HTML(Content language): Hyper test markup language

GUI: Graphic user interface

Jsp pages, JSX, portlets, Timeleaf (Advance Html)

We have various tags in html like header, paragraph, break, Tables, Lists, division etc

Body consists of all things which we want to display to user

Here we have tags and also attributes within the tags

**Frames & Iframes**

**Frames: One HTML page can have multiple divisions. Each division can depict a frame**

**Entire webpage we don’t need to load if we use frames**

**Iframes: These are inline in nature. It is like embedding a document**

In HTML, JS can work inside head, body and External JS file.

IDE provides us with Autointellisense(Automatic Recommendations)

**Angular Testing**

Bootstrap & ES6

CSS(Styling language): Cascading style sheets

**MODEL VIEW CONTROLLER ARCHITECTURE:**

**We follow this approach, we have different types of classes**

**1)Service**

**2)Dao**

**3)Repository**

**4)Controller**

**Each activity will have a skeleton using all of the above classes. So, whenever we need a new functionality. We create those classes for every new functionality. So, we you want to roll off(Go to next project)Then we need to knowledge transfer using those classes to another new person. At that time of knowledge transfer these classes will help the new person to understand things easily. Ex: index.css – (All Css content in CSS file externally)**

**To make our code clean and nice**

**F5 – For running in VScode.**

**Test Suite:** It will consist of some 10 test cases, where we can execute in different browser platforms.

**MODEL VIEW CONTROLLER ARCHITECTURE:**

**Model: It is type of business data**

**View: Front end(Angular & React)**

**Controller: Framework that we use**

**Bootstrap: Do my coding easily, use already defined things to make our job easy.**

**Framework: Facilitate the things using already predefined codes from library**

**It has libraries related to HTML, CSS and JS  
Viewport: whatever is visible in our webpage is viewport.**

**Container: Everything goes into container It is used to set the margin and fix the display area.**

**Bootstrap makes it easier, fast, builtin libraries, viewport, container**

**Grid: It is a box, pixel concept**

**Type of grid: Phone, laptop, Ipad**

**Grid system: The way we display the things**

**Built in classes:**

**Container: This will define the layout of display. It has a fixed width**

**Container fluid: It will take the width of the view port.(we can zoom till a limit only)**

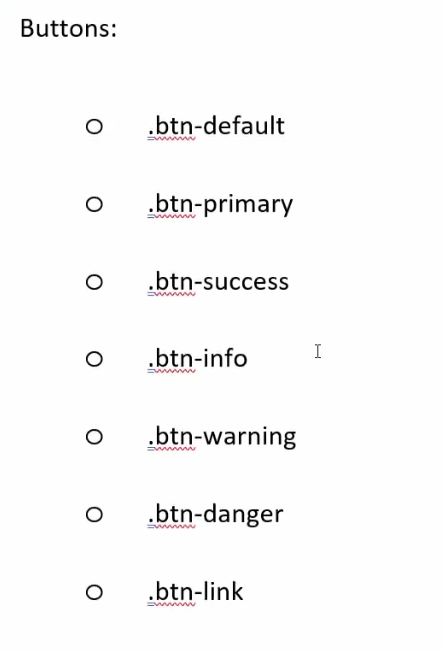
**If we normally use these in built things in html or css it will error out. We need to use these in bootstrap. So that our browser library can understand.**

**We have another classes like:**

**Jumbotron(When we want to focus or highlight on a particular part of the webpage. We can use jumbotron.**

**Container**

**Buttons(btn)**



**Classes available in bootstrap grid system:**

**Xs – For phones**

**Sm – For tablets**

**Md – For laptops**

**Lg - Big desktops**

**It is a HTML, CSS and JS framework. It makes our job easy when we want to build our website**

**It is faster. It uses HTML and CSS tags only**

**NodeJS: It is a JS run time environment**

**Nodejs, Angular UI, Anguar JS, NPM(Node package manager)**

**Developers: They will learn Spring, Angular, Micro services**

**Tester: Protractor, Cypress, Jasmine and karma**

**Scope may vary for different application**

**2 and 4 are generic**

**But some others are specific to angular or something else specifically**

**Btn – means button**

**ES6: (ECMA script 6)**

**It defines a standard for all your scripting languages.Ex: Js, jscript etc**

**After compilation the js code also becomes .class file similar to java file**

**To use ES6 we need to install nodejs**

**Nodejs is a platform which facilitates frameworks like angular, react and ES6 etc.**

**We have all concepts on java in ES6 as well like constants(literals), variables, operators, loops, Arrays, functions, objects, classes and other advance features.**

**Local, Instance, static/class these are variables that we have:**

**Advance features of ES6:**

**Spread operator:**

**Let var = […values] – spread operator**

**Let arr1 = [1,2,3,4]**

**Let arr2 = […arr1,5,6,7]**

**Console.log(arr2);**

**o/p: It will print all numbers**

**Rest Operator:**

**This works with the function parameters, It should last argument in the function.**

**Function f1(int a, int b) // These are the parameters**

**Funtion f(a,b,c,…moreArgs){**

**}**

**F(1,2,3,4,5) // These are the arguments**

**The more values which is passed will be arguments.**

**At the receiver end it is called parameters.**

**Let, Const**

**Constant a = 5;**

**Access modifiers:  
1)Private: Only within the class**

**2)Public: Available outside the package**

**3)Protected: It is available for sub classes and for package**

**4)Default: Only with in the package**

**Labeled break and Labeled continue**

**New feature is Labeled function:**

**F: function sample(){**

**}**

**Arrow function(ES6 feature):**

**It is also called anonymous function:**

**They work similar to Lamda functions in java**

**Let fname = (a,b) => {**

**}**

**After installing node.js**

**Open command promt and type the following**

**To install libraries of nodeJS we need to use** npm install npm -global

Use node –v: In terminal(To know the version)

**For linking html server**

**Commands:**

**1)npm install http-server(npm i http-server –g)**

**2)npx http-server**

Node firstprogram.html(To run file)

**Bootstrap: It’s a framework of Html, css and js**

**Es6 is js generization(Standardisation)**

**Object is instance of a class.**

**This is a class**

**Symbol: It is a type in js**

**Symbol, class, object, promise, collections**

**ES6 is added new primitive type called symbol**

**It will help to store some small data in memory of a small duration of time**

**Mutable/Immutable**

**Reference by**

**All these are used in advanced ES6**

**String s= ‘abc’ is same as String s= new string(“abc”)**

**S -> Memory location**

**The values returned by symbol constructor is immutable**

**Const s = symbol()**

**Symbol Stores value immutably**

**== (False)**

**===(True)**

**Promise: Used to deal with asynochonous things.**

**One thread doing a activity and another doing other activity**

**Synchromised**

**Callbacks: help me understand how to tackle this situations (Promises)**

**Notify()**

**NotifyAll()**

**.then()**

**Promise:**

**Let p – new promise((resoleve, reject)=>{**

**Let x=1;**

**If(x==1){**

**Resolve(‘success’)**

**}**

**Else{**

**Reject()**

**}**