. Wicked, tricksy, false!

output

- **text-align can be** left, right, center, or justify

text-decoration

```
p {
text-decoration: underline;
}                                                    CSS
```

This paragraph uses the style above.

output

- **can also be** overline, ~~line-through~~, blink, or none

- **effects can be combined:**

text-decoration: overline underline;

CS380

The list-style-type property

```
ol { list-style-type: lower-roman; }
                                                          CSS
```

- Possible values:

  i. `none` : No marker

  ii. `disc` (default), `circle, square`

  iii. `Decimal`: 1, 2, 3, etc.

  iv. `decimal-leading-zero`: 01, 02, 03, etc.

  v. `lower-roman`: i, ii, iii, iv, v, etc.

  vi. `upper-roman`: I, II, III, IV, V, etc.

  vii. `lower-alpha`: a, b, c, d, e, etc.

  viii. `upper-alpha`: A, B, C, D, E, etc.

  x. `lower-greek`: alpha, beta, gamma, etc.

  others: hebrew, armenian, georgian, cjk-ideographic, hiragana...

Body styles

```css
body {
font-size: 16px;
}
                                                    CSS
```

- ☐ Applies a style to the entire body of your page
- ☐ Saves you from manually applying a style to each element

For more study materials>www.ktustudents.in

Cascading Style Sheets

- Properties of an element *cascade* together in this order:
  - browser's default styles
  - external style sheet files (in a <link> tag)
  - internal style sheets (inside a <style> tag in the page's header)
  - inline style (the style attribute of the HTML element)

Inheriting styles

```css
body { font-family: sans-serif; background-color: yellow;
}
p { color: red; background-color: aqua; }
a { text-decoration: underline; }
h2 { font-weight: bold; text-align: center; }
```
CSS

This is a heading

A styled paragraph. Previous slides are available on the website.

- A bulleted list
output

- ☐ when multiple styles apply to an element, they are inherited

- ☐ a more tightly matching rule can override a more general inherited rule

CS380

Styles that conflict

```
p, h1, h2 { color: blue; font-style: italic; }
h2 { color: red; background-color: yellow; }
```
CSS

This paragraph uses the first style above.

This heading uses both styles above.

output

- when two styles set conflicting values for the same property, the latter style takes precedence

CS380

W3C CSS Validator

```
<p>
<a href="http://jigsaw.w3.org/css-
validator/check/referer">
<img src="http://jigsaw.w3.org/css-validator/images/vcss"
alt="Valid CSS!" /></a>
</p>
```
CSS

output

- □ jigsaw.w3.org/css-validator/

- □ checks your CSS to make sure it meets the official CSS specifications

CS380

CSS properties for backgrounds

property	description
background-color	color to fill background
background-image	image to place in background
background-position	placement of bg image within element
background-repeat	whether/how bg image should be repeated
background-attachment	whether bg image scrolls with page
background	shorthand to set all background properties

For more study materials>www.ktustudents.in

background-image

```css
body {
background-image: url("images/draft.jpg");
}
                                                              CSS
```

This is the first paragraph

This is the second paragraph...
It occupies 2 lines

- background image/color fills the element's content area

background-repeat

```css
body {
background-image: url("images/draft.jpg");
background-repeat: repeat-x;
}
                                                            CSS
```

This is the first paragraph

This is the second paragraph...
It occupies 2 lines

□ can be repeat (default), repeat-x, repeat-y, or no-repeat

background-position

```css
body {
background-image: url("images/draft.jpg");
background-repeat: no-repeat;
background-position: 370px 20px;
}                                                    CSS
```

This is the first paragraph

This is the second paragraph...
It occupies 2 lines

DRAFT

- □ value consists of two tokens, each of which can be top, left, right, bottom, center, a percentage, or a length value in px, pt, etc.

- □ value can be negative to shift left/up by a given amount

CS380

Aside: Favorites icon ("favicon")

```html
<link href="filename" type="MIME type" rel="shortcut icon"
/>
                                                            HTML
```

```html
<link href="yahoo.gif" type="image/gif" rel="shortcut
icon" />
                                                            HTML
```


- The link tag, placed in the HTML page's head section, can specify an icon

  - this icon will be placed in the browser title bar and bookmark/favorite

CS380

Bootstrap

What is Bootstrap

Bootstrap is a sleek, intuitive, and powerful mobile first front-end framework for faster and easier web development. It uses HTML, CSS and JavaScript.

History

Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter. It was released as an open source product in August 2011 on GitHub.

Why use Bootstrap?

- **Mobile first approach:**
  Since Bootstrap 3, the framework consists of Mobile first styles throughout the entire library instead of in separate files.

- **Browser Support:**
  It is supported by all popular browsers.
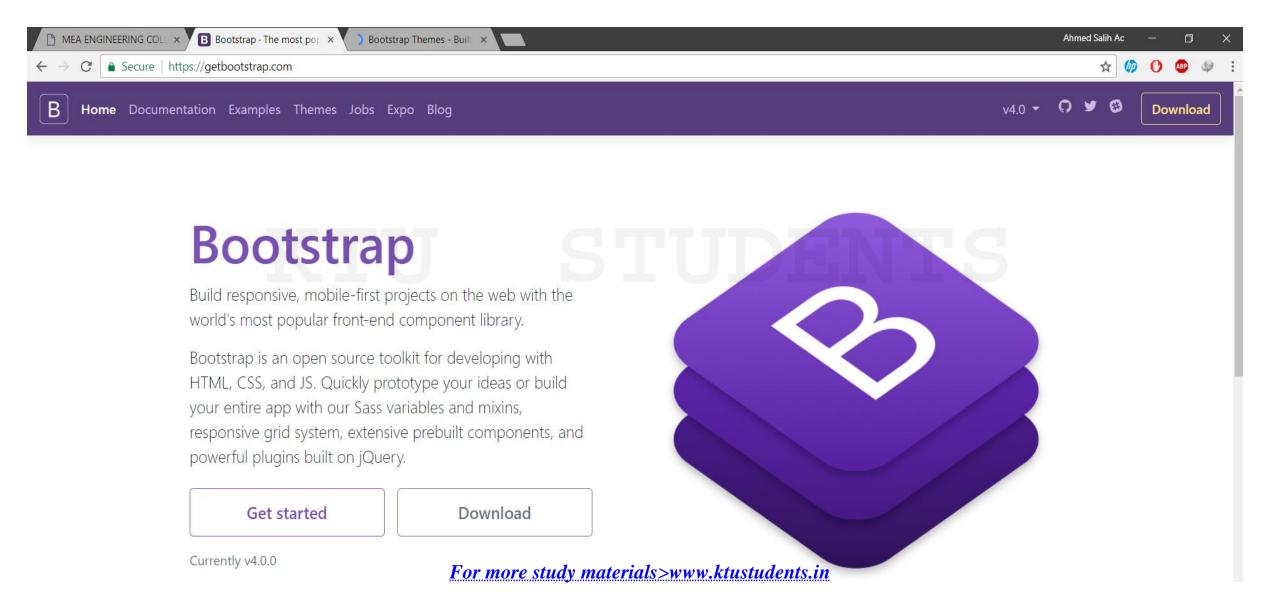
- **Easy to get started:**
  With just the knowledge of HTML and CSS anyone can get started with Bootstrap. Also the Bootstrap official site has a good documentation.

- **Responsive design:**
  Bootstrap's responsive CSS adjusts to Desktops, Tablets and Mobiles. More about responsive design in the chapter Bootstrap Responsive Design

- Provides a clean and uniform solution for building an interface for developers.
- It contains beautiful and functional built-in components which are easy to customize.
- It also provides web based customization.
- And best of all it is an open source

https://getbootstrap.com

Direct Documentation
CSS

Copy-paste the stylesheet <link> into your <head> before all other stylesheets to load our CSS.

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
crossorigin="anonymous">
```

JS

Many of our components require the use of JavaScript to function. Specifically, they require jQuery, Popper.js, and our own JavaScript plugins. Place the following <script>s near the end of your pages, right before the closing </body> tag, to enable them. jQuery must come first, then Popper.js, and then our JavaScript plugins.

```
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
```

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>
```
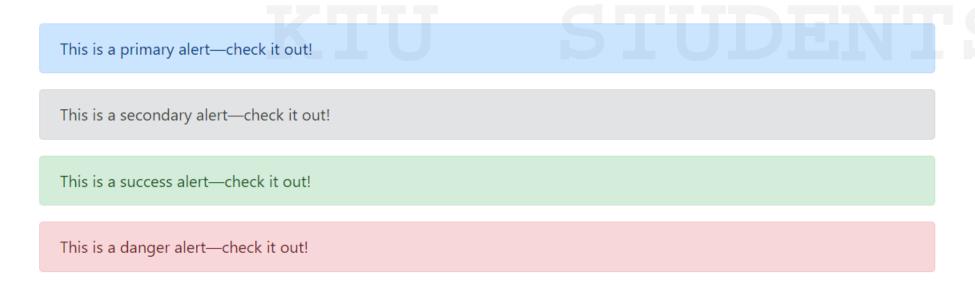
```
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl" crossorigin="anonymous"></script>
```

External Documentation

```
<meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

        <!-- Latest compiled and minified CSS -->
    <link rel="stylesheet" href="bootstrap/dist/css/bootstrap.min.css" >

    <script src="bootstrap/js/tests/vendor/jquery.min.js"></script>

    <!-- Latest compiled and minified JavaScript -->
    <script src="bootstrap/dist/js/bootstrap.min.js"></script>
```

Alerts

```
<div class="alert alert-primary" role="alert"> This is a primary alert—check it out! </div>
 <div class="alert alert-secondary" role="alert"> This is a secondary alert—check it out! </div>
<div class="alert alert-success" role="alert"> This is a success alert—check it out! </div>
 <div class="alert alert-danger" role="alert"> This is a danger alert—check it out! </div>
```

This is a primary alert—check it out!

This is a secondary alert—check it out!

This is a success alert—check it out!

This is a danger alert—check it out!

Link color

Use the .alert-link utility class to quickly provide matching colored links within any alert.

<div class="alert alert-primary" role="alert"> This is a primary alert with an example link. Give it a click if you like. </div>

<div class="alert alert-secondary" role="alert"> This is a secondary alert with an example link. Give it a click if you like. </div>

<div class="alert alert-success" role="alert"> This is a success alert with an example link. Give it a click if you like. </div>

This is a primary alert with **an example link**. Give it a click if you like.

This is a secondary alert with **an example link**. Give it a click if you like.

This is a success alert with **an example link**. Give it a click if you like.

This is a danger alert with **an example link**. Give it a click if you like.

Button


```
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>

<button type="button" class="btn btn-link">Link</button>
```

Sizes

Fancy larger or smaller buttons? Add `.btn-lg` or `.btn-sm` for additional sizes.

Large button **Large button**

Copy

```html
<button type="button" class="btn btn-primary btn-lg">Large button</button>
<button type="button" class="btn btn-secondary btn-lg">Large button</button>
```

Small button **Small button**

Copy

```html
<button type="button" class="btn btn-primary btn-sm">Small button</button>
<button type="button" class="btn btn-secondary btn-sm">Small button</button>
```

Outline buttons

In need of a button, but not the hefty background colors they bring? Replace the default modifier classes with the `.btn-outline-*` ones to remove all background images and colors on any button.

Primary Secondary Success Danger Warning Info Light Dark

Copy

```html
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-secondary">Secondary</button>
<button type="button" class="btn btn-outline-success">Success</button>
<button type="button" class="btn btn-outline-danger">Danger</button>
<button type="button" class="btn btn-outline-warning">Warning</button>
<button type="button" class="btn btn-outline-info">Info</button>
<button type="button" class="btn btn-outline-light">Light</button>
<button type="button" class="btn btn-outline-dark">Dark</button>
```

Links

Using the contextual `.badge-*` classes on an `<a>` element quickly provide *actionable* badges with hover and focus states.

<p>Primary Secondary Success Danger Warning Info Light Dark</p>

```
<a href="#" class="badge badge-primary">Primary</a>
<a href="#" class="badge badge-secondary">Secondary</a>
<a href="#" class="badge badge-success">Success</a>
<a href="#" class="badge badge-danger">Danger</a>
<a href="#" class="badge badge-warning">Warning</a>
<a href="#" class="badge badge-info">Info</a>
<a href="#" class="badge badge-light">Light</a>
<a href="#" class="badge badge-dark">Dark</a>
```

Span

<p>Primary Secondary Success Danger Warning Info Light Dark</p>

Copy

```
<span class="badge badge-primary">Primary</span>
<span class="badge badge-secondary">Secondary</span>
<span class="badge badge-success">Success</span>
<span class="badge badge-danger">Danger</span>
<span class="badge badge-warning">Warning</span>
<span class="badge badge-info">Info</span>
<span class="badge badge-light">Light</span>
<span class="badge badge-dark">Dark</span>
```

Form

@ | Username

Recipient's username | @example.com

Your vanity URL

https://example.com/users/

$ | .00

With textarea

```html
<div class="input-group mb-3">
 <div class="input-group-prepend">
 <span class="input-group-text" id="basic-addon1">@</span> </div>
<input type="text" class="form-control" placeholder="Username" aria-label="Username" aria-describedby="basic-addon1">
</div>

 <div class="input-group mb-3">
<input type="text" class="form-control" placeholder="Recipient's username" aria-label="Recipient's username" aria-describedby="basic-addon2">
<div class="input-group-append">
<span class="input-group-text" id="basic-addon2">@example.com</span>
</div>
 </div>

<label for="basic-url">Your vanity URL</label>
<div class="input-group mb-3">
 <div class="input-group-prepend">
<span class="input-group-text" id="basic-addon3">https://example.com/users/</span>
</div>
 <input type="text" class="form-control" id="basic-url" aria-describedby="basic-addon3">
 </div>
```