

Intro To Advanced Programming Language Sessional

CSE 224

LEC RAIYAN RAHMAN

Dept of CSE, MIST

raihan@cse.mist.ac.bd

Your Journey So Far!



What're Advanced Programming Languages Then?



The “not-advanced” prog languages



Advanced prog languages - “Stronger”?



or “Cooler”?

Well.... Not Exactly



It's “What” we use programming languages for and
“How” we use them!

Course Outcomes (Hopefully)

- ❏ **CO 1:** To use programming languages to “Design” Real Time Projects
- ❏ **CO 2:** To use programming languages to “Build” and “Develop” Real Time Projects.
- ❏ **CO 3:** To use programming languages to store and retrieve data efficiently.

OBE Course Curriculum: [Click Here](#)

So We'll be using programming to design and develop
real time, user centric projects.

And we'll do it for Web platform.

The WebDev Process

FrontEnd

1. How You “draw” (render) the Website.
2. It’s what the user sees
3. Executed at user end as well! Using web browsers.



BackEnd

1. Store and retrieve user’s data.
2. Communicate with the server for processing.



Further “Processing”

1. Edit an image! (DIP)
2. Suggest best products to buy for a particular customer (ML, AI)
3. Talk to a user with a chatbot! (AI - NLP)

Web Development Process

The sites you develop may range from a tiny personal blog site to a full-scaled social media application with image processing, chatbots, targeted ads and whatnot!

The WebDev Process - Frontend

www.facebook.com
<http://amazon.com/>

The WebDev Process - Frontend

- ❑ Need to use specific programming languages to “draw” a website.
- ❑ It’s what we present to the user, visually.
- ❑ It’s the combination of all the colors, links, buttons, forms, textboxes, images, animations in a website, the specific style & combination of which can make the site look either terrible or amazing!
- ❑ It’s what drives the user’s experience (at least the first impression).
- ❑ We design how the user interacts with various pages and elements of the website.
- ❑ And all of the commands are executed at each user’s own device, in their web browser or application.

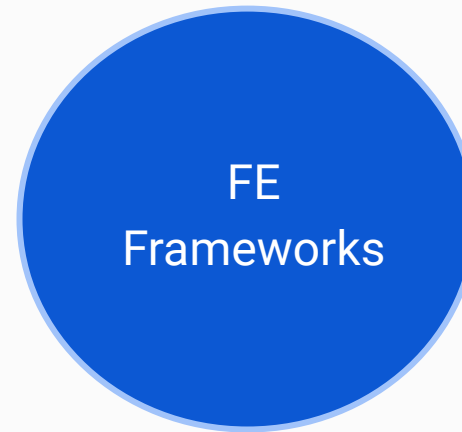
The WebDev Process - Frontend



Draws the structure
of the website



Helps you give
further style and
color to the website



Ensures device
dynamicity and
makes your job
easier



Packages codes into
a time-saving form

What Language?

- A combination of languages
- HTML and CSS – to design the front end or UI of the website
- Javascript – To make the page designed with HTML and CSS dynamic



HTML

- Hyper Text Markup Language
- Follows a specific format that your browser can read very easily.
- Has specific sections or elements that categorize a web page into sections.
- Can insert specific “tags” into each section to design the page as you want.
- These tags and elements together tell the browser how to render (draw!) the website.
- Current Standard is HTML5

CSS

- CSS stands for Cascading Style Sheets
- Gives more detail on each element of HTML document
- Describes the “style” of an HTML document.
- You can use
 - 1) Color, text styles, fonts
 - 2) Background and images
 - 3) Text alignment
 - 4) Padding and Margin
 - 5) Floating positionTo render the web page to the exact design you have in mind.

It Gets Easier!

- Might get harder to manage and time consuming to build from scratch as websites get larger.
- Use frameworks!
- Or Content Management Systems (CMS) like Wordpress
- Bootstrap- A front end web framework. Makes building the web UI much faster.



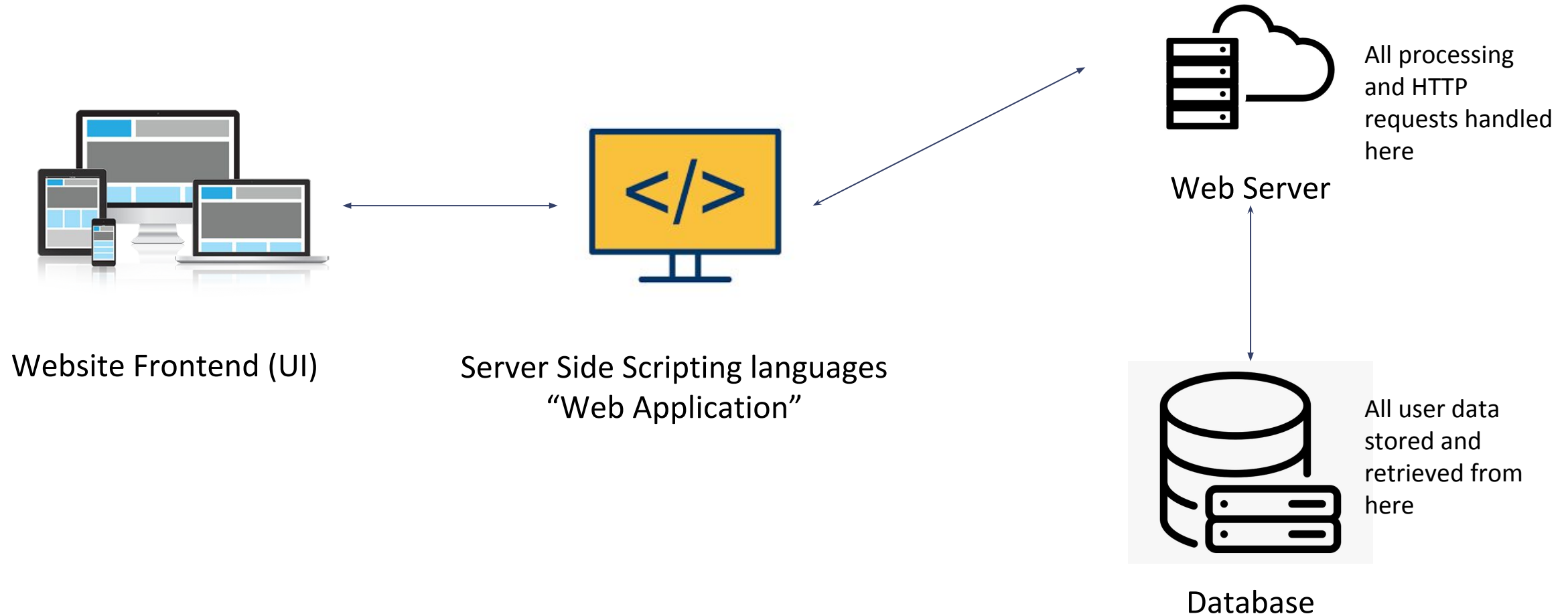
The WebDev Process - Backend

- ❑ A website would be quite shallow and not useful without a backend.
- ❑ Whenever there's any "functionality" involved, it's very likely that backend processing is taking place.
- ❑ For example, It's when
 - ❑ you create and log into your account in a website
 - ❑ Post a new picture, write a post, order an item etc.
 - ❑ Browse other peoples' profiles, posts, videos, content etc.
- ❑ In other words, it's when you store, process and/or retrieve data from the user.

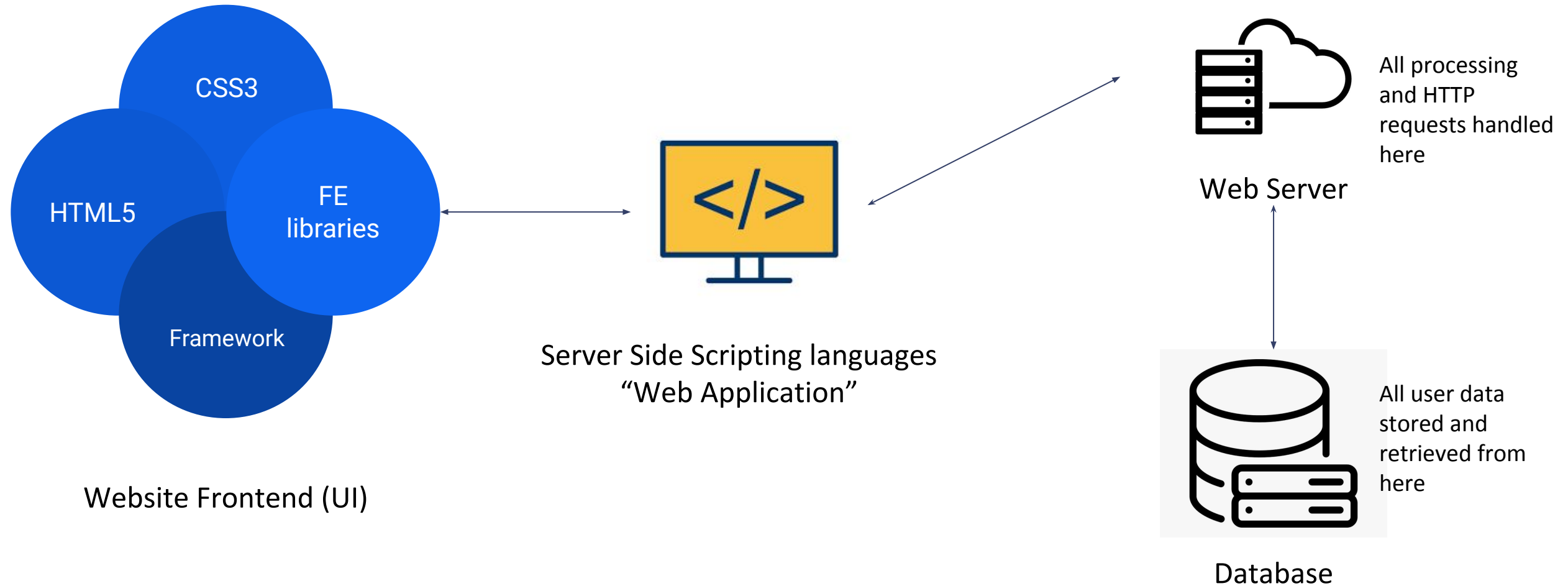
The WebDev Process - Backend

www.facebook.com
<http://amazon.com/>

The WebDev Process - Backend



The WebDev Process - Full stack



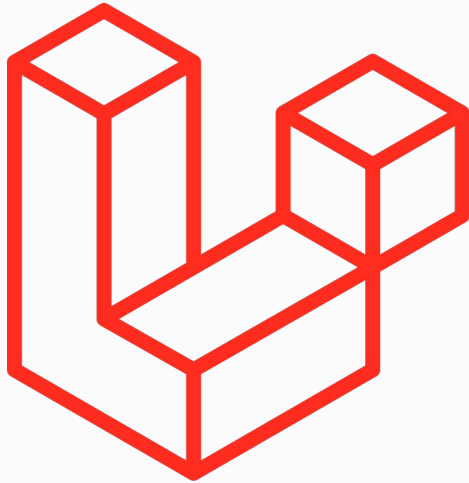
What Languages?

- PHP, Javascript, JS, Python – Server-side scripting language. Perform some processing or action remotely and return results to your webpage.
- SQL – Standard Query Language. Used to store, manipulate and retrieve data to/from a webpage.



It Gets Easier!

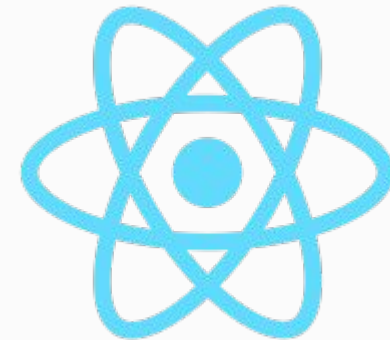
- Use web frameworks to build sites faster and more efficiently.



Laravel



Codeigniter



React

Now, How to write Codes?

- Use simple text editors to write the code and just name the file extension as .html
- Then use your browser to open the file.
- Better to use specialized text editors like Notepad++, Atom or Microsoft visual studio code.

Download Atom: <https://atom.io/>

And How to Publish your Projects?

- There are quite a few free cloud servers.
- Upload there and point to it by your own domain. For free services, use their free subdomains.
- Or if you have your own unique domain, use that.
- Github is one such free server. It also helps you to store and do version control of your software projects!

Course Outline - What We'll Cover

Course Curriculum

Sample Projects - What You'll do

Get a few ideas - Last Year's Sessional!

Tasks

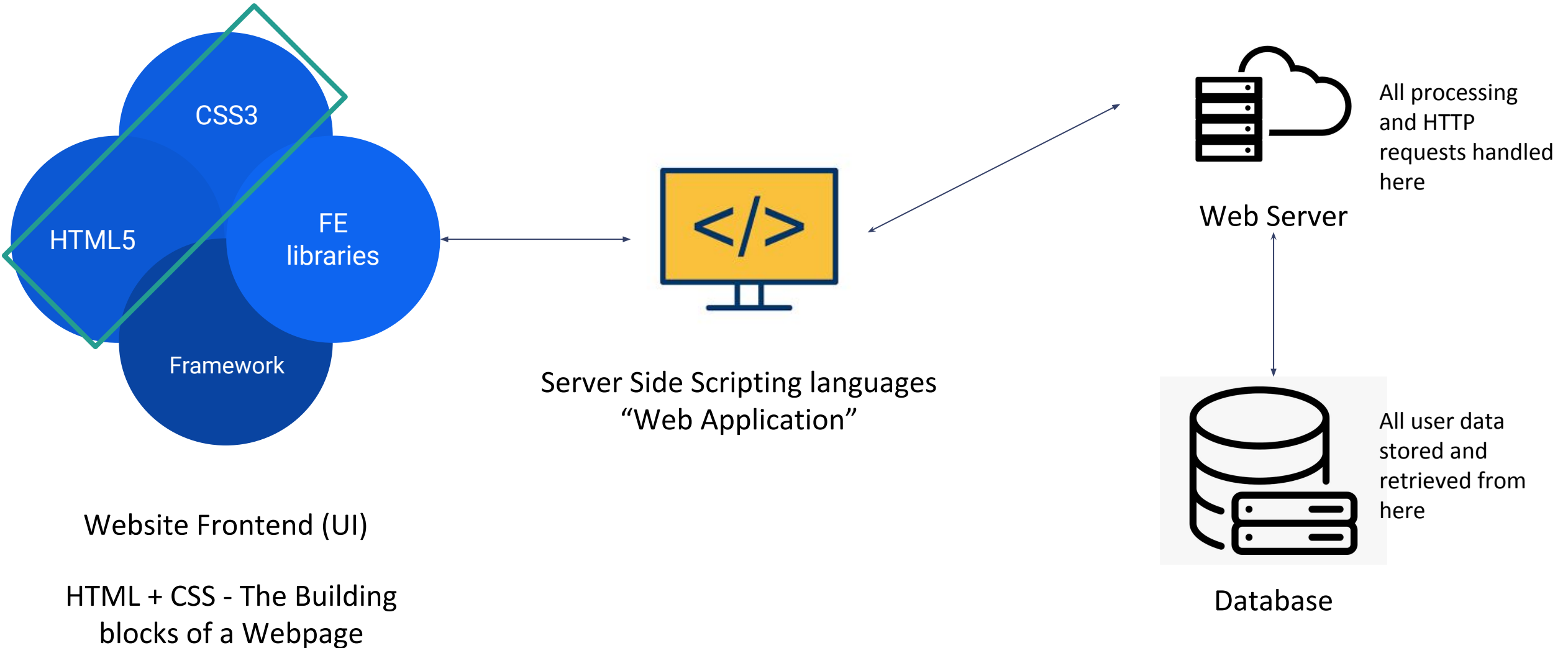
- ❏ Go through the sample projects and start thinking of project ideas
- ❏ Download Atom or MS visual studio code.

Intro to Making Webpages with HTML and CSS

Week 2, CSE 224

Lec Raiyan Rahman
Dept of CSE, MIST
raiyan@cse.mist.ac.bd

What We'll Focus on Today



HTML

- ❏ **H**yper **T**ext **M**arkup **L**anguage
- ❏ The building block of the web page layout
- ❏ Makes the “Skeleton” (/the layout) of the Webpage
- ❏ Follows a specific format that your browser can read very easily.
- ❏ Written with specific “tags” to design the page as you want.
- ❏ These tags together tell the browser how to render (draw!) the website.
- ❏ Current Standard is HTML5

HTML



What your Webpage will Look Like With the HTML “Skeleton”

HTML

[Go to blog page](#)

Hello World!

Lorem ipsum dolor sit, *consectetur adipiscing elit*. [GOTO MIST Webpage](#) magni saepe ratione eligendi ipsam quidem quos accusamus, labore, impedit autem debitis quaerat explicabo molestiae nulla suscipit eveniet maxime.

ipsum dolor sit amet.

- [List Point 1](#)
- [List Point 2](#)
- [List Point 3](#)
- [List Point 4](#)

1. [List Point 1](#)
2. [List Point 2](#)
3. [List Point 3](#)
4. [List Point 4](#)


Name	Student ID	Email
Raiyan	1234	raiyan@cse.mist.ac.bd
Muhaimin	1233	muhaimin@cse.mist.ac.bd

Leave a Comment!

Name

Email

Age

Date 

User Type


Comment

- [Home](#)
- [About](#)
- [Services](#)

Affordable Professional Web Design


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam eu luctus ipsum, rhoncus semper magna. Nulla nec magna sit amet sem interdum condimentum.

Subscribe To Our Newsletter



HTML5 Markup

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus mi augue, viverra sit amet ultricies



What your Webpage will Look Like With the HTML “Skeleton”

What We'll Do Today

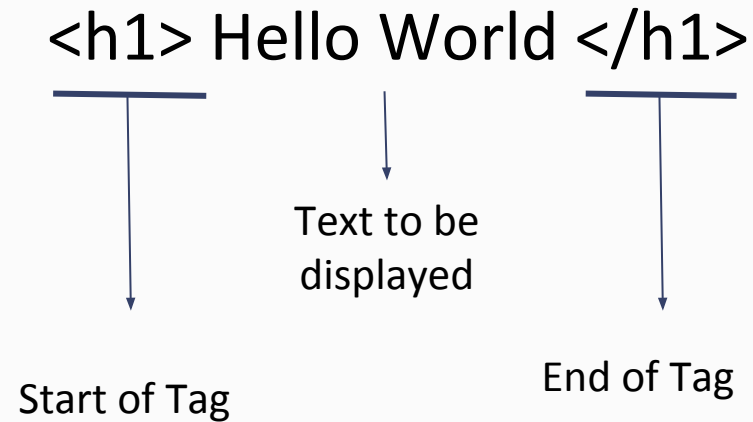
1. Make the HTML layouts of two pages from MIST Website.

[Page 1](#) [Page 2](#)

We'll learn the basics of HTML through this.

2. Then, we'll see how to make a fully responsive complete website with HTML + CSS.
We'll have a brief intro with CSS through this, more on next class.

HTML Tags



The specific tags tell how the text/item will be displayed in the web page. H1 is a tag that displays texts as a headline.

Tags are used to denote single line specific texts to an entire section (say the header or the footer) of the web page!

HTML Tags With Attributes

` Visit MIST Website `

First Attribute of the tag <a>
Tells where to go when the link
is clicked

Second Attribute of the tag <a>
Tells in which browser tab the
link will open

The text that'll be displayed as
the link

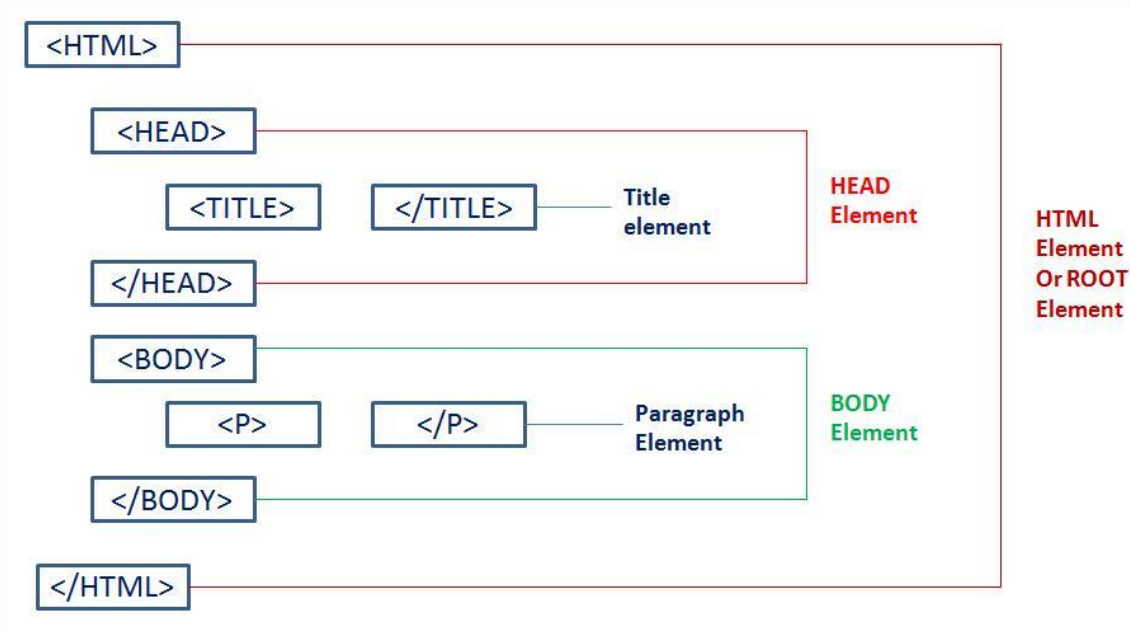
<a> tag links another page to a text. The above line will result in something like:
Visit MIST Website in our webpage!

Block vs Inline HTML Tags

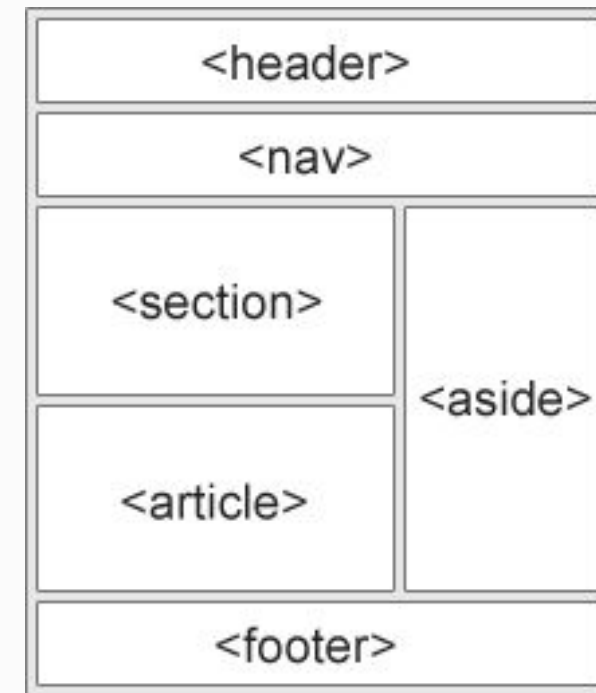
Block	Inline
<p>Goes to a new line automatically after the tag is done.</p> <p><h1> - <h6> <p> <div> <form></p> <p>All are block tags.</p>	<p>Doesn't go to a new line after the tag is done.</p> <p> <label> <input></p> <p>All are inline tags.</p>

Let's Take a look at a web page and try to see if we can make out a common structure!

HTML5 Semantics



HTML Elements



Some HTML5 semantic elements inside `<BODY>`

HTML5 Semantics

Start of CSE 224
Hello CSE 224! Hope you'll pick up some new skills! We'll continue it with Database next year.
- CSE 224 teachers



```
<html>
  <header>
    <p> Start of CSE 224 </p>
  </header>
  <body>
    <p> Hello CSE 224! </p>
    <p> Hope you'll pick up some new skills!</p>
    <p> We'll continue it with Database next year. </p>
  <footer>
    <p> -CSE 224 teachers </p>
  </footer>
</body>
</html>
```

Let's Get to Coding (Part 1)!

[Page 1](#) [Page 2](#)
[Full Code](#)

Intro to CSS

Week 3, CSE 224

Lec Raiyan Rahman
Dept of CSE, MIST
raiyan@cse.mist.ac.bd

CSS

- ❏ CSS stands for Cascading Style Sheets
- ❏ Gives more detail on each element of HTML document
- ❏ Describes the “style” of an HTML document.
- ❏ We can take each HTML tag and define exactly how they’ll look in the corresponding CSS file.

You can use the attributes-

- 1) Color, text styles, fonts
- 2) Background and images
- 3) Text alignment
- 4) Padding and Margin
- 5) Floating position and many more

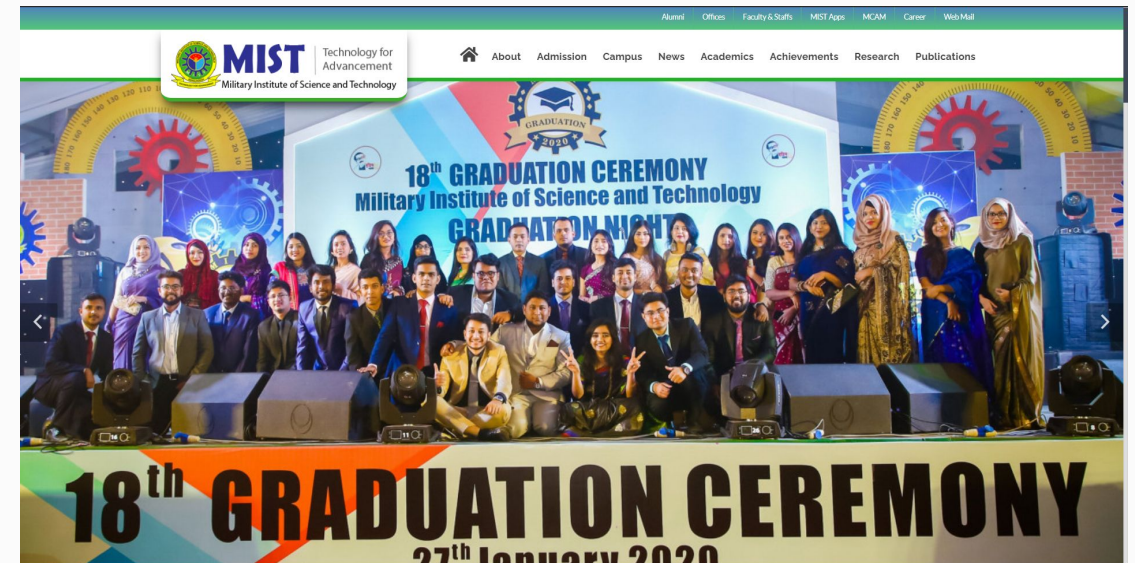
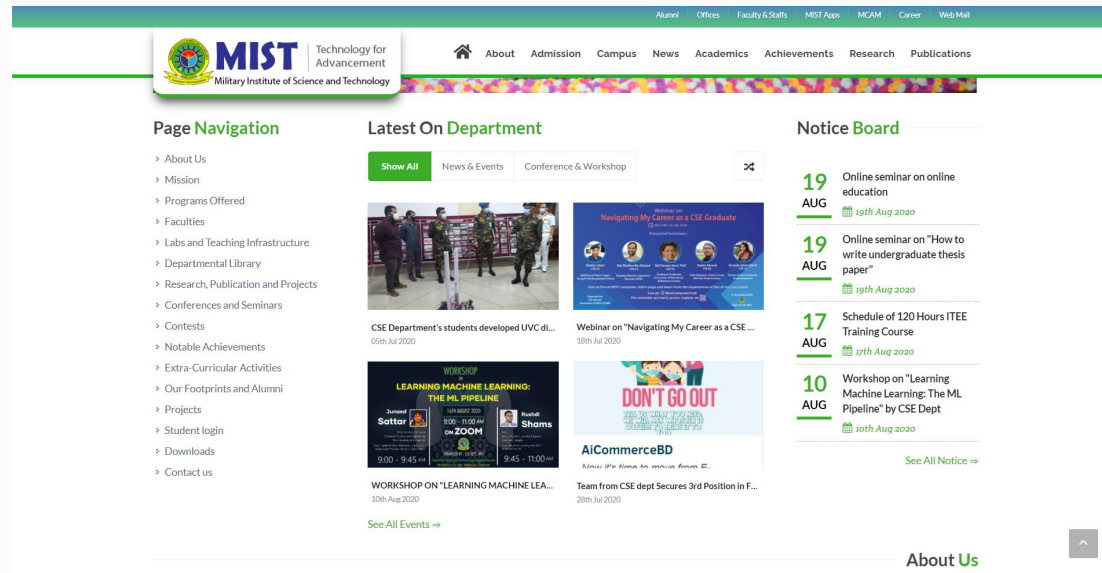
To render the web page to the exact design you have in mind.

HTML + CSS



What your Webpage will Look Like With the HTML + CSS (and then some)

HTML + CSS



What your Webpage will Look Like With the HTML + CSS (and then some)

Ways to Define HTML Tags with CSS

Global
(Directly use that tag)

`<body>` or `<div>` or `<nav>` or `<h1>`

Class

`<div class = "blogpost">` or `<nav class="mistnav">`

id

`<h1 id = "blogheader">` or `<p id="blogtext">`

Writing Style Specs In CSS File

Global	Class	ID
<pre>ul { margin:0; padding:0; }</pre> <p>From now on, all unordered lists will have 0 padding and margin.</p>	<pre>.container { width:80%; margin:auto; overflow:hidden; }</pre> <p>Whenever you write class="container" in a div the tags inside will follow the design attributes of container.</p>	<pre>header #branding { float:left; }</pre> <p>The specific tag with the ID "branding" inside header tag will float to the left.</p>

CSS - “Nested” Declarations

```
header li {  
  /*CSS attributes*/  
}
```

We’re defining the design of the list Item (li) inside header.

```
body .blogpost h1 {  
  /*CSS attributes*/  
}
```

We’re defining the design of the the header inside the class named “blogpost” inside global tag body.

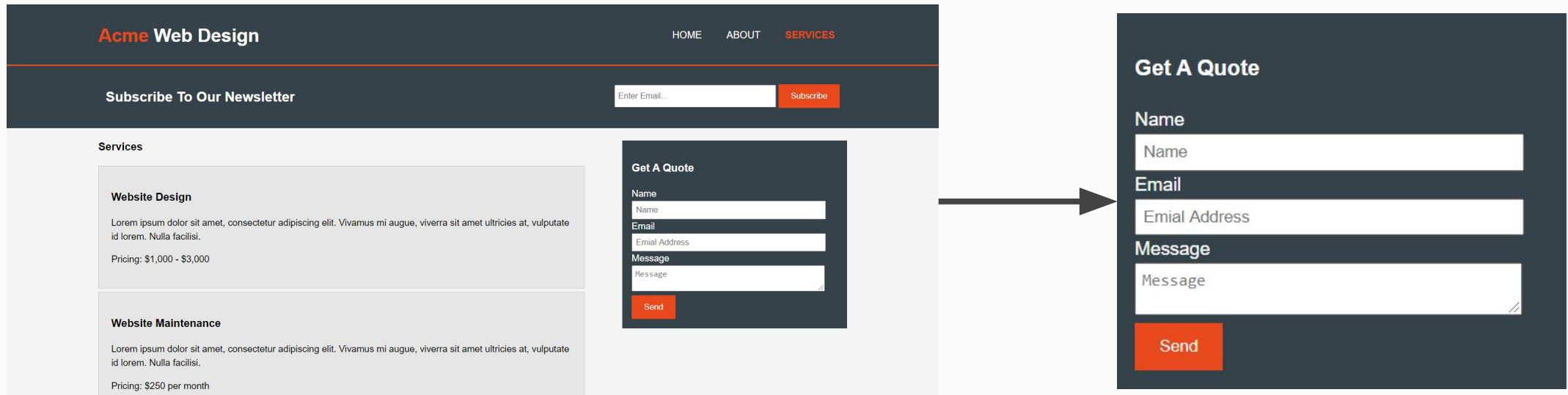
```
aside #deptnav ul {  
  /*CSS attributes*/  
}
```

We’re defining the design of the unordered list inside a class named “deptnav” in the aside tag.

CSS - “Nested” Declarations

```
aside#sidebar .quote input,  
aside#sidebar .quote textarea  
{  
  width:90%;  
  padding:5px;  
}
```

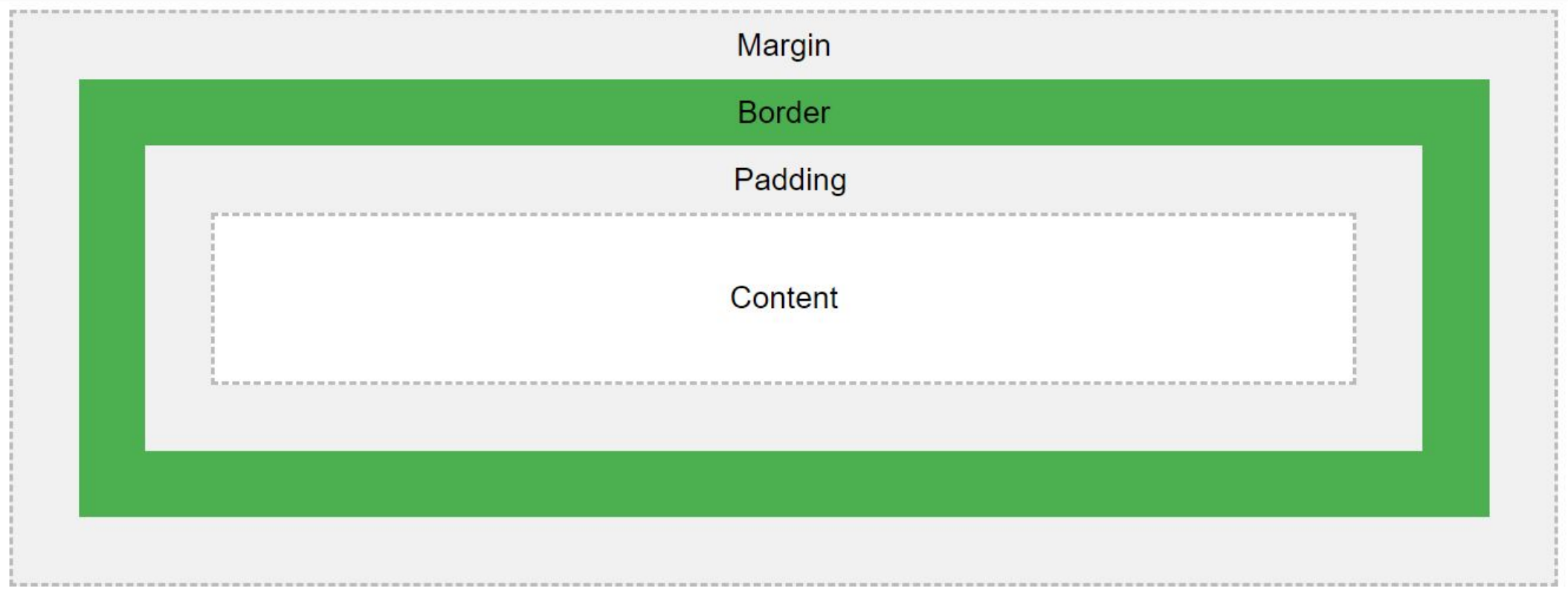
We’re defining style of two items here. The input field/tag inside the id “quote” inside the class “sidebar”, all of which are inside the tag aside. Also the textarea inside the same id, class and global tag.



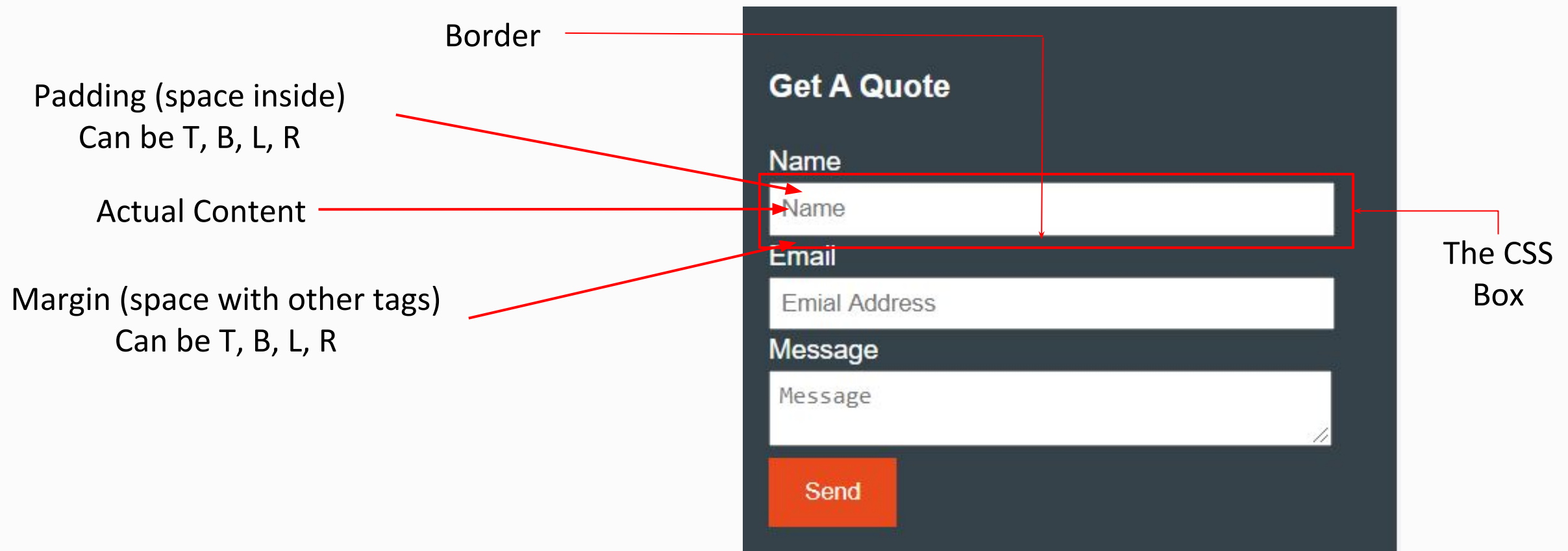
Global vs Class vs Id - Summary

Global	Class	ID
<p>Take the exact html tag and assign design attributes to it globally.</p> <p>So, they'll be applicable for whenever you use that Tag in the HTML file.</p>	<p>A specific set of design rules for a particular tag (Usually <div>)</p> <p>Usually written as a class in CSS if it'll be used multiple times in the website.</p> <p>Written with a preceding . in CSS.</p>	<p>Define a set of design rules of a particular tag or "sub-tag".</p> <p>Usually written as a ID in CSS if it'll be used once in the website.</p> <p>Written with a preceding # in CSS</p>
<pre>ul{ margin:0; padding:0; }</pre> <p>From now on, all unordered lists will have 0 padding and margin.</p>	<pre>.container{ width:80%; margin:auto; overflow:hidden; }</pre> <p>Whenever you write class="container" in a div the tags inside will follow the design attributes of container.</p>	<pre>header #branding{ float:left; }</pre> <p>The specific tag with the ID "branding" inside header tag will float to the left.</p>

CSS Box Model



CSS Box Model



CSS Attributes

(We'll get to know them from a simple website!)

Let's Get to Coding (Part 2)

Full Code

Resources

1. [A Guide to HTML](#)
2. [A Guide to CSS3](#)
3. [A Responsive Website Design With HTML5](#)

Task

- ❑ Go through the classroom codes, slides and provided resources. Then-
 - ❑ Submit Your Idea through [this form](#). (Submit after next week 4)
 - ❑ The idea must satisfy the following conditions -
 - ❑ Have at least 7 different and distinct pages in the front end.
 - ❑ Should make use of HTML, CSS and some FE Framework (to be taught).
 - ❑ Should have dynamic FE elements with JS (to be taught).
 - ❑ Should have some DB functionality, at least posting and retrieving comments in a blog, user accounts and contact us form. (to be taught)
- ❑ The project idea should not be too generic or a direct copy of (or have uncanny similarities to) any of last year's projects.
- ❑ Check out previous year's [projects](#) to get an idea of the project scope and complexity.

<p id =“thankStyle”>Thank You!</p>