CS603 – Web Engineering

PREPARED BY: DR. REEMA PATEL

HTML

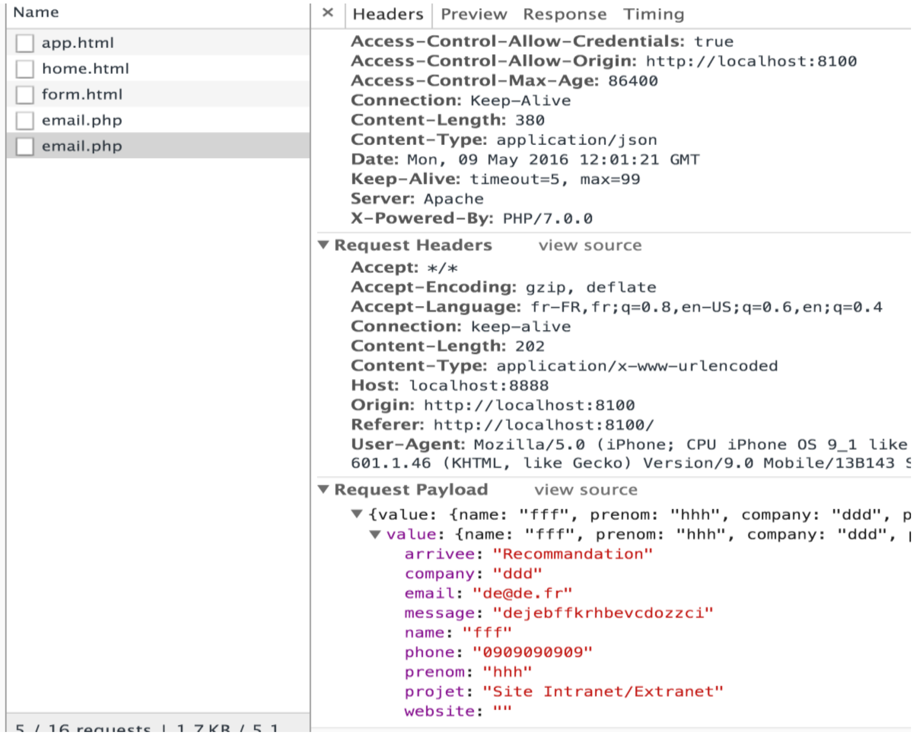
- HTML (Hypertext Markup Language) is a plain-text, human-readable language that is used for representing content on the Web

- HTML specifies how to **structure the data**
  - but not (necessarily) how to display it

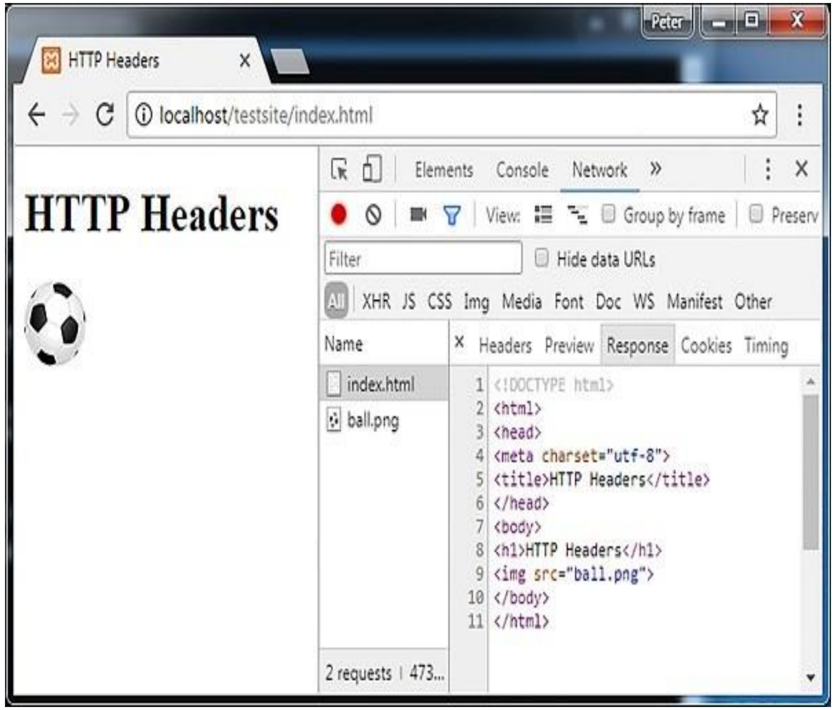
- The browser chooses how to display the content

HTML and HTTP

- When a browser makes a request for a document (e.g. first.html), the HTML will be returned in the body of the response and displayed in the browser



```
Access-Control-Allow-Credentials: true
Access-Control-Allow-Origin: http://localhost:8100
Access-Control-Max-Age: 86400
Connection: Keep-Alive
Content-Length: 380
Content-Type: application/json
Date: Mon, 09 May 2016 12:01:21 GMT
Keep-Alive: timeout=5, max=99
Server: Apache
X-Powered-By: PHP/7.0.0
▼ Request Headers      view source
Accept: */*
Accept-Encoding: gzip, deflate
Accept-Language: fr-FR,fr;q=0.8,en-US;q=0.6,en;q=0.4
Connection: keep-alive
Content-Length: 202
Content-Type: application/x-www-urlencoded
Host: localhost:8888
Origin: http://localhost:8100
Referer: http://localhost:8100/
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 9_1 like
601.1.46 (KHTML, like Gecko) Version/9.0 Mobile/13B143 S
▼ Request Payload      view source
  ▼ {value: {name: "fff", prenom: "hhh", company: "ddd", p
    ▼ value: {name: "fff", prenom: "hhh", company: "ddd", |
        arrivee: "Recommandation"
        company: "ddd"
        email: "de@de.fr"
        message: "dejebffkrhbevcdozzci"
        name: "fff"
        phone: "0909090909"
        prenom: "hhh"
        projet: "Site Intranet/Extranet"
        website: ""
```

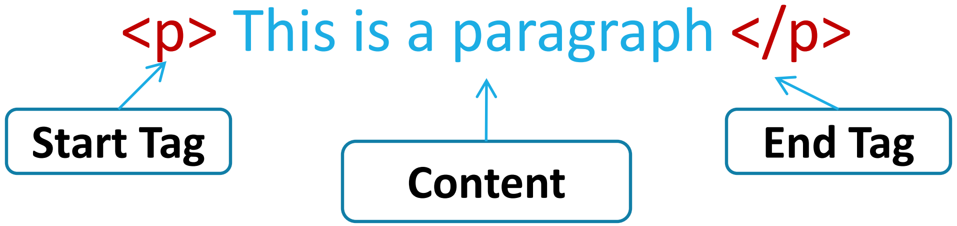


```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>HTTP Headers</title>
</head>
<body>
<h1>HTTP Headers</h1>
<img src="ball.png">
</body>
</html>
```

Introduction

- Document contain elements
- Each element generally includes a start tag, some content, and an end tag

- HTML structure is hierarchical
- Elements may be nested, i.e. content of an element can be another element



<p> This is a paragraph </p>

Important HTML Tags - <!DOCTYPE>

- NOT actually a tag, rather a declaration to the web browser of what version of html the following document uses

- For HTML 5, doctype declaration is as follows:
  - <!DOCTYPE html>

- This declaration must be the very first line of an html document

Important HTML Tags: <html>

- After the declaration, HTML pages must start with the <html> tag

- Often referred to as the root element because it can be considered the root of the tree-like structure of elements in an HTML page

```
<!DOCTYPE html>

<html>

</html>
```

Important HTML Tags: <head>

- Contains information about the document, not content
- Common elements included within <head> </head>:
  - <title> - contains page title, displayed in browser’s title bar
  - <link> - used to add CSS style-sheets and icon pages
  - <meta> - used to specify metadata like page description and keywords
  - <script> - used to add JavaScript code to the page

Important HTML Tags: <body>

- Appears directly beneath the head element
- Contains all web page content (images, text, etc)
- Most web pages have one single body element

```
<!DOCTYPE html>
<html>
<head> …. </head>

<body>
		Hello!
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
        <meta charset="utf-8">
        <title>My First Web Page</title>
        <meta name="Description" content="Learning about
HTML">
        <meta name="keywords" content="html, web
development">
        <link rel="stylesheet" type="text/css" href="/style.css">
        <script src="code.js" type="text/javascript"></script>
</head>
<body>
        Hello!
</body>
</html>
```

Important HTML Tags: <p>

- Appears anywhere within the body to represent a paragraph of text expressing a single thought
- Usually displayed with a vertical space before and after paragraph

```
<!DOCTYPE html>
<html>
<head>… </head>
<body>
<p>This is a paragraph within the p tag. The paragraph should be
surrounded with vertical white space before and after paragraph</p>
</body>
</html>
```

HTML Content

- The HTML specifies the **structure** but not how the content will be **displayed**
- It is up to the browser to decide how to display the content

```
<!DOCTYPE html>
<html>
<body>
<p>This is some text

            Some more text
            Some more text
</p>
</body>
</html>
```

Out put:

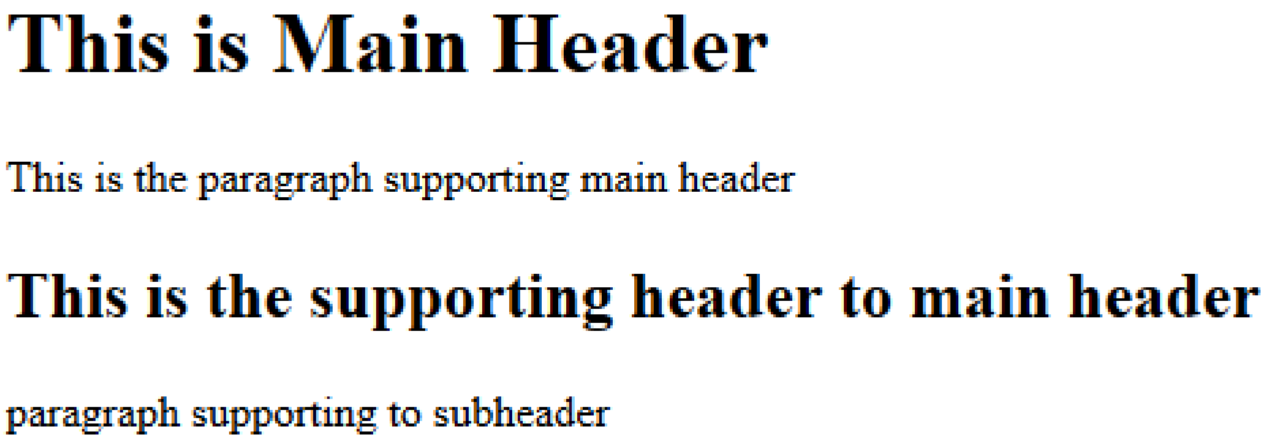
This is some text Some more text Some more text

Important HTML Tags: <h#>

- Referred to as section heading tags
- HTML supports 6 heading tags
- **h1** - <h1> **-** used for most important titles (ex: title of entire webpage)
- **h2** - <h2> - used for next important subheader
- **h3** - <h3> .. <h6> - used for least important subheaders
- **h4** - Magnitude of importance affects font size that will be displayed
- **h5** - Can be used to divide content into readable subsections
- h6 - Browser determines font and size depending on header magnitude

Important HTML Tags: <h#>

```
<h1>This is Main Header</h1>
<p> This is the paragraph supporting main header</p>

<h2>This is the supporting header to main header </h2>
<p>paragraph supporting to subheader</p>
```



This is Main Header

This is the paragraph supporting main header

This is the supporting header to main header

paragraph supporting to subheader

Important HTML Tags: <b> and <i>

- <b> - indicates that the text should be **bold**
- <i> - indicated that the text should be *italicized*

- Similar tags are **<strong>** and **<em>,** respectively, which are meant to demonstrate that the text is “important”

Important HTML Tags: <hr> and <br>

- Both tags are used to mark a break in the content
- <hr> represent more serious shift in content,
  - Separate content by inserting a visible line between preceding and subsequent content
- <br> - represent a single line break, inserts a blank line

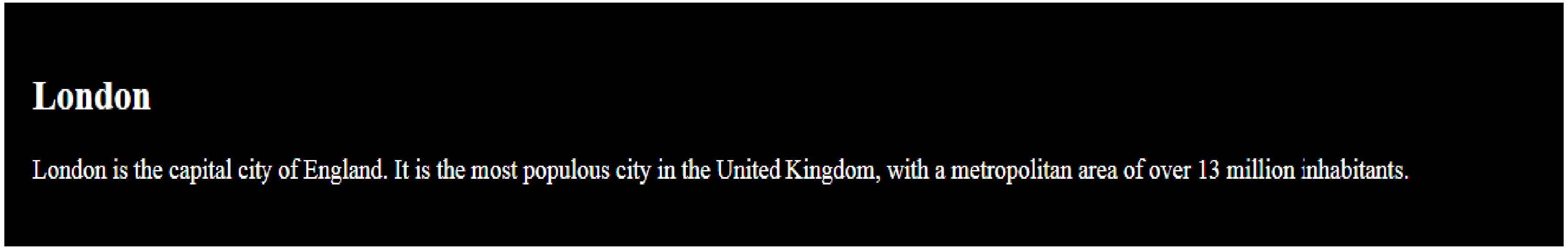
Important HTML Tags: <div>

- Provides additional structure to web page
- Block-level container for organized content
- Often used for:
  - Page headers and footers
  - Menu or navigation bar
  - Photo galleries
  - Ads or outside media

Important HTML Tags: <div>

- When used together with CSS, the <div> element can be used to style blocks of content:

-
  ```
  <div style="background-color:black;color:white;padding:20px;">
    <h2>London</h2>
    <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
  </div>
  ```



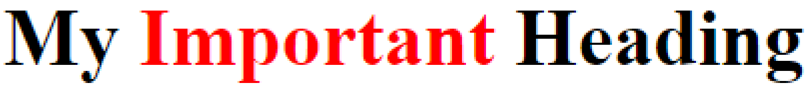
London

London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.

Important HTML Tags: <span>

- Inline container for organized content
- Similar to <div> tag but different in following ways:
  - Block-levels elements (<div>) are designed to contain larger chunks of content designed to stand alone as a unit
  - Always start with a new line

  - Inline elements <span> designed to contain smaller pieces of content,
  - usually within a larger block of content
  - Does not start with a new line



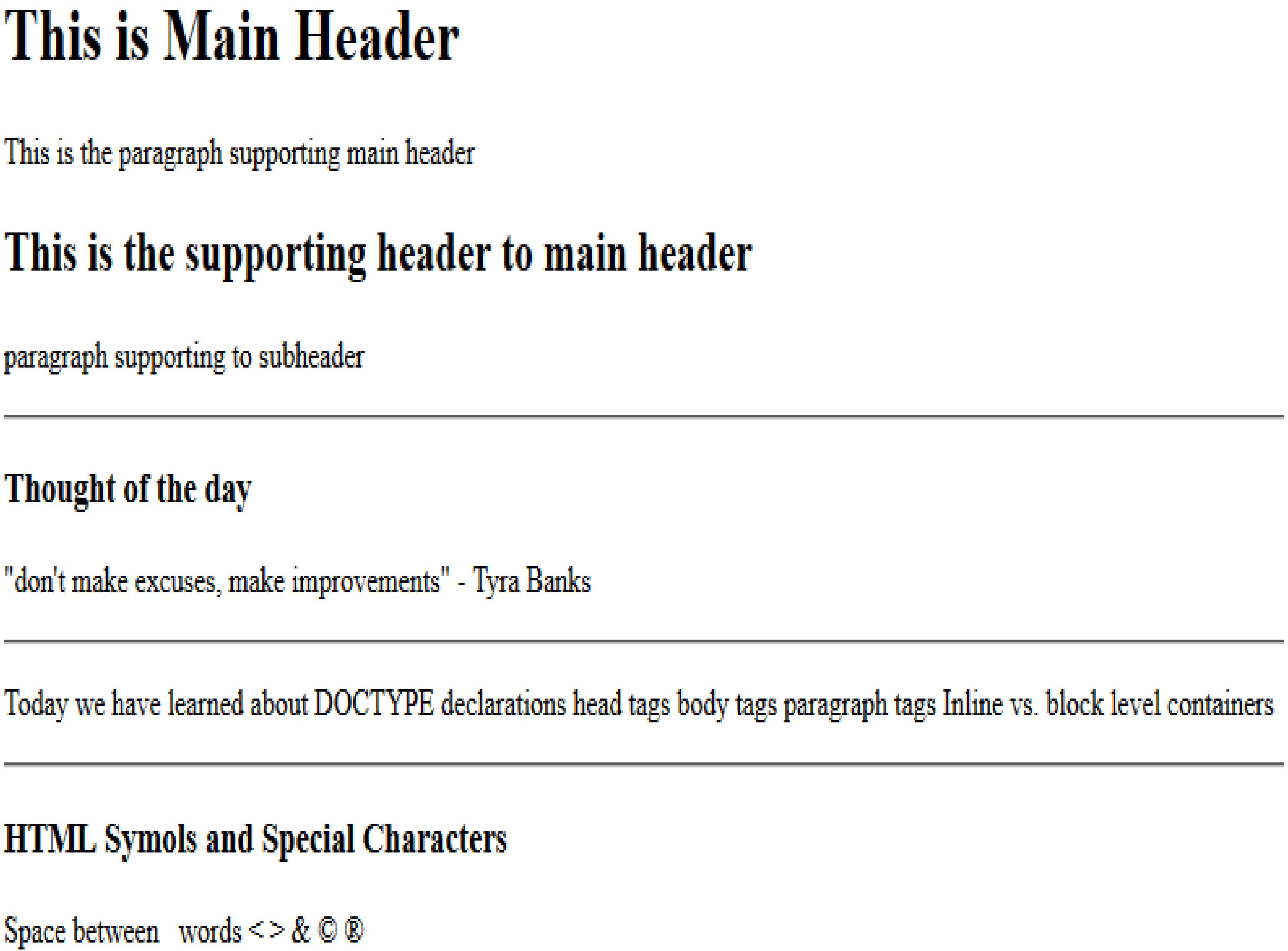
My Important Heading

Important HTML Tags: <!--comments-->

- Text in the HTML that will not be rendered in the browser
- Often used for:
  - Explaining the HTML or leaving notes for other programmers
  - Temporarily removing HTML content
- Syntax:
  - <!--comments-->

HTML symbols and special characters

HTML Entity	Appearance
&nbsp;	Non breaking space; allows for extra white spaces between words
&lt;	<
&gt;	>
&amp;	&
&copy;	©
&reg;	®



This is Main Header

This is the paragraph supporting main header

This is the supporting header to main header

paragraph supporting to subheader

Thought of the day

"don't make excuses, make improvements" - Tyra Banks

Today we have learned about DOCTYPE declarations head tags body tags paragraph tags Inline vs. block level containers

---

HTML Symols and Special Characters

Space between   words < > & © ®

HTML Formatting and Attributes

HTML Attributes

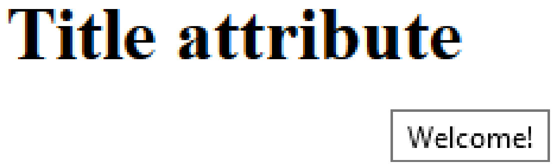
- Purpose of an attribute: provide additional information about a particular HTML element
- Always included within element’s start tag
- Usually comes in name/value pair:
  - name=“value”
  - Name – specifies the property of the element for which additional information is being provided
  - Value – this is selected from set of possible values for given property

```
<p name=“value”>This is a paragraph with defined attribute
</p>
```

Core Attributes: title

- Provides a suggested title for an element
- When user hovers over the element, a “tooltip” will appear at the cursor with suggested title

```
<h1 title=‘Welcome!’> Title attribute </h1>
```



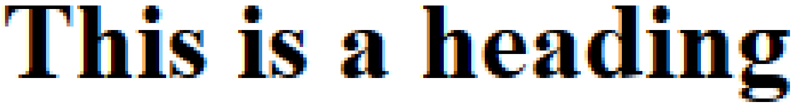
Title attribute

Welcome!

Core Attributes: style

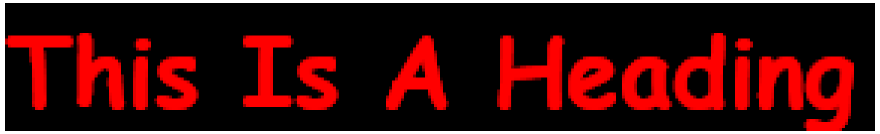
- Used to change the visual presentation of an element
- Value string will be include multiple ***key:value*** pairs (separated by semicolons)

```
<h1> This is a heading</h1>
```



This is a heading

```
<h1 title='Welcome!' style="color: red; text-transform: capitalize;
background-color: black; font-family: cursive;"> This is a heading
</h1>
```

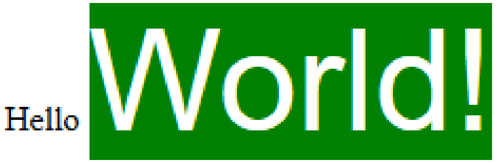


This Is A Heading

Other properties of “style” attribute

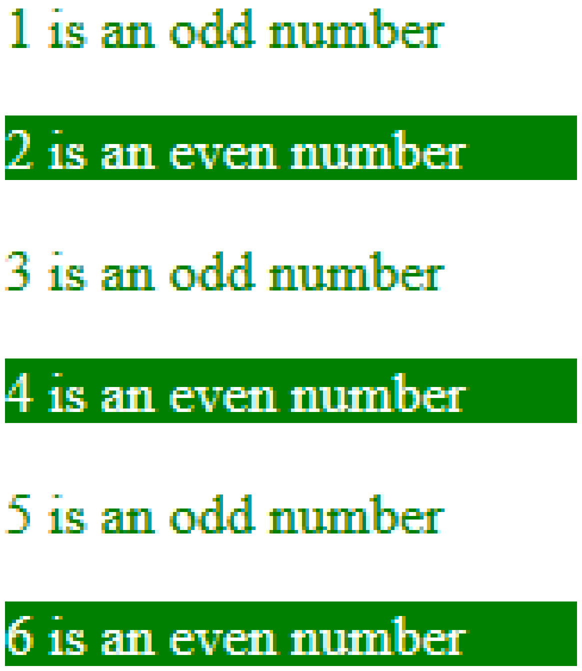
Property	Description	Some Value
background-color	The color that appears behind the text	red, black, yellow, #012345
font-family	Font used to render the text	Verdana, courier, cursive
font-size	Size of text; possibly relative to default size	10px, 12pt, 200%
text-align	Horizontal text alignment	Left, right, center

```
Hello <span style="color: white; background-color: green; font-size:
50pt; font-family: fantasy;" >World!</span>
```



Hello World!

```
<body>
<p style="color: green"> 1 is an odd number </p>
<p style="color: white; background-color: green;">2 is an even
number</p>
<p style="color: green"> 3 is an odd number </p>
<p style="color: white; background-color: green;">4 is an even
number</p>
<p style="color: green"> 5 is an odd number </p>
<p style="color: white; background-color: green;">6 is an even
number</p>
</body>c
```



```
1 is an odd number

2 is an even number

3 is an odd number

4 is an even number

5 is an odd number

6 is an even number
```

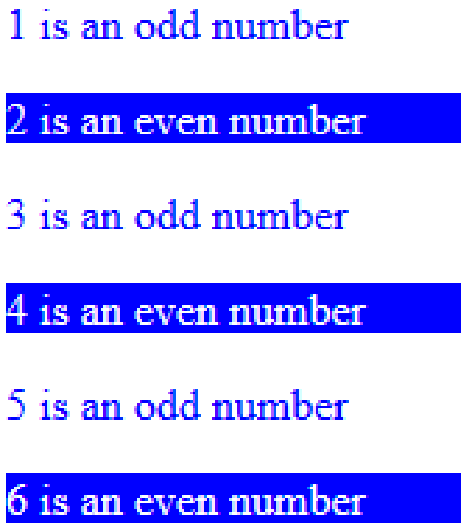
Core Attributes: id and class

- Used to uniquely identify elements within an HTML document

- id:
  - Provide ability to refer to specific element
  - id must be unique
  - Example: header, footer

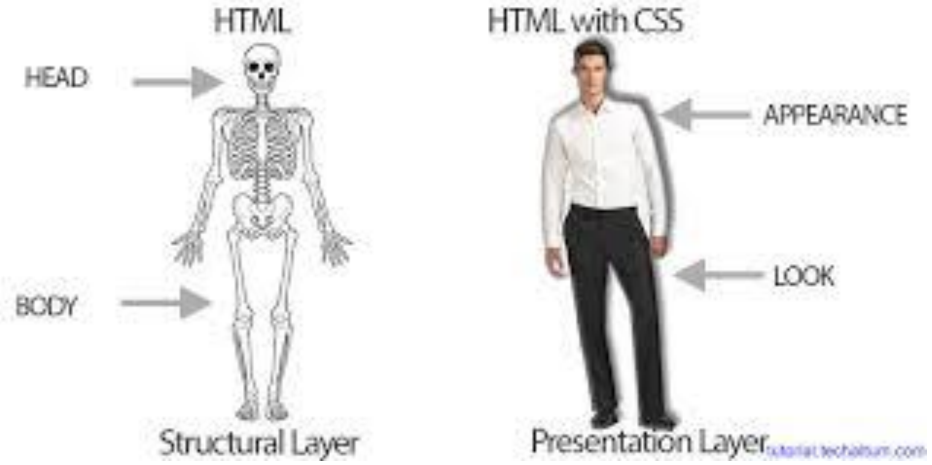
- class:
  - Provide ability to refer subgroups of elements within HTML document
  - Does not have to be unique
  - Example: comment, warning

```
<html>
<head>
        <style type="text/css">
        .odd{
                color: blue;
        }
        .even{
                color: white;
                background-color: blue;
        }
        </style>
</head>
<body>
        <p class="odd"> 1 is an odd number </p>
        <p class="even">2 is an even number</p>
        <p class="odd"> 3 is an odd number </p>
        <p class="even">4 is an even number</p>
        <p class="odd"> 5 is an odd number </p>
        <p class="even">6 is an even number</p>
</body>
</html>
```

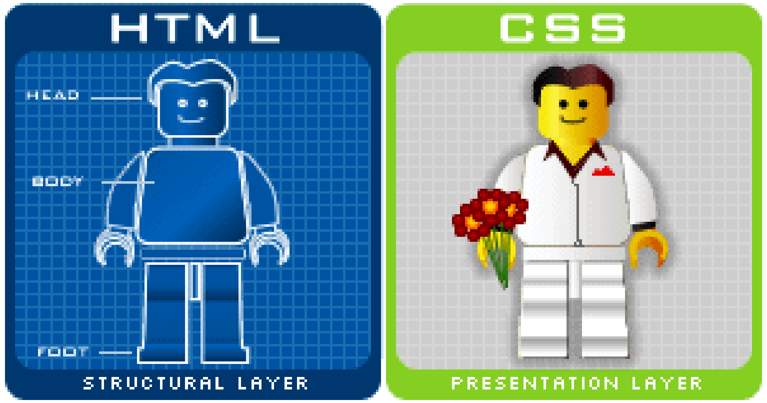


```
1 is an odd number

2 is an even number

3 is an odd number

4 is an even number

5 is an odd number

6 is an even number
```

HTML
Structure
CSS
Style
JavaScript
Behaviour



tutorial.techaltum.com



CSS Syntax

What is CSS?

- Cascading Style Sheet (CSS) is a formatting language used to describe the appearance of content in HTML file
- CSS has standardized specification defined by the World Wide Web Consortium (W3C)

Why CSS?

- HTML ("The Content")
  - What information does the page contain?
  - What is the headings, body, etc.
  - How is the information structure?

- CSS ("The Presentation")
  - What does the page look like?
  - What color, formatting, text size, etc. should the various parts have?

How does CSS work?

1. The web browser receives the HTML page from the server via HTTP
2. HTML page can include CSS either in same file or with link to separate file
   - If it’s separate file, the web browser will request that file separated via HTTP
3. When all HTML and CSS files are available, the browser will render the page

How does CSS work?

4. For each element in the HTML page, the web browser will display the content and use the CSS to style it
5. Ideally, exactly one set of CSS style sheets will apply to any given element
6. If there are conflicting styles defined, complex rules determine which gets applied

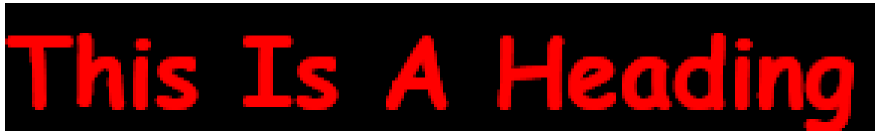
How do we use CSS?

- Inline – use tag’s “style” attribute to specify appearance
- Internal – create <style> elements in HTML and assigns to different tags and classes, etc.
- External – specify styling in a separate CSS file

Inline CSS as “style” attribute

- Include CSS within the element tag element as a “style” attribute

```
<h1 title='Welcome!' style="color: red; text-transform: capitalize;
background-color: black; font-family: cursive;"> This is a heading
</h1>
```



This Is A Heading

Inline CSS as “style” attribute

- Advantages
  - Easy to use
  - Good for quick, one-off situations

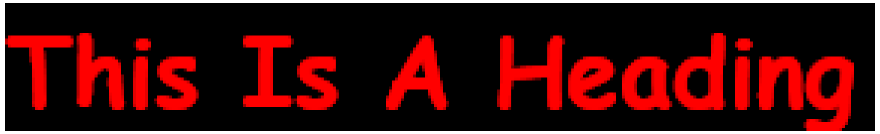
- Disadvantages
  - Mixing content and presentation – this should be avoided
  - Hard to manage for large, complex pages

Internal CSS using <style> tag

- Include CSS within the head of the HTML using <style> elements

```
<head>
<style type="text/css">
h1 {
          color: red;
          text-transform: capitalize;
          background-color: black;
          font-family: cursive;

}
</style>
</head>
<body>
<h1> This is a heading </h1>
</body>
```

CSS Selector - h1


This Is A Heading

property:value

Internal CSS using <style> tag

- Advantages:
  - Separates content and presentation
  - Easy to use
    - If you want to apply the same style to all tags of a certain type (say <p>) , it is easier to do it with internal CSS than inline CSS
    - Good if you only have a limited amount of CSS in the page

- Disadvantages
  - Hard to manage for large and complex pages
  - **Cannot use across multiple pages**

External CSS

- Include CSS in a separate file and link the file in the head o

Example.html

```
<head>
<link rel="stylesheet" type="text/css" href="sample.css">
</head>
<body>
<h1> This is a heading </h1>
</body>
```

Sample.css

```
                        h1 {
                                color: white;
                                text-transform:
capitalize;
                                background-color:
green;
                                font-family: cursive;
                        }
```

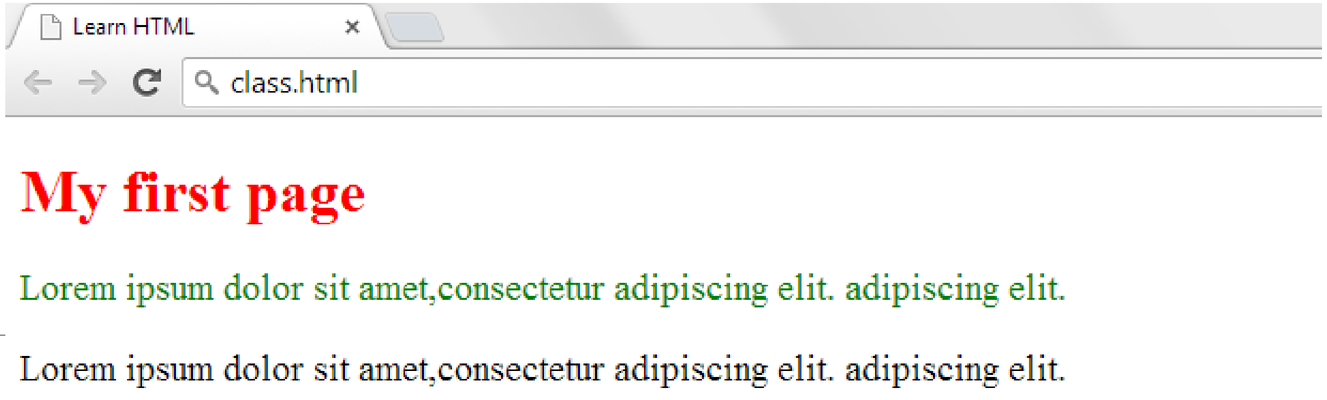
External CSS

- Advantages
  - Separates content and presentation
  - Can include many different CSS pages with multiple <link> tags

- Disadvantages
  - Lots of different files to manage (which might be overkill for smaller pages)

CSS selectors

Type of selector	What’s in the CSS file?	What does this selector apply to?	What does the HTML file contain?
Element selector	h1 {color:red;}	All <h1> elements	<h1> … </h1>
Class selector	.odd {…}	All elements in class “odd”	<div class=“odd”> …. </div>
Id selector	#section1{…}	Unique element with id “section1”	<p id=“section1”> … </p>



My first page

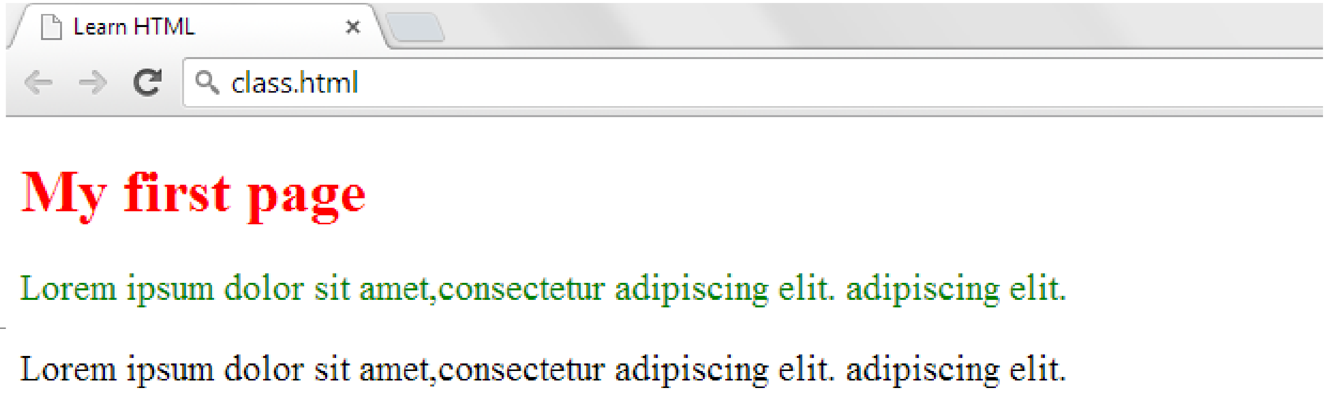
Lorem ipsum dolor sit amet,consectetur adipiscing elit. adipiscing elit.

Lorem ipsum dolor sit amet,consectetur adipiscing elit. adipiscing elit.

Class Selector

```
<h1>My first page</h1>

<p class="intro">Lorem ipsum dolor sit amet,
adipiscing elit. adipiscing elit.</p>

<p>Lorem ipsum dolor sit amet,consectetur
adipiscing elit. adipiscing elit.</p>
```

```
h1{color:red;}
p{font-size:20px;}
.intro{color:green;}
```



My first page

Lorem ipsum dolor sit amet,consectetur adipiscing elit. adipiscing elit.

Lorem ipsum dolor sit amet,consectetur adipiscing elit. adipiscing elit.

Id and Class Selector

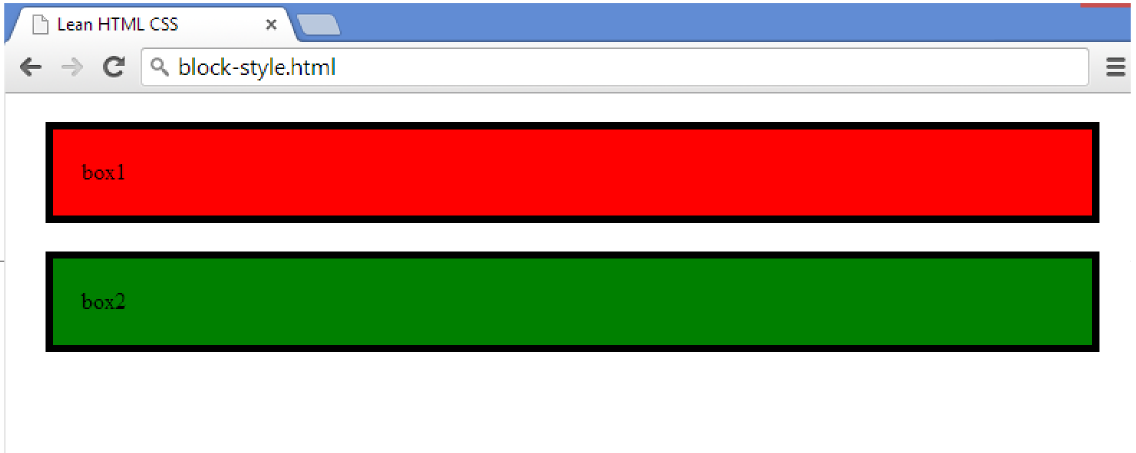
```
<h1>My first page</h1>

<p id="intro" class="intro">Lorem ipsum dolor
adipiscing elit. adipiscing elit.</p>

<p>Lorem ipsum dolor sit amet,consectetur
adipiscing elit. adipiscing elit.</p>
```

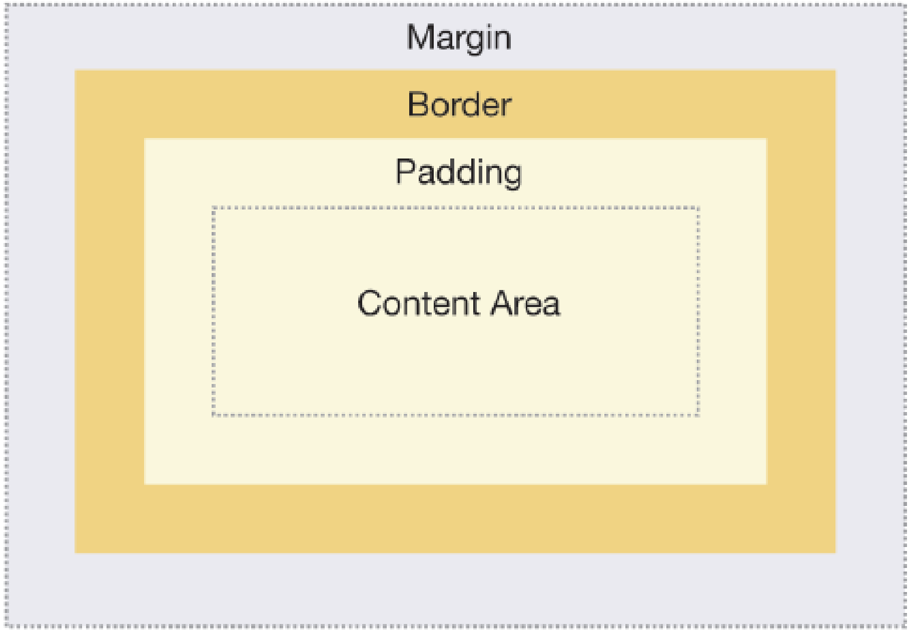
```
h1{color:red;}
p{font-size:20px;}
#intro{color:green;}
.intro{color:blue;}
```

Box Model (Block)

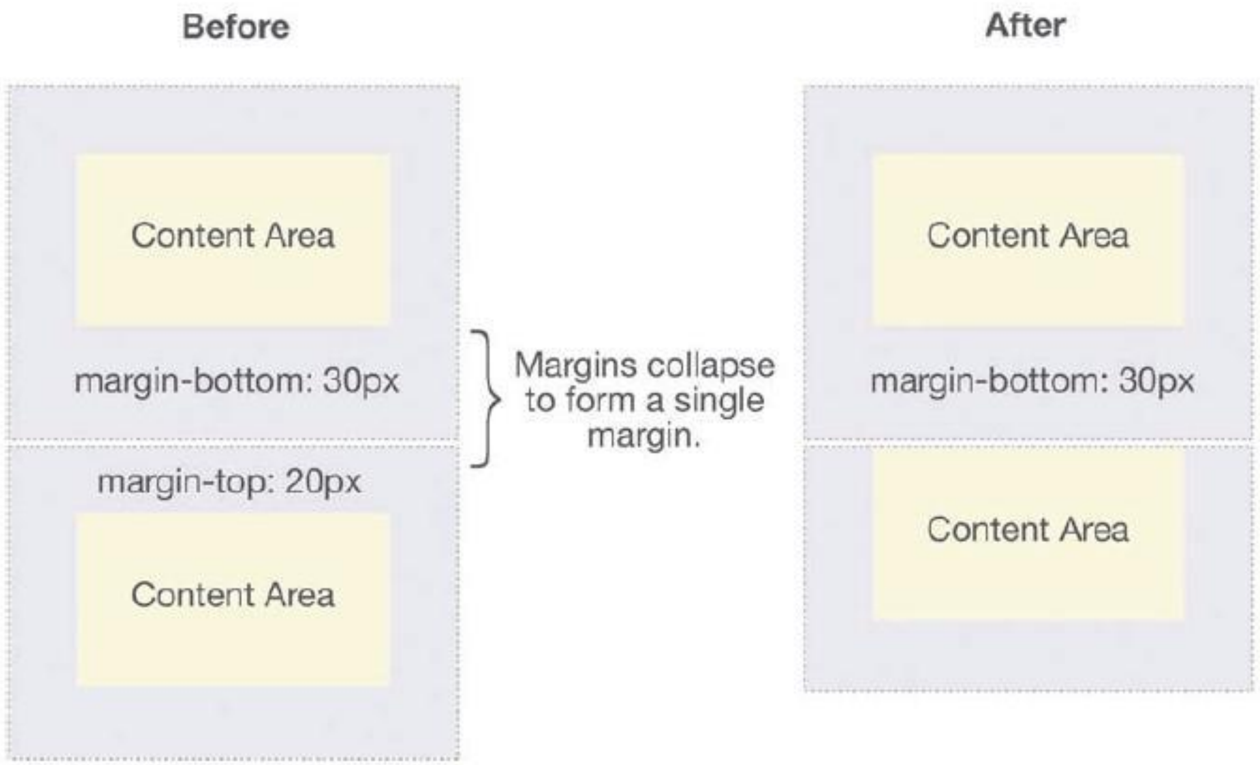
```
<div class="box1">box1</div>

<div class="box2">box2</div>
```

```
.box1{
  background:red;
  padding:20px;
  margin:20px;
  border: 5px solid black;
  }
  .box2{
  background:green;
  padding:20px;
  margin:20px;
  border: 5px solid black;
  }
```

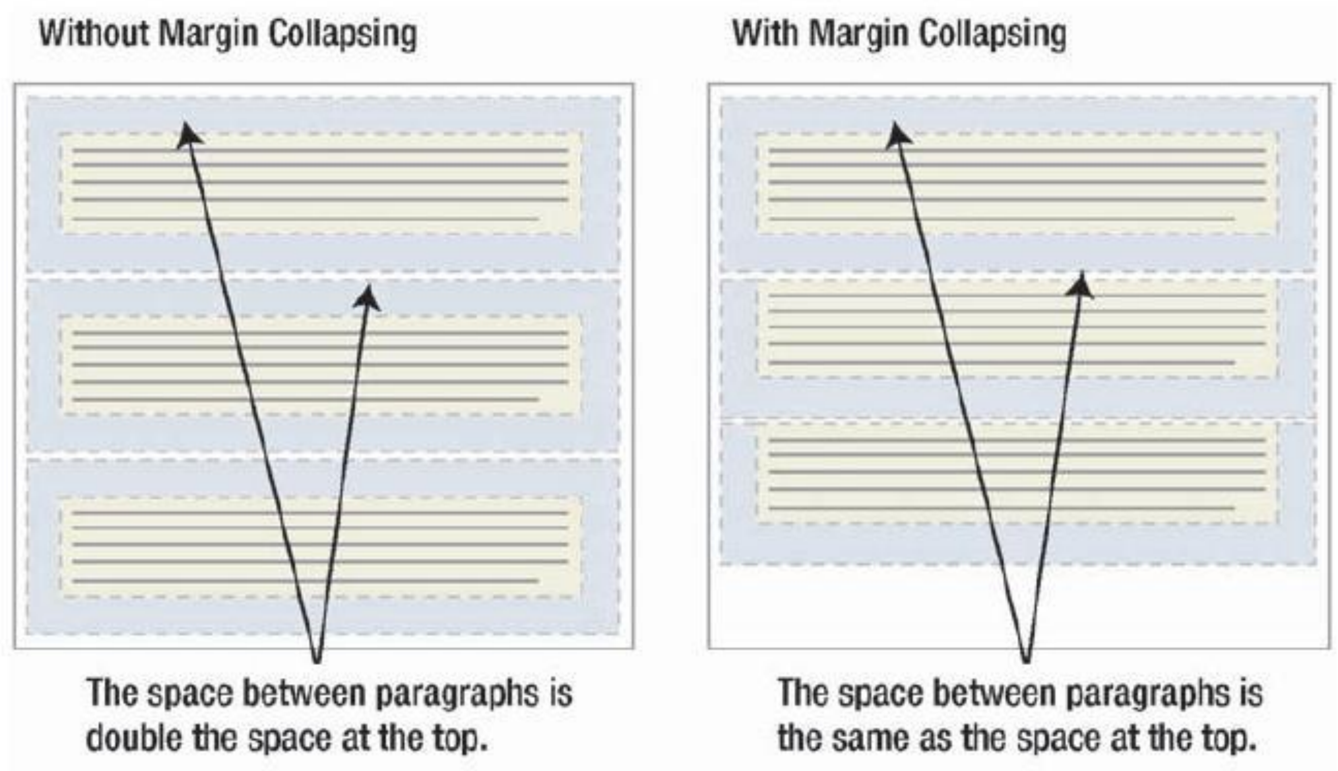




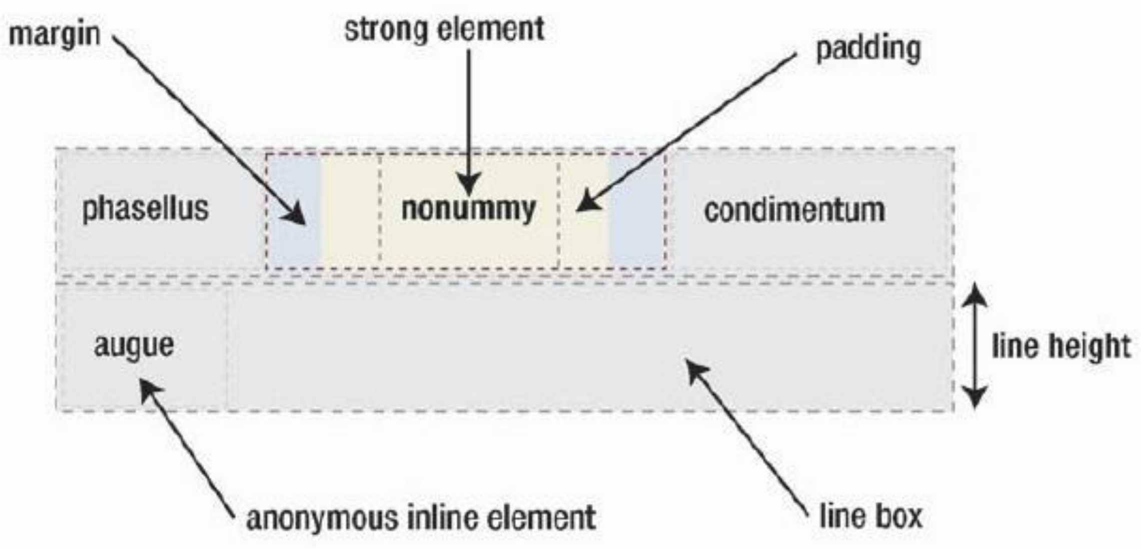
Margin Collapse



Margin Collapse


Without Margin Collapsing
With Margin Collapsing
The space between paragraphs is double the space at the top.
The space between paragraphs is the same as the space at the top.

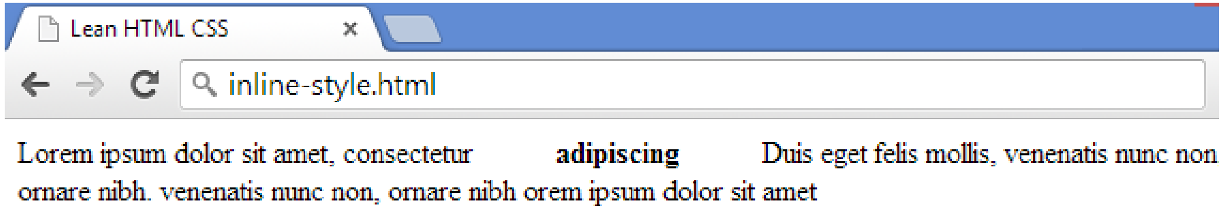
Box Model (Inline)



Box Model (Inline)

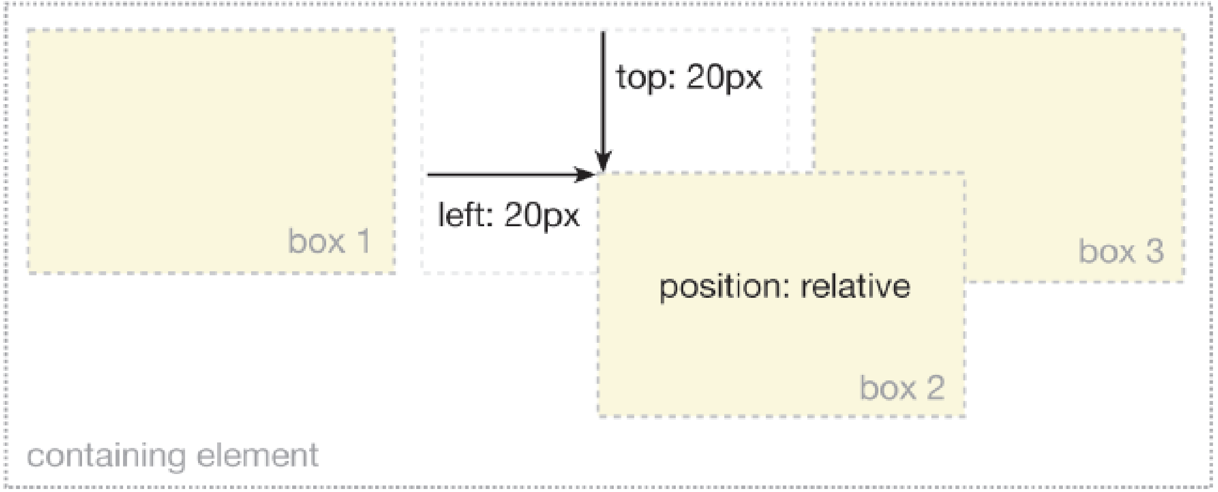
```
<p>Lorem ipsum dolor sit amet,
consectetur <strong>adipiscing</strong>
Duis eget felis mollis, venenatis
nunc non, ornare nibh. venenatis
nunc non, ornare nibh orem ipsum dolor sit
amet </p>
```

```
strong{
   padding:20px;
   margin:20px;
}
```

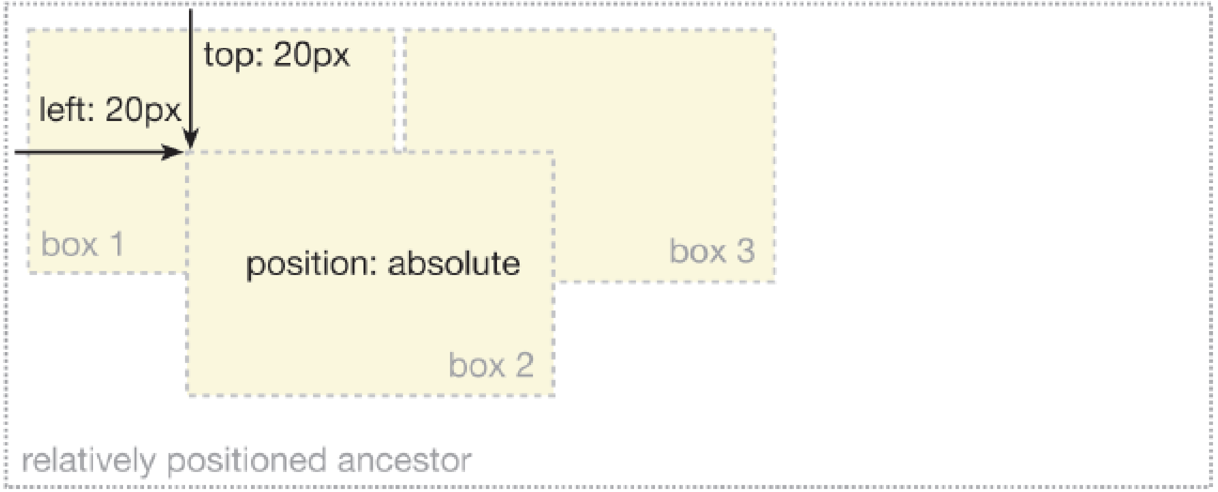


Lorem ipsum dolor sit amet, consectetur **adipiscing** Duis eget felis mollis, venenatis nunc non ornare nibh. venenatis nunc non, ornare nibh orem ipsum dolor sit amet

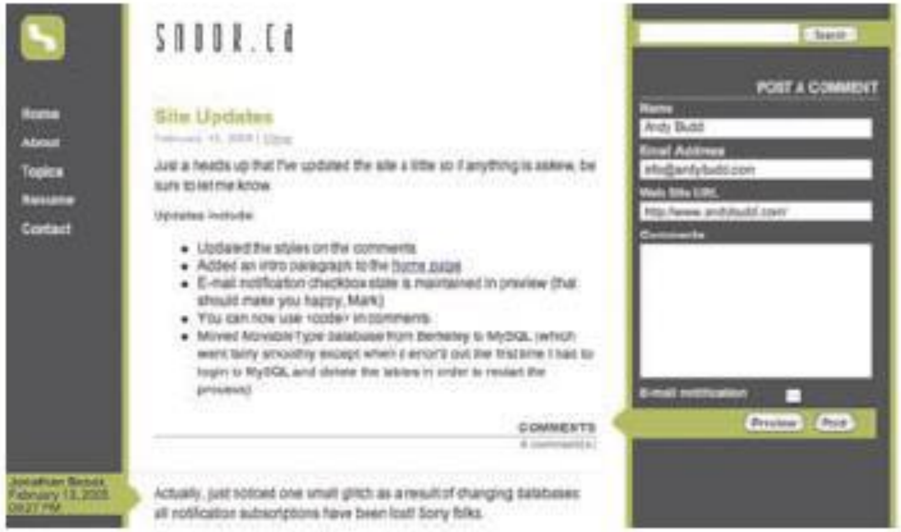
Relative Position



Absolute Position



Fixed Position



SNOOK.CA

Site Updates

February 13, 2005 | Sites

Just a heads up that I've updated the site a little so if anything is askew, be sure to let me know.

Updates include:

- Updated the styles on the comments
- Added an intro paragraph to the home page
- E-mail notification checkbox state is maintained in preview (that should make you happy, Mark)
- You can now use <code> in comments
- Moved MovableType database from Berkeley to MySQL (which went fairly smoothly except when it error'd out the first time I had to login to MySQL and delete the tables in order to restart the process)

COMMENTS
4 comment(s)

Jonathan Snook
February 13, 2005
09:27 PM

Actually, just noticed one small glitch as a result of changing databases: all notification subscriptions have been lost! Sorry folks.

POST A COMMENT

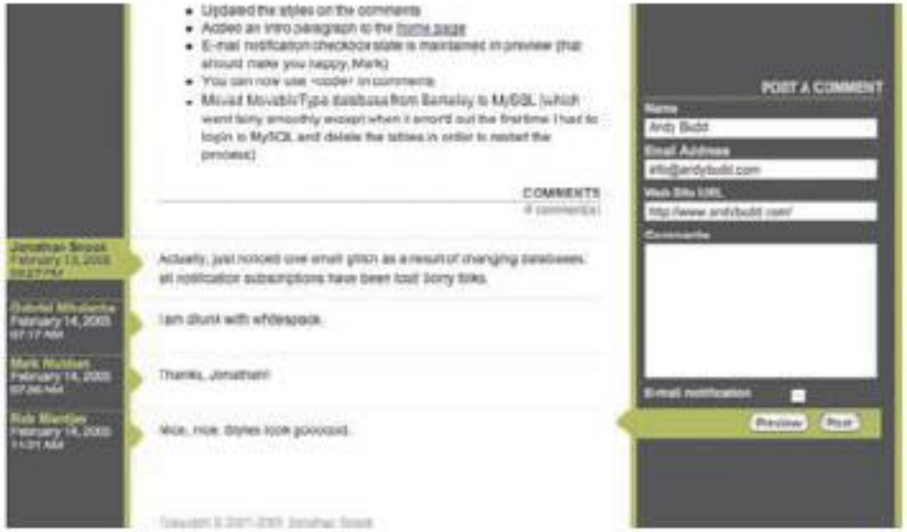
Name: Andy Budd

Email Address: info@andybudd.com

Web Site URL: http://www.andybudd.com/

Comments:

E-mail notification □



- Updated the styles on the comments
- Added an intro paragraph to the home page
- E-mail notification checkbox state is maintained in preview (that should make you happy, Mark)
- You can now use <code> in comments
- Moved MovableType database from Berkeley to MySQL (which went fairly smoothly except when it error'd out the first time I had to login to MySQL and delete the tables in order to restart the process)

COMMENTS
4 comment(s)

Jonathan Snook
February 13, 2005
09:27 PM

Actually, just noticed one small glitch as a result of changing databases: all notification subscriptions have been lost! Sorry folks.

Gabriel Mihalache
February 14, 2005
07:17 AM

I am drunk with whitespace.

Mark Wubben
February 14, 2005
07:26 AM

Thanks, Jonathan!

Rob Mientjes
February 14, 2005
11:31 AM

Nice, nice. Styles look goooood.

Copyright © 2001-2005 Jonathan Snook

POST A COMMENT

Name: Andy Budd

Email Address: info@andybudd.com

Web Site URL: http://www.andybudd.com/

Comments:

E-mail notification □

HTML Lists and Forms

Lists in HTML : <ol> and <ul>

- Lists can be either ordered or unordered

Ordered list: <ol>

1. January
2. February
3. March
4. April

```
<ol>
        <li>January</li>
        <li>February</li>
        <li>March</li>
        <li>April</li>
</ol>
```

Unordered list: <ul>

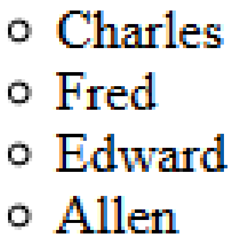
- Charles
- Fred
- Edward
- Allen

```
<ul>
        <li>Charles</li>
        <li>Fred</li>
        <li>Edward</li>
        <li>Allen</li>
</ul>
```

Unordered Lists - style

- Use CSS style properties to determine the style of bullet point used within the lists as follows (inline CSS example)

```
<ul style="list-style: circle;" >
            <li>Charles</li>
            <li>Fred</li>
            <li>Edward</li>
            <li>Allen</li>
</ul>
```



○ Charles
○ Fred
○ Edward
○ Allen

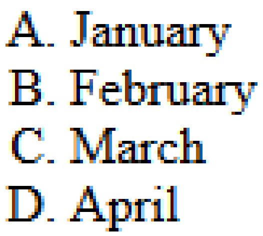
List-style-type : value	Bullet view
disc •	Disc is the default value
circle ○	This is the circle view
square ■	This is the square view
none	No bullets, just list items

Ordered Lists – style

- Use the type attribute of <ol> tag to select numbering of each item

```
<ol type='A'>
    <li>January</li>
    <li>February</li>
     <li>March</li>
     <li>April</li>
</ol>
```

Type	View
type = "1"	1. This is the default setting 2. For ordered lists
type = "A"	A. Items ordered with B. Uppercase letters
type = "a"	a. Items ordered with b. Lowercase letters
type = "I"	I. Items ordered with II. Uppercase Roman Numerals
type = "i"	i. Items ordered with ii. Lowercase Roman Numerals

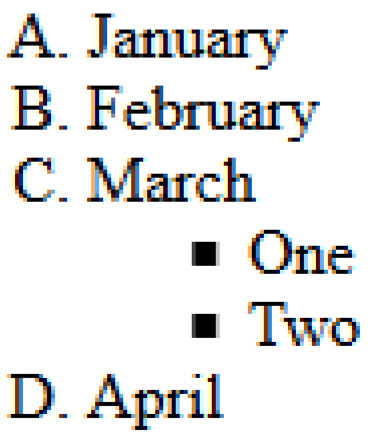


A. January
B. February
C. March
D. April

Nested Lists

- List items can themselves include lists as well to produce a nested list effect

```
<ol type=‘A’>
    <li>January</li>
    <li>February</li>
     <li>March</li>
          <ul style="list-style: square;">
                 <li>One</li>
                 <li>Two</li>
             </ul>
     <li>April</li>
</ol>
```



A. January
B. February
C. March
    ■ One
    ■ Two
D. April

HTML Links, Images, and Tables

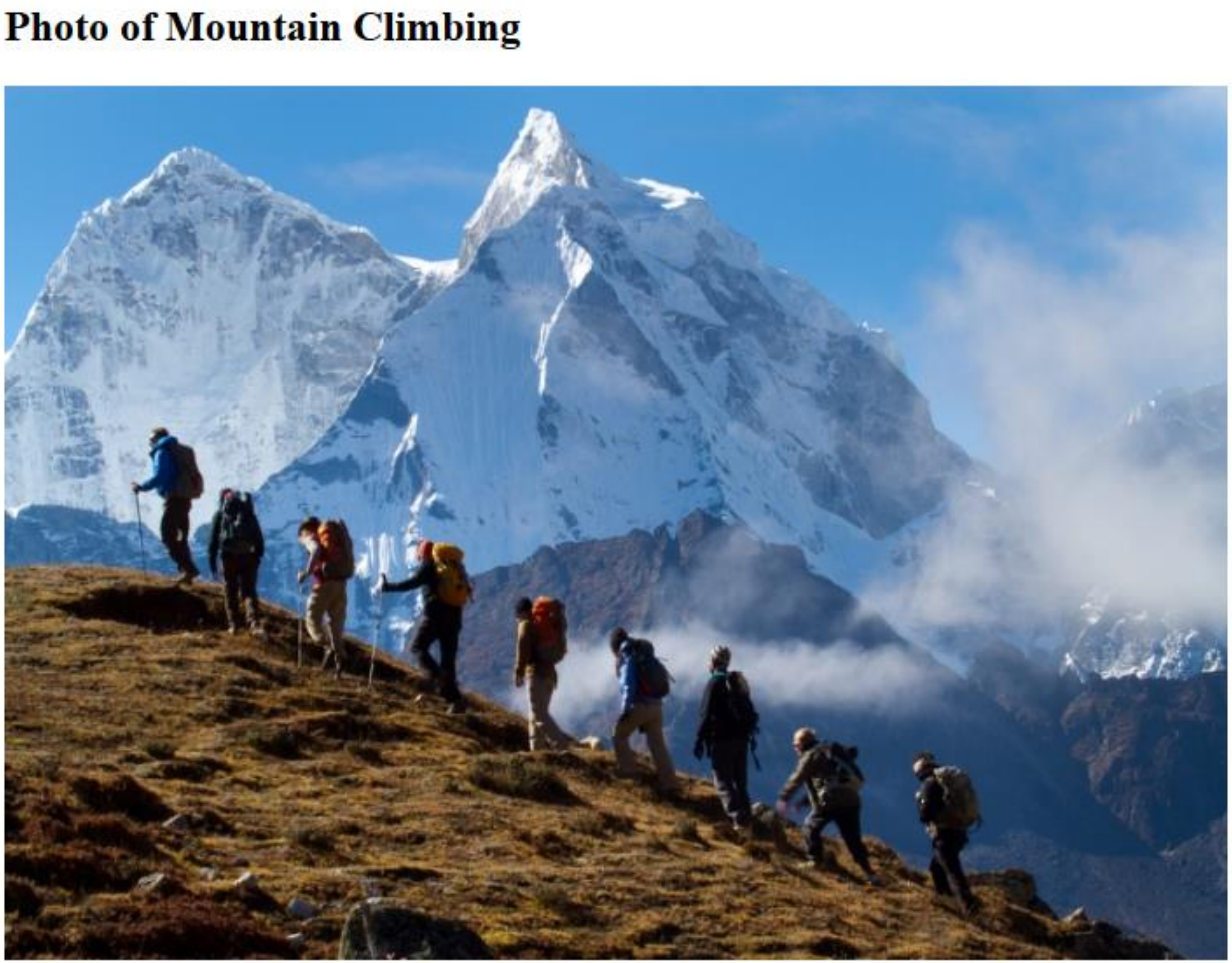


Photo of Mountain Climbing

Images in HTML : <img>

- <img> tag includes an image in HTML page
- Required attributes:
  - src-specifies the link to the image to be included. Can be a relative or absolute path
  - alt – text to include as image description in case image does not load, or additional information is needed
- Other attributes:
  - height, width – specified in pixels (e.g. 50px)
  - title – tooltip text displayed when mouse hovers over image

Absolute Path vs. Relative Path

- <img src=“path/to/image”>
- Absolute Path – URL that can be accessed if provided to browser on its own

```
<img src="https://mtnweekly.com/wp-content/uploads/2014/06/Mountain-Climbing-734x489.jpg" height="500px"  width="700px" alt="Mountain Climbing Image">
```

- Relative Path – path to a file included within local file system, relative to the HTML page

```
<img src="mountain_climb.jpg" height="500px"  width="700px"
alt="Mountain Climbing Image">
```

```
<img src=“photos/mountain_climb.jpg" height="500px"  width="700px"
alt="Mountain Climbing Image">
```

Links in HTML: <a>

- <a> tag used to include a link to another page/resource on Web
- Associated attributes:
  - href (required) – specifies the location to which to navigate if link is clicked
  - target – specifies where the link should be opened when clicked
    - target="_self" will open web page in same tab of browser - default
    - targe="_blank" will open web page in new tab of browser
- Similar to images, value of href attribute can either be a relative or absolute link
- Text of element will be displayed as clickable link

```
<a href="https://en.wikipedia.org/wiki/Main_Page" target="_blank">Wikipedia</a>
```

Creating Bookmarks using Relative URLs

- Define an id attribute for the element for which you would like to create a bookmark

```
<h3 id="social_network">Social Networking Websites</h3>
```

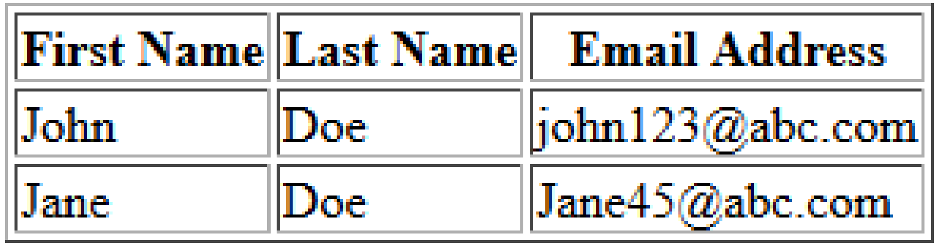
- Create a link to serve as a bookmark
  - From within same page:

```
<a href="#social_network">Jump to Social Network Section</a>
```

  - From a different page:

```
<a href=“index.html#social_network">Jump to Social Network
Section</a>
```

Tables



First Name	Last Name	Email Address
John	Doe	john123@abc.com
Jane	Doe	Jane45@abc.com

```
<table border="1">
<tr>
      <th>First Name</th>
      <th>Last Name</th>
      <th>Email Address</th>

</tr>
<tr>
       <td>John</td>
       <td>Doe</td>
       <td>john123@abc.com</td>
</tr>
<tr>
      <td>Jane</td>
     <td>Doe</td>
    <td>Jane45@abc.com</td>
</tr>
</table>
```

•<table> - tag is used to define and contain a table

•<tr> - tag defines a table row
•<th> - tag defines table header (first row, titles)
•<td> - tag defines table cell (“table data”)

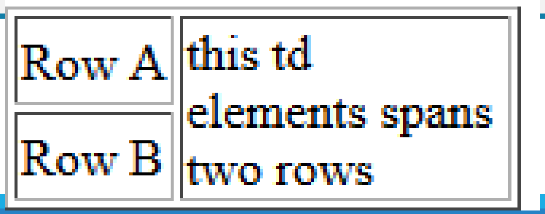
Table CSS Properties

Property	Description	Sample Values
border	Draws border around table, rows, or cell Default is no border	1px solid black
padding	Space between cell content and its borders	15px
text-align	Horizontal text alignment	center, left, right
border-spacing	Space between cells	2px

Merging Cells

- Use **rowspan** attribute to span multiple rows (merge vertically)

```
<table border="1px solid black">
<tr>
     <td>Row A</td>
<td rowspan="2" width="100px">this td elements spans two rows</td>
</tr>
<tr>
      <td>Row B</td>
</tr>
</table>
```

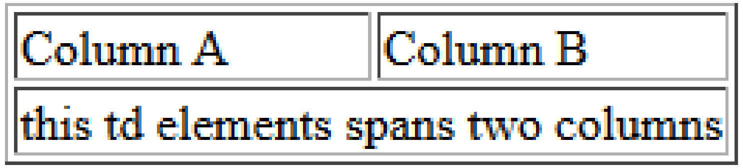


Row A	this td elements spans two rows
Row B	

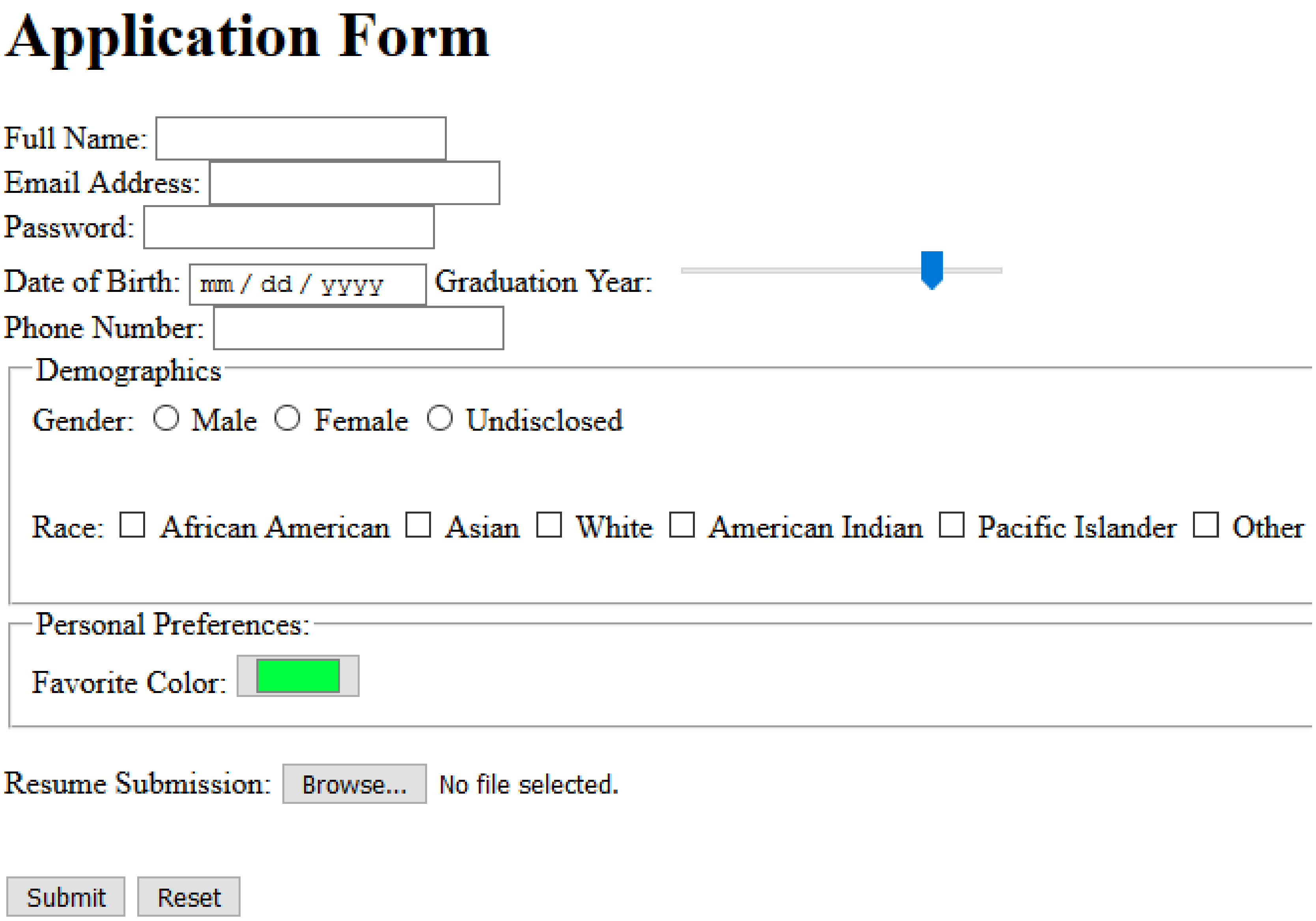
Merging Cells

- Use **colspan** attribute to span multiple columns (merge horizontally)

```
<table border="1px solid black">
<tr>
            <td>Column A</td>
            <td>Column B</td>
</tr>
<tr>
        <td colspan="2">this td elements spans two columns</td>
</tr>
</table>
```



Column A	Column B
this td elements spans two columns	



Application Form

Full Name:

Email Address:

Password:

Date of Birth: mm / dd / yyyy Graduation Year:

Phone Number:

Demographics

Gender: ○ Male ○ Female ○ Undisclosed

Race: □ African American □ Asian □ White □ American Indian □ Pacific Islander □ Other

Personal Preferences:

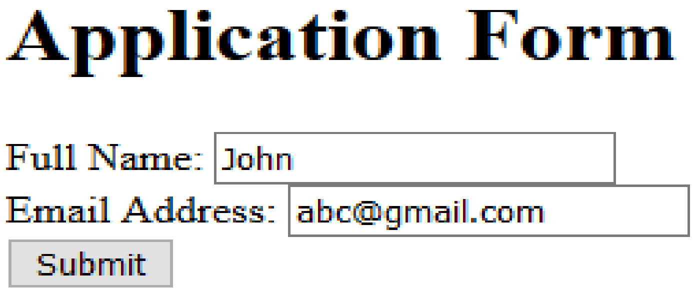
Favorite Color: 🟩

Resume Submission: Browse... No file selected.

Forms

- Forms are used to retrieve information from user of a web page
- Enclose all form fields within <form> element

```
<form>
  Full Name: <input type="text"
name="username">
<br>
 Email Address: <input type="email"
name="email">
<br>
<input type="submit" value="Submit">
</form>
```



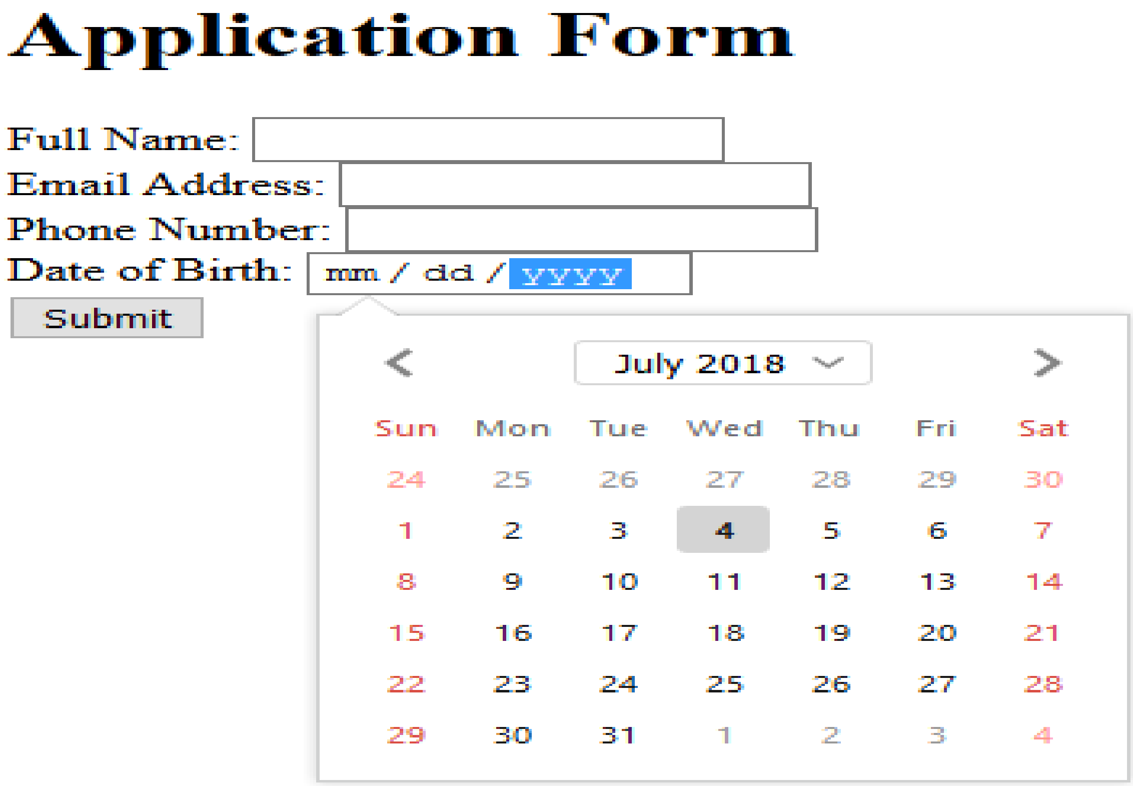
Application Form

Full Name: John

Email Address: abc@gmail.com

Form: <input type=“value”>

- Use the type attribute of the input tag to provide more information on what type of data to expect from the user
- Different types often allow different input formats that make data entry more convenient for the user



Application Form

Full Name:

Email Address:

Phone Number:

Date of Birth: mm / dd / yyyy

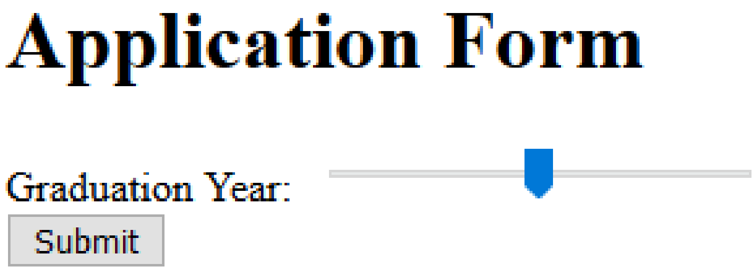
Submit

July 2018

Sun	Mon	Tue	Wed	Thu	Fri	Sat
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

```
<form>
  Full Name: <input type="text" name="username">  <br>
  Email Address: <input type="email" name="email">  <br>
  Phone Number: <input type="tel" name="phoneNum">  <br>
  Date of Birth: <input type="date" name="dob"> <br>
<input type="submit" value="Submit">
</form>
```

Form – Slider/Range input

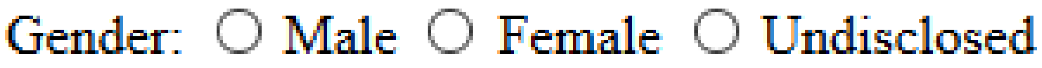


Application Form

Graduation Year:

```
<form>
Graduation Year: <input type="range" name="gradYr" min="1950"
max="2030"> <br>
<input type="submit" value="Submit">
</form>
```

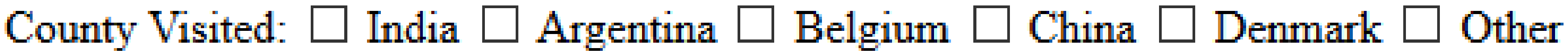
Form – Radio Button Input



Gender: ○ Male ○ Female ○ Undisclosed

```
<form>
    Gender:
   <input type="radio" name="genderOption" value="male"> Male
   <input type="radio" name="genderOption" value="female"> Female
   <input type="radio" name="genderOption" value="undisclosed">
Undisclosed
</form>
```

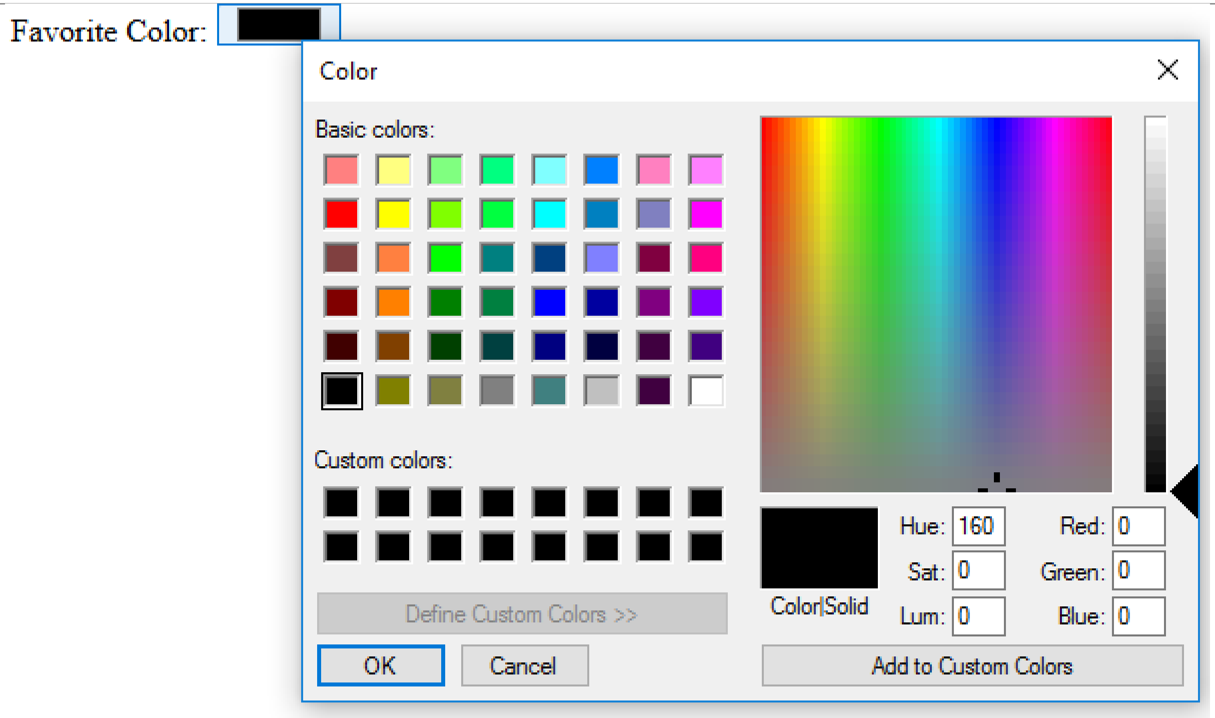
Form – Checkbox Input



County Visited: □ India □ Argentina □ Belgium □ China □ Denmark □ Other

```
<form>
County Visited:
   <input type="checkbox" name="raceOption" value="India"> India
   <input type="checkbox" name="raceOption" value="Argentina"> Argentina
   <input type="checkbox" name="raceOption" value="Belgium"> Belgium
   <input type="checkbox" name="raceOption" value="China"> China
   <input type="checkbox" name="raceOption" value="Denmark"> Denmark
   <input type="checkbox" name="raceOption" value="other"> Other
</form>
```

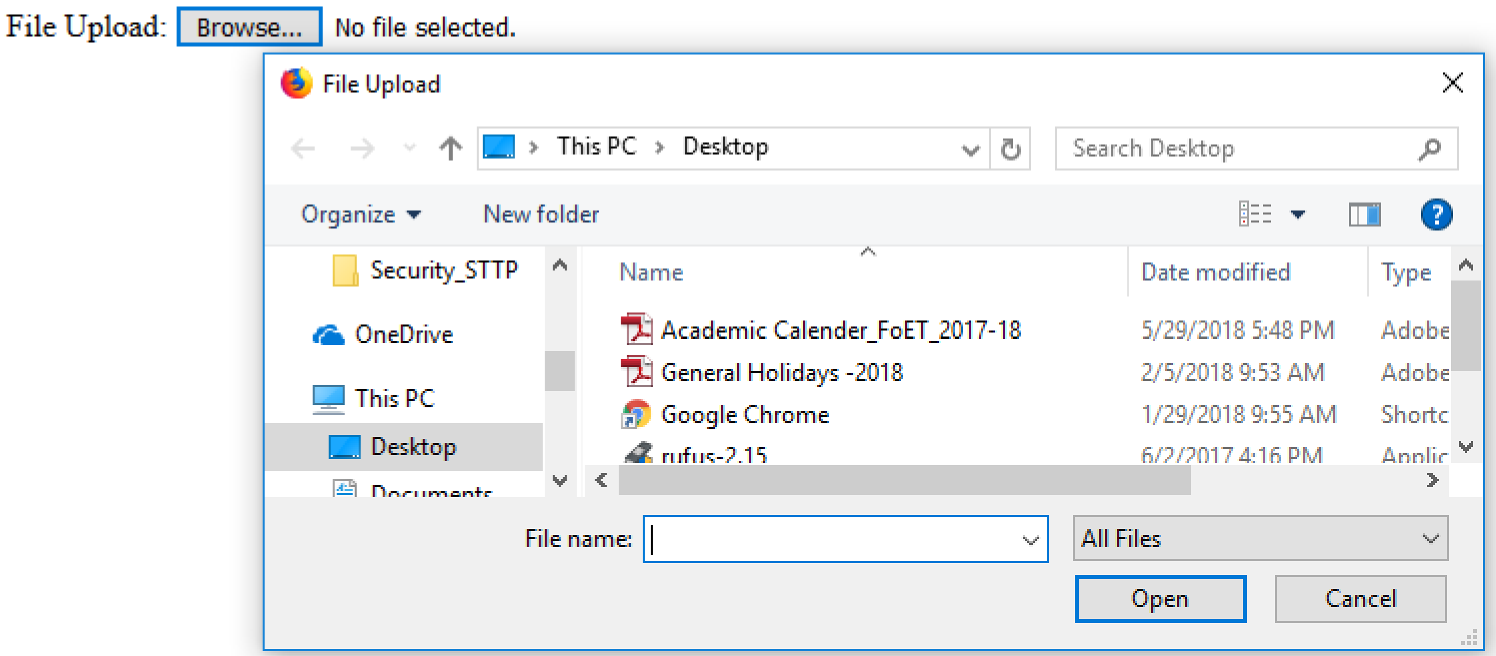
Form – color chooser



Favorite Color:

```
<form>
            Favorite Color:  <input type="color" name="favcolor">
</form>
```

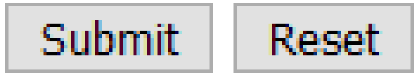
Form – File Upload



Name	Date modified	Type
Academic Calender_FoET_2017-18	5/29/2018 5:48 PM	Adobe
General Holidays -2018	2/5/2018 9:53 AM	Adobe
Google Chrome	1/29/2018 9:55 AM	Shortc
rufus-2.15	6/2/2017 4:16 PM	Applic

```
<form>
        File Upload:   <input type="file" name="fileupload">
</form>
```

Form – Buttons Input



```
<form>
            <input type="submit" value="Submit">
             <input type="reset" value="Reset">
</form>
```

Forms – Input Types

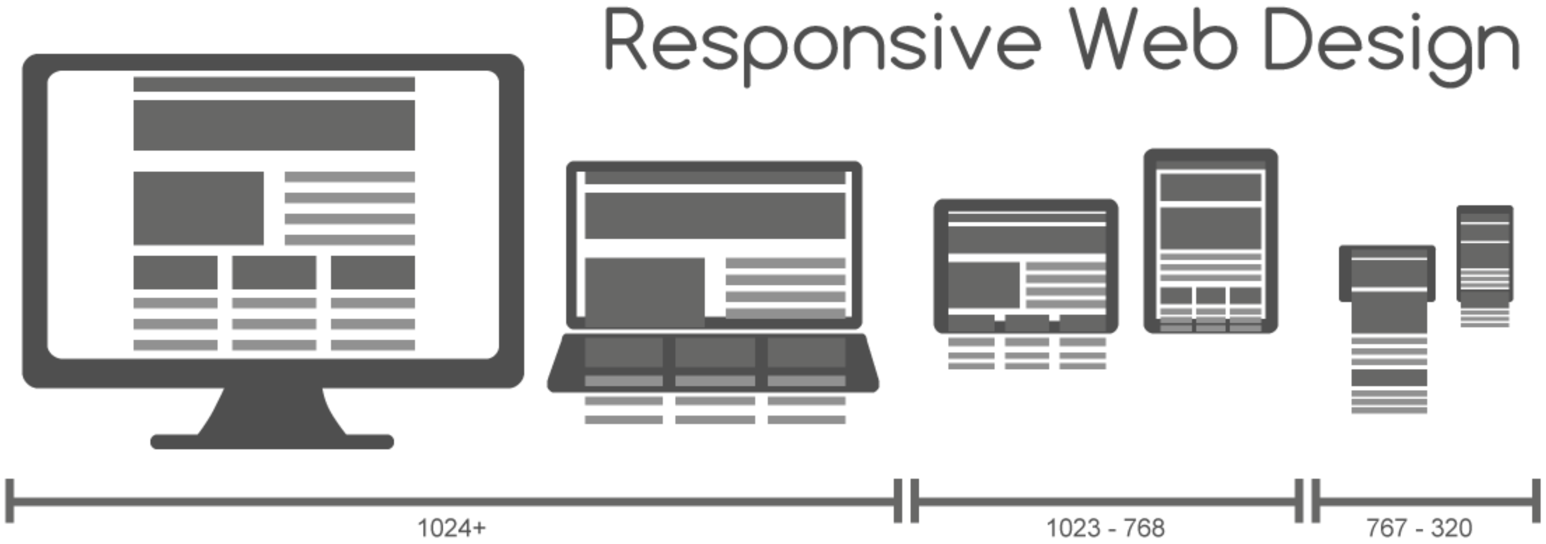
Type	Form Field
button	Clickable button
checkbox	Checkbox options
color	Color picker
date	Date picker (Year, month, day)
email	Email address field
file	File select/browse for file uploading
hidden	Hidden input field
image	Allow image to serve as submit button
month	Month and year control
time	Time control

Forms – Input Types

Type	Form Field
number	Number format entry
password	Masked characters for password entry
radio	Radio button options
range	Slider control to enter number as a range
reset	Reset all form values to default values (button)
search	Text field for searching
submit	Submit button
tel	Phone number input
text	Plaintext field
url	Field for URL

Responsive Web Design with Bootstrap

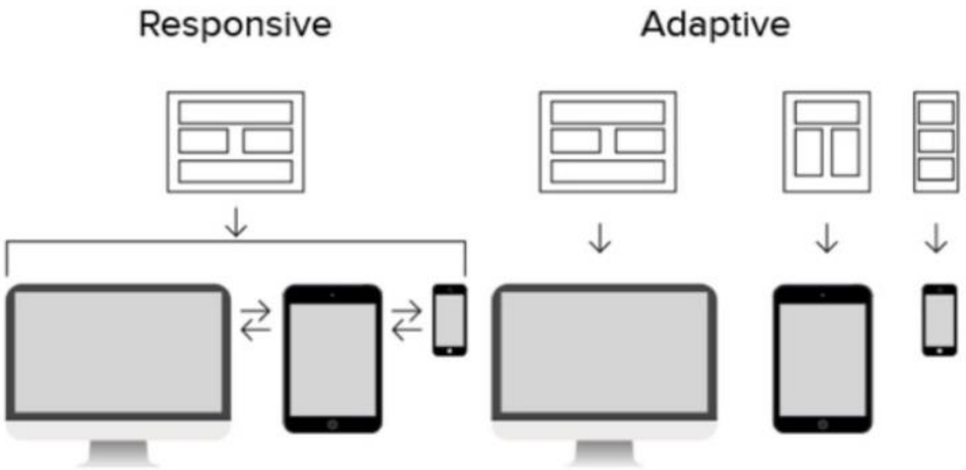
Responsive Web Design



Responsive Web Design

- Responsive Web Design (RWD): approach to designing web pages in a way that takes all possible devices into account

- Approach must be visually flexible to adapt to all potential screen sizes



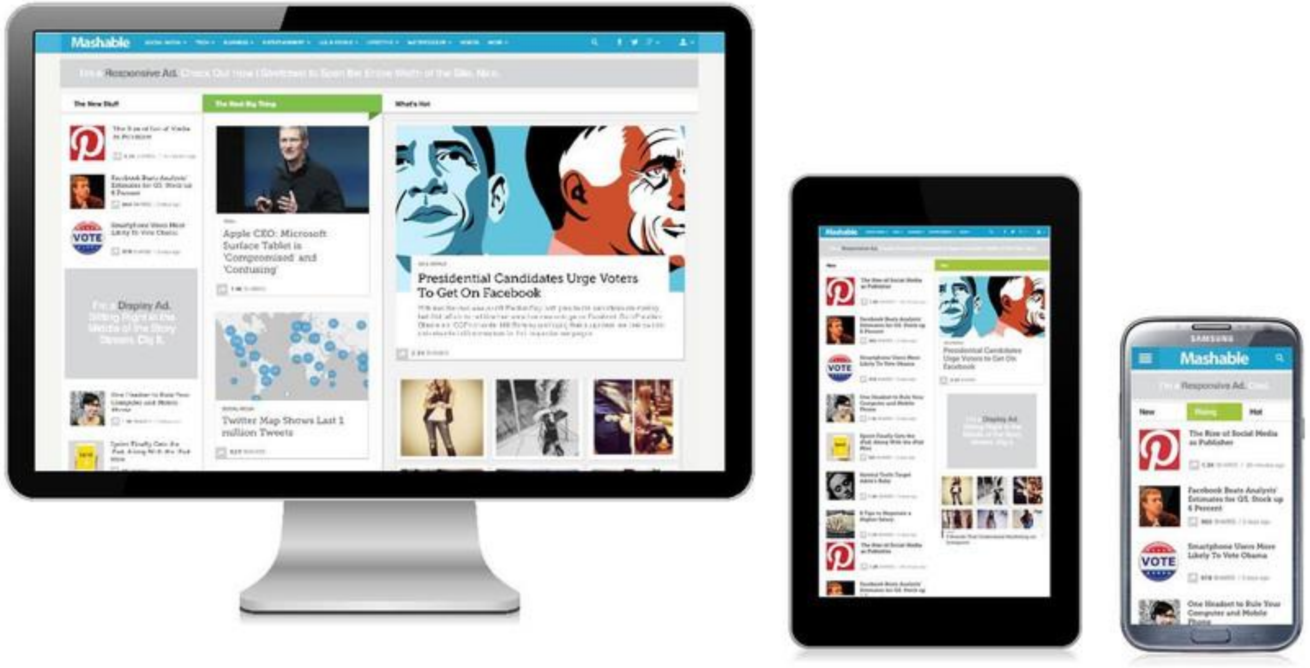
Responsive Web Design

CMPNY.
Link 1 Link 2 Link 3 Link 4
Welcome to our awesome website

Responsive Web Design

home | menus | news + events | groups | gallery | about us
table
FOOD · DRINK
+ quality ingredients
+ traditional methods
+ contemporary interpretation
news + events
Uncorked
ConstrucTABLE
Introductions
news + events
discover a new dining experience in canmore, alberta
butchery and preserving

Responsive Web Design


Mashable
Responsive Ad.
The New Stuff
The Next Big Thing
What's Hot
Apple CEO: Microsoft Surface Tablet is 'Compromised' and 'Confusing'
Presidential Candidates Urge Voters To Get On Facebook
Display Ad.
Twitter Map Shows Last 1 million Tweets
VOTE
SAMSUNG
Mashable
Responsive Ad.
New
Rising
Hot
The Rise of Social Media as Publisher
Facebook Beats Analysts' Estimates for Q3, Stock up 6 Percent
Smartphone Users More Likely To Vote Obama
One Headset to Rule Your Computer and Mobile Phone

Responsive Web Design

www.lawfirmghostwriter.com
LAW FIRM GHOST WRITER
Toll Free: 1.888.901.3939
Services
News & Notes
About
Contact
The Leader in Law Firm Internet Marketing
WEBSITE CONTENT & LEGAL BLOG WRITERS
INTERNET LEGAL MARKETING
SOCIAL MEDIA MARKETIN
LEGAL BLOG TOPICS
LEGAL MARKETING NEWS
AFFILIATES
Internet Legal Marketing
Of Writing Legal Blogs, Court Cases & Pedestrian Safety Tips

Achieving RWD - Challenges

- How should we go about achieving RWD?
- Providing CSS for each platform is challenging as new devices become available
- Manual CSS implementation would take quite a bit of time and effort.
- `<meta name="viewport" content="width=device-width, initial-scale=1">`

Solution – automatic RWD

- **Bootstrap** – open source front-end development framework produced and maintained by Twitter that aids in producing clean, responsive web pages and applications
  - “mobile-first” CSS framework
  - CSS with predefined tags for developers’ use
  - Include UI components, layouts, and other tools
  - developing responsive, mobile first projects on the web.

Bootstrap Grid System

- Grid System: Bootstrap’s solution to making the most of the space provided on a given platform
  - Includes between 1 and 12 columns depending on scale of device viewport
  - Utilizes predefined classes to maximize web page layout options

Using Bootstrap

1. Begin with Basic HTML page

```
<!DOCTYPE html>
<html>
<head>
        <title>Bootstrap Example</title>
</head>
<body>
        Content goes here…
</body>
</html>
```

Using Bootstrap

1. Include Bootstrap in to your project
   1. Include the following within the <head> section
   2. See https://getbootstrap.com/docs/3.3/getting-started/ for most recent versions

```
<head>
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet”
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<!-- jQuery library -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

<!-- Latest compiled JavaScript -->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

</head>
```

Using Bootstrap

- Additional details: utilize “mobile-first” framework (Add within <head> section above other tags)

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

- Add content in the <body>
  - Create a <div> using the “container” class

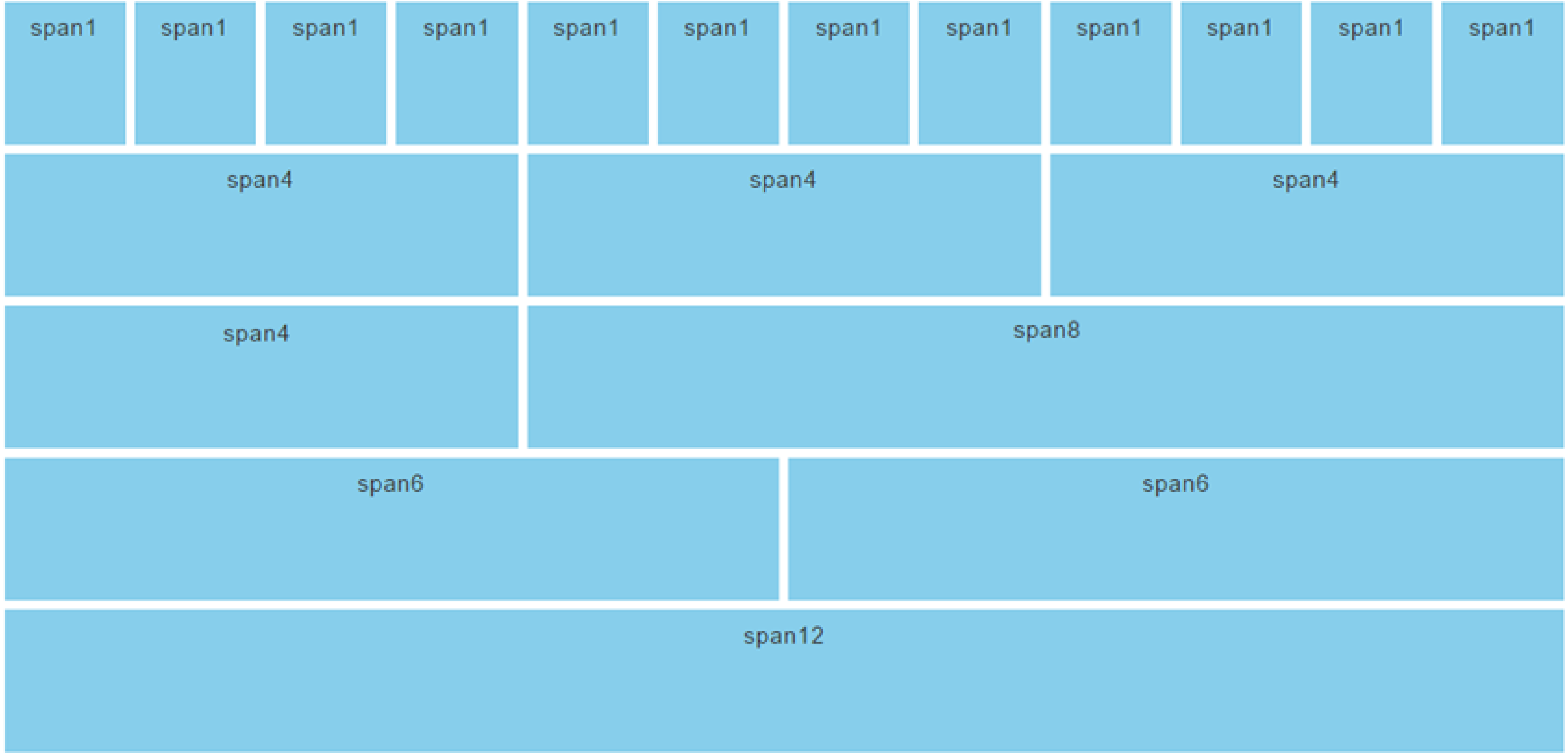
Bootstrap

- Once you have linked to a bootstrap.min.css file, you get a whole bunch of classes set up for you, for free!

1. The “container” class is essential. Everything needs to be inside of it.
2. The “row” class is almost as important. Use it to specify groupings of columns.
3. Finally, there are a number of classes named with the pattern col-?-?

```
<div class="col-md-12">Some Content here.</div>
```

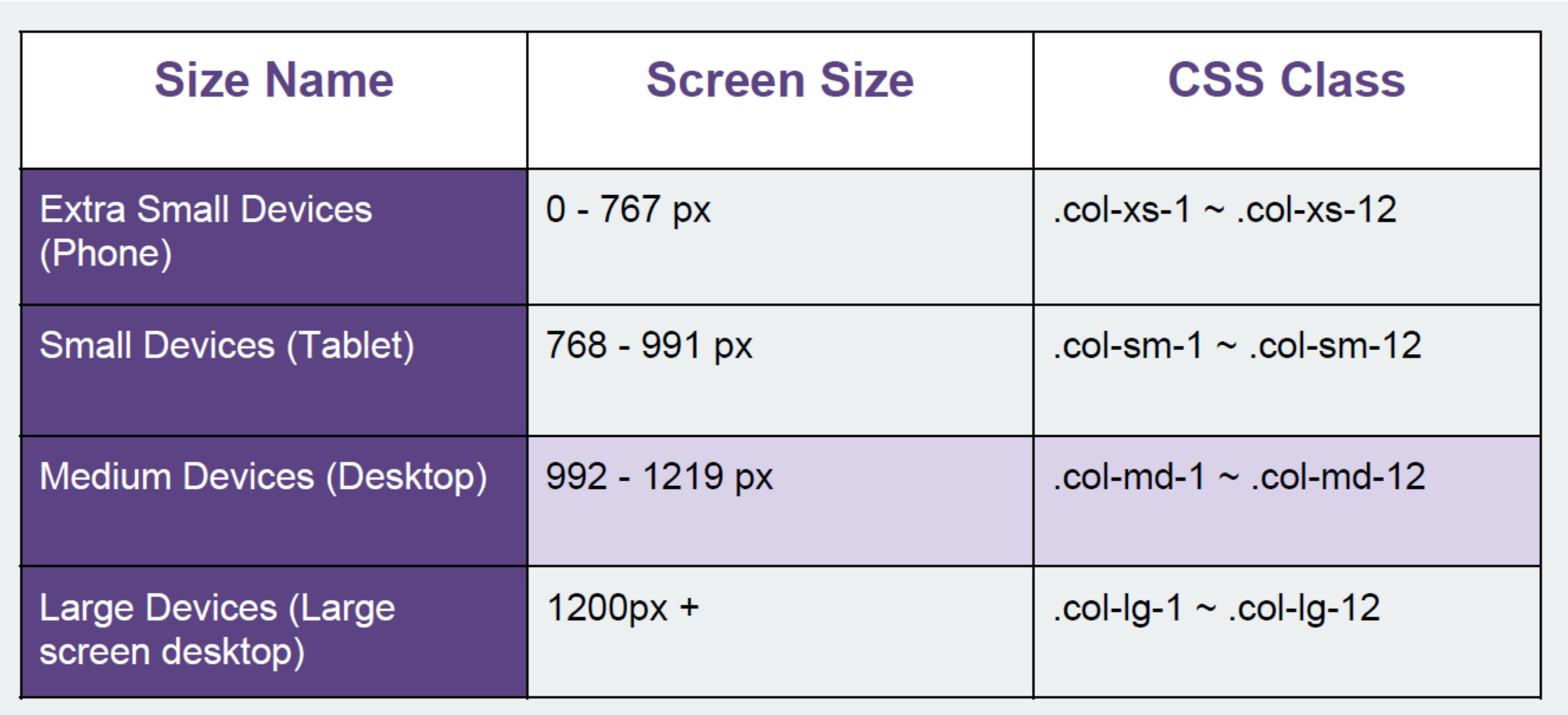
Spans are indicated in units of 12ths



<table>
<tr><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td><td>span1</td></tr>
<tr><td colspan="4">span4</td><td colspan="4">span4</td><td colspan="4">span4</td></tr>
<tr><td colspan="4">span4</td><td colspan="8">span8</td></tr>
<tr><td colspan="6">span6</td><td colspan="6">span6</td></tr>
<tr><td colspan="12">span12</td></tr>
</table>

In bootstrap, the column spans always need to add up to 12

4 Sizes of Bootstrap Grid

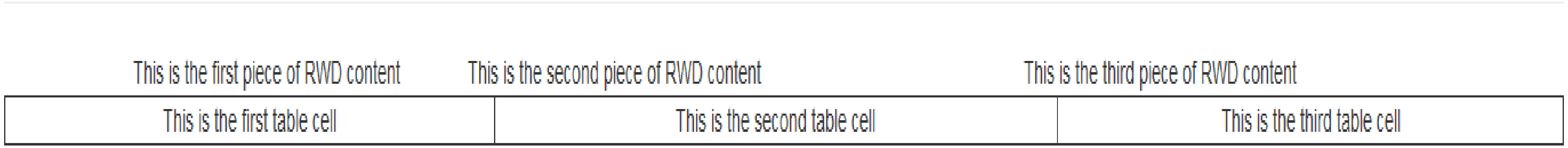


Size Name	Screen Size	CSS Class
Extra Small Devices (Phone)	0 - 767 px	.col-xs-1 ~ .col-xs-12
Small Devices (Tablet)	768 - 991 px	.col-sm-1 ~ .col-sm-12
Medium Devices (Desktop)	992 - 1219 px	.col-md-1 ~ .col-md-12
Large Devices (Large screen desktop)	1200px +	.col-lg-1 ~ .col-lg-12

Using Bootstrap Grid System

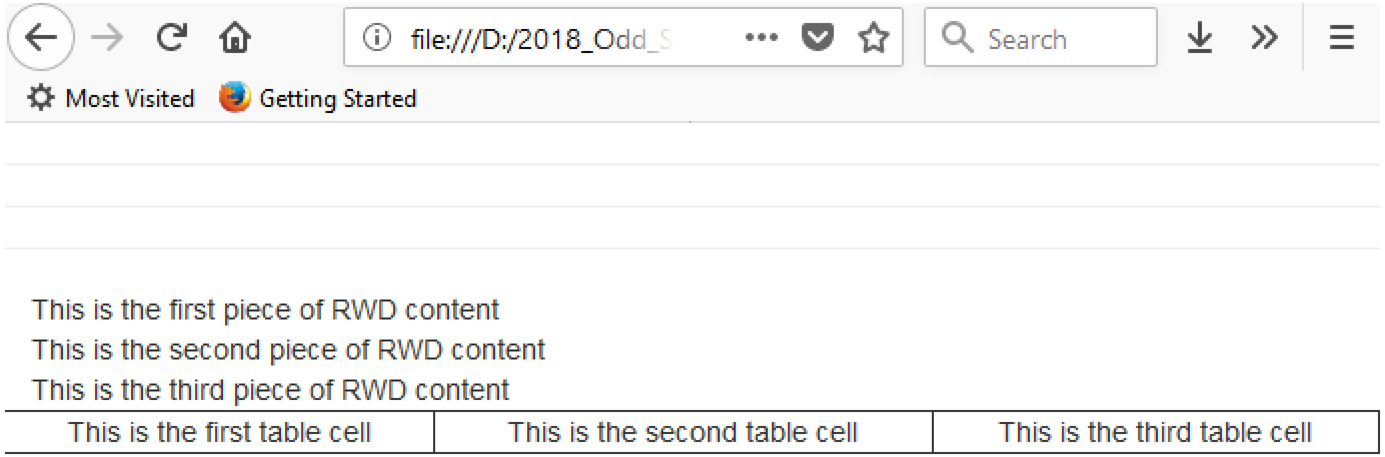
- Content is organized into **rows**: “horizontal group of columns”
- If no set width is specified, Bootstrap will automatically size all columns in a row so that they are equally spaced
- Column classes allow user to indicate width of a column represented as columns out of 12 that should be used

```
<div class="container">
    <div class="row">
        <div class="col-md-3">This is the first piece of RWD content </div>
        <div class="col-md-5">This is the second piece of RWD content
</div>
         <div class="col-md-4">This is the third piece of RWD content </div>
    </div>
</div>

  <table width="100%" border="1">
     <tr align="center">
        <td>This is the first table cell</td>
        <td>This is the second table cell</td>
         <td>This is the third table cell</td>
    </tr>
  </table>
```



This is the first piece of RWD content This is the second piece of RWD content This is the third piece of RWD content

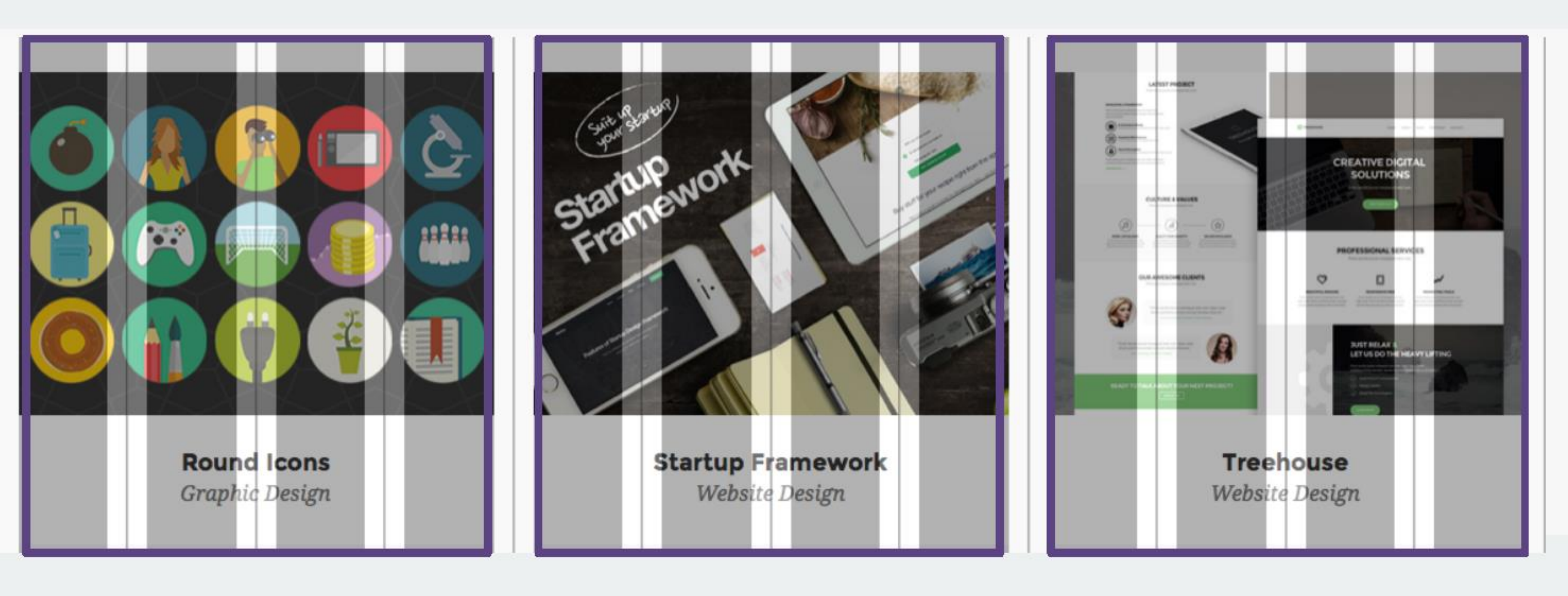
This is the first table cell	This is the second table cell	This is the third table cell



This is the first piece of RWD content
This is the second piece of RWD content
This is the third piece of RWD content

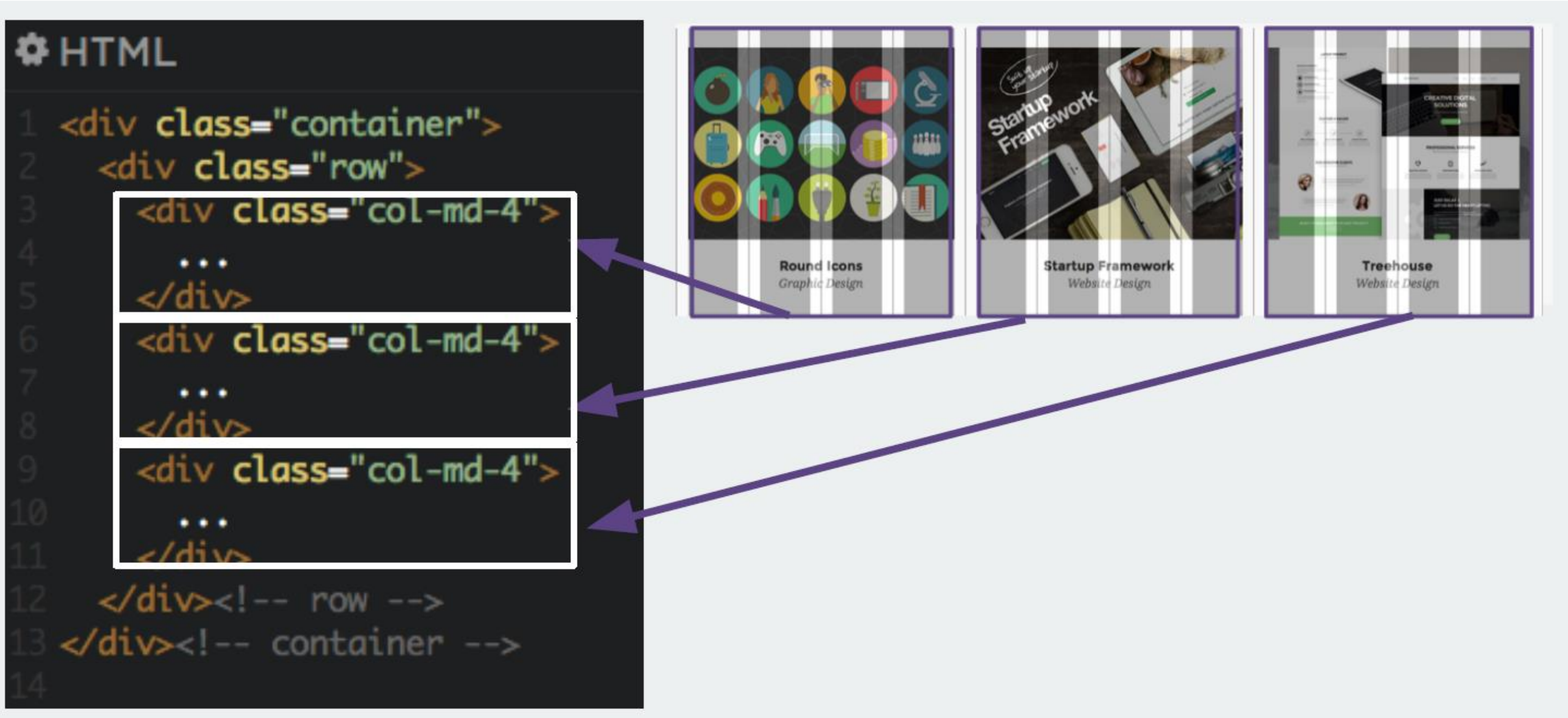
This is the first table cell	This is the second table cell	This is the third table cell

Bootstrap Grid Example



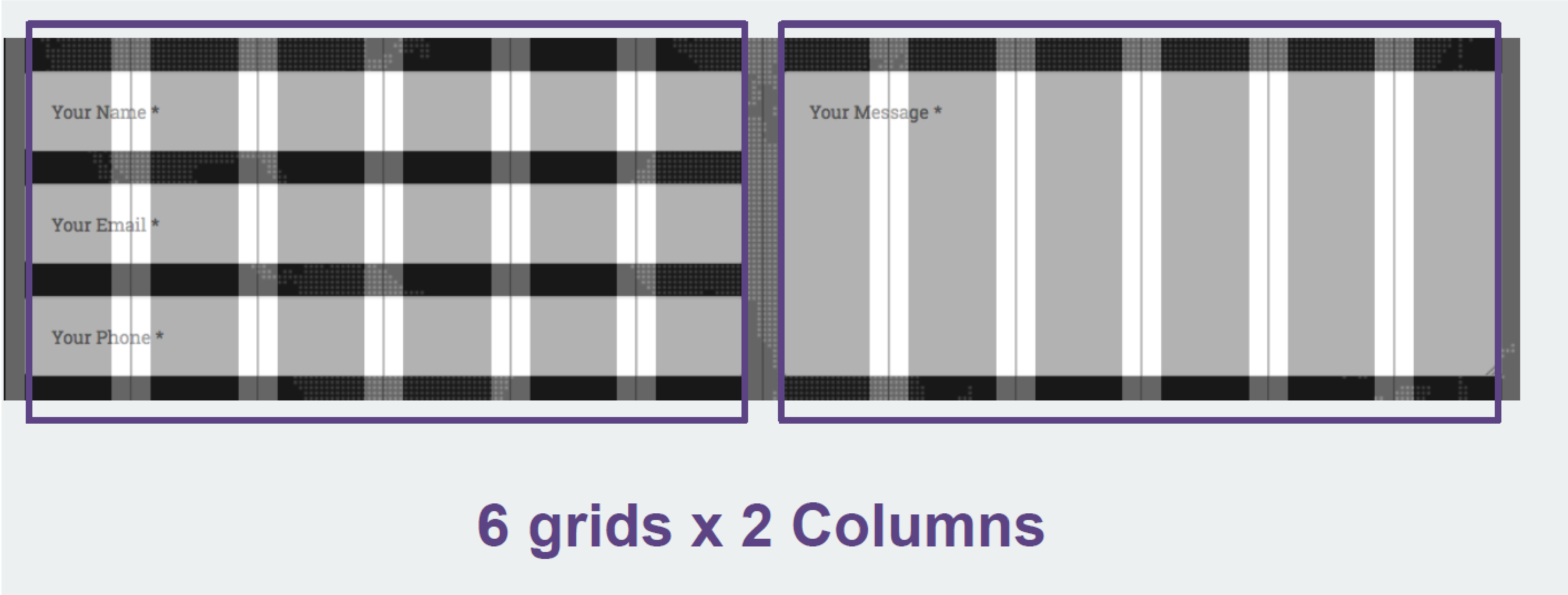
4 grids x 3 Columns

Bootstrap Grid Example

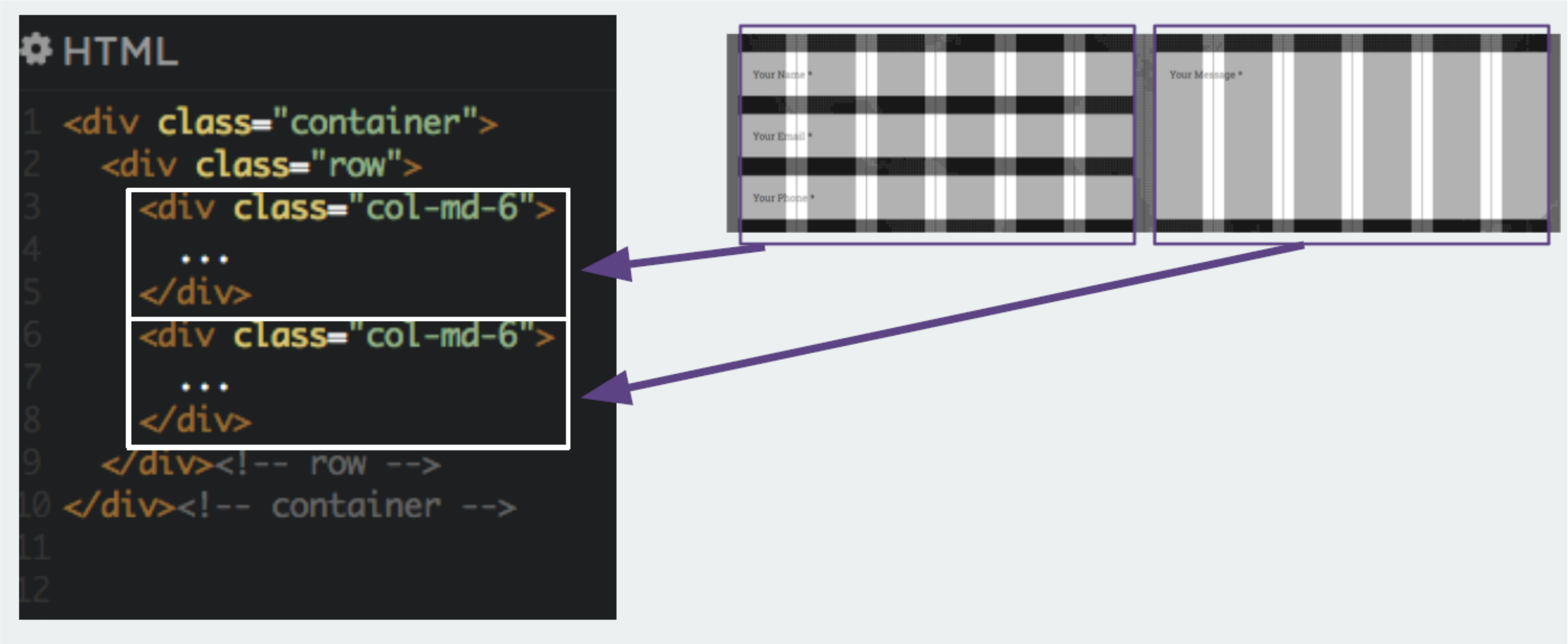


```
<div class="container">
  <div class="row">
    <div class="col-md-4">
      ...
    </div>
    <div class="col-md-4">
      ...
    </div>
    <div class="col-md-4">
      ...
    </div>
  </div><!-- row -->
</div><!-- container -->
```

Bootstrap Grid Example

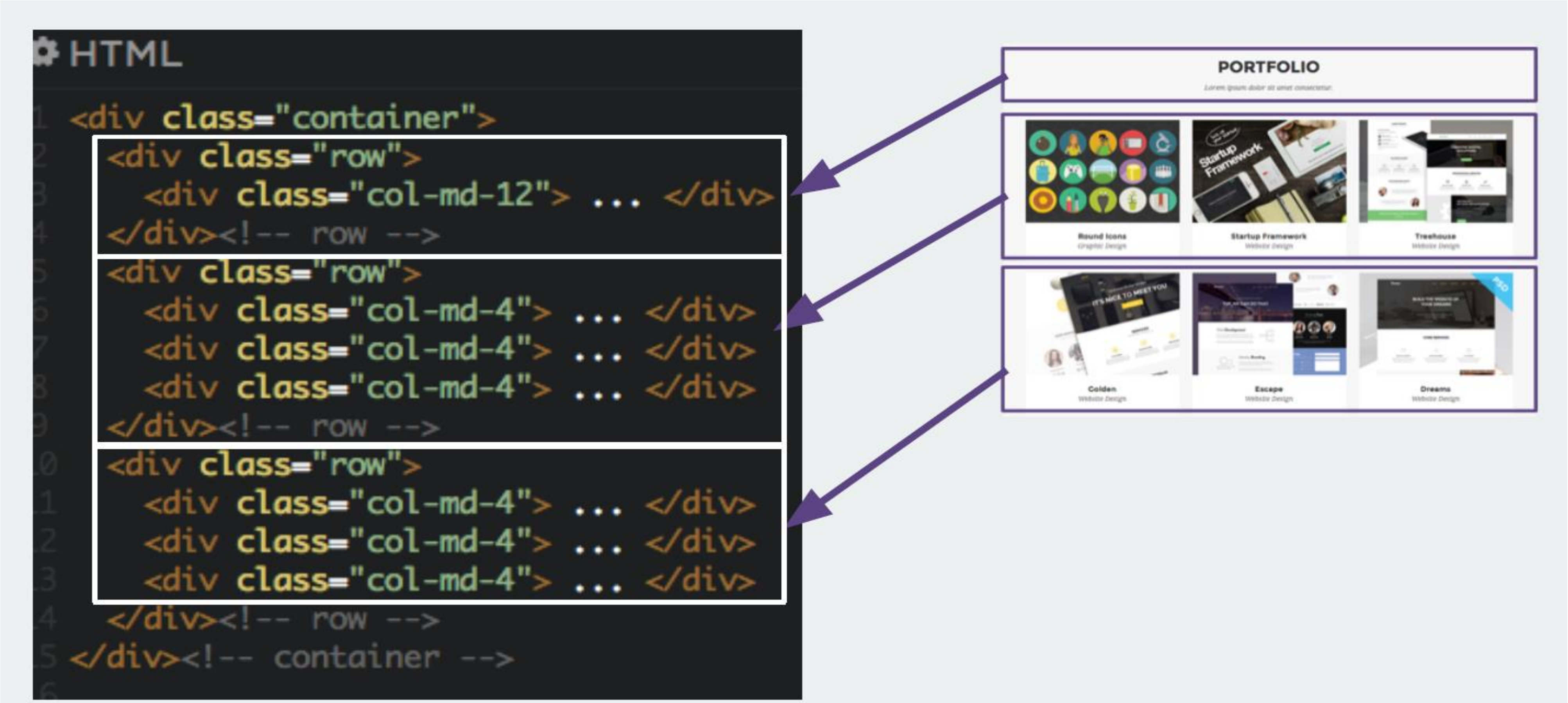


Bootstrap Grid Example



```html
<div class="container">
  <div class="row">
    <div class="col-md-6">
      ...
    </div>
    <div class="col-md-6">
      ...
    </div>
  </div><!-- row -->
</div><!-- container -->
```

Bootstrap Grid Example



HTML

```html
<div class="container">
  <div class="row">
    <div class="col-md-12"> ... </div>
  </div><!-- row -->
  <div class="row">
    <div class="col-md-4"> ... </div>
    <div class="col-md-4"> ... </div>
    <div class="col-md-4"> ... </div>
  </div><!-- row -->
  <div class="row">
    <div class="col-md-4"> ... </div>
    <div class="col-md-4"> ... </div>
    <div class="col-md-4"> ... </div>
  </div><!-- row -->
</div><!-- container -->
```