**Day 1 : 08—5-2021**

**MEAN Stack :**

**Mongo Db/MySQL Express JS Angular Node JS**

**Phase 1 :**

**Agile : Self learning : SDLC**

**Git : Self Learning : Version Control System.**

**DevOp:**

**Web Technologies :**

**HTML/HTML5, CSS/CSS3, JavaScript using ES5 and ES6 and bootstrap etc.**

**Overview of Node JS**

**Babel**

**Webpack**

**Typescript Overview**

**Phase 2 : TypesScript and Angular 8,9,10,11 etc**

**Phase 3 : Node JS, Express JS (REST API), Mongo DB and Mongoose etc.**

**Phase 4 : Deploy the application in AWS , ES2 and S3 etc**

**SVN :**

**Git : Version Control System**

**Version Control System that records changes on files or folder or project.**

**Distributed Version Control system.**

**Online Shopping :**

**10 member**

**1 login page**

**2 application page**

**3 product details**

**4 product view details**

**5 order details.**

**After finish all coding they have to push the code in central folder (repository).**

**Central Repository (online shopping)**

**Git hub**

**AWS**

**Azure**

**Google cloud**

**Oracle cloud**

**Etc**

**10 member**

**Pull the project from central repository**

**Git**

**1**

**To**

**10**

**After finish the task they push the code to central repository.**

**First command is use to check the version**

**git --version**

**git init** This command is use to create the local repository.

**git status** : This command is use to check status of last command in local repository.

**git add filename :** This command is use to add the file in staging area. Staging area is consider as a buffer memory for git which store information about what will go the next commit.

git commit –m “message” : This command is use move the file from staging area to git local repository.

**Create folder : which contains set of files or project**

**git init : This command only one time(first time)**

**git status**

**git add filename**

**git status**

**git commit –m “message”**

**git status**

**git add . : all file and folder present in current directory.**

**Git hub :**

**It is a one of the remote repository.**

**Git : It is open source software which help to interact with the remote repository (git hub etc).**

**AWS : Code Commit : remote repository**

**git remote add origin URL :**

**This command is use to connect local repository to remote repository.**

**git push –u origin HEAD : This command is use to push the data from local repository to remote repository.**

**Please add new file in local repository folder.**

**git status**

**git add .**

**git commit –m “file added”**

**git push –u origin HEAD (HEAD means last commit)**

**git clone URL : It download all file or folder present in remote repository to local machine as well as it make that folder as local repository.**

**git clone URL : if we are downloading all files or filed first time. That time we have to use git clone URL.**

**git pull : This command is use to pull any new updated in existing repository.**

**git branch : git branch is a moveable pointer which hold more than commit details.**

**git branch : This command is use to display all branches present in local terminal.**

**Default branch may be master or main**

**Old version default branch is master consider**

**New version default branch is main consider.**

**Creating user-defined branch**

**git branch branchname**

**how to switch to user defined branch**

**git checkout branchname**

**Manager – Akash**

**Banking App**

**Sample Template created..**

**Git init**

**Git add .**

**Git commit –m “project created”**

**Git remote add origin URL**

**Git push –u origin HEAD**

**Default branch is main/master**

**Ajay Vijay**

**Git clone URL git clone URL**

**main default branch main default branch**

**Application page Customer page**

**Please create user-defined branch**

**If he/she do changes on default branch**

**git add .**

**git commit –m “message”**

**git push –u origin HEAD**

**Creating the branch**

**git branch branchname**

**git checkout branchname**

**Or**

**git checkout –b branchName**

**Command to remove or delete the branch**

**git checkout main/master branch**

**git branch –D branchName**

**git pull in main/master**

**git push (user-defined branch)**

**Day 2 : 09—5-2021**

**Git init**

**Git add .**

**Git commit –m “message”**

**Git remote add origin URL**

**Git push –u origin branchName (all commit pass)**

**Or**

**Git push –u origin HEAD (last commit in that branch)**

**Git push –u origin**

**To rollback to specific commit we have to use the command as**

**git log : This command is use to display all commit details.**

**git checkout commit-id**

**Use-defined branch**

**Combine 2 or n branch code in one branch using 2 ways**

1. **Merge : Merging brings two line(branches) of development code together while preserving the ancestry of each commit history.**
2. **Rebase : rebasing unifies the line(branches) of development by rewriting changes from source branch so that they appear as children to the destination branch.**

**Web Technologies**

**https://**[**www.google.com**](http://www.google.com) **: URL : Uniform Resource Locator**

**http: protocol : set of rules which help to communicate more than one device or machine.**

**Hyper text transfer protocol.**

**S : secure.**

**www: world wide web**

**google : domain or server or search engine.**

**Com : commercial application.**

**req(http/https)--🡪**

**Client Server**

**🡨--Res(http/https)**

**Html / html5**

**CSS / CSS3**

**JS**

**HTML / HTML5 is use to display the content in browser in different format. Skeleton**

**CSS /CSS3 is use to apply formatting style or presentation logic to content. Skin**

[**http://gmail.com**](http://gmail.com)

**Form Validation**

**JS : Using JS we can do event or action on that contents.**

**HTML : Notepad**

**Or**

**HTML5**

**VSCode**

**HTML : Hyper text mark up language: which help to create the web pages.**

**HTML provide lot of pre-defined tags or elements.**

**Syntax**

**<tagName> opening tag**

**</tagName> closing tag**

**<tagName/> self closing tag**

1. **Html**
2. **Head**
3. **Body**
4. **Title**
5. **P – paragraph**
6. **Br – break**
7. **Heading tag : 6 types**

**H1 to h6 : h1 largest and h6 smallest**

**Attribute : Attribute is known as properties of tags.**

**Attribute we have to use in opening in the form of name-value pairs.**

**Value may be single or double quote or without also possible in html.**

**<tagName name1=”value” name2=’value2’ name3=value3>**

**</tagName>**

**Font tag : That tag is use to change color, style and size of contents.**

**3 attribute color, size and face**

**Hyper link tags**

1. **One page application : using internal hyper link or bookmark.**

**<a href=”#a1”></a>**

**<a name=”a1”></a>**

1. **Multi page application : using external hyper link**

**<a href=”pageName.html”>Text</a>**

**a : anchor tag**

**href : hyper reference**

**Image tags**

**<img src=”imageName.formatOfImage”/>**

**List tag**

**2 types**

**Unorder list**

**<ul>**

**<li></li>**

**</ul>**

**Ul : unorder**

**Li : list item**

**Order list**

**<ol>**

**<li></li>**

**</ol>**

**Ol: order list**

**Li : list item**

**Table tag**

**Employee Details**

**Id Name Salary**

100 Ravi 12000

101 Ramesh 14000

103 Lokesh 16000

**In React JS if we are planning to display the data using table tag we have to use thead and tbody mandatory otherwise we will get the warning message in React JS.**

**Form tags**

**Login Page**

**UserName TextField**

**Password Password**

**Submit Reset**

**Before HTML5**

**<input type=”text/password/radio/checkbox/button/file/submit/reset”/>**

**HTML forms methods**

**By default form consider as get method.**

**If method is get (by default get). The information of forms send through URL using url re-writing (query param) technique.**

**URL?key=value&key=value&key=value**

**If method is Get information send through URL**

**So data is not secure.**

**We can send maximum 255 character data only.**

**If method is get request body is empty.**

**If we can to secure data then we have to use**

**Post method**

**If method is post data send through body part of request.**

**So in post method data is secure.**

**Performance wise post method is slower than get methods.**

**SPA (Single Page Application )**

**Using Angular or React or Vue JS**

**Day 3: 15—5-2021**

**HTML4 : .xhtml**

**<!doctype html PUBLIC =”url.dtd”/>**

**document type definition**

**This file provide the structure of the HTML web page.**

**Root tag name**

**Many child tag head and body**

**Insider body how many p, b, table, tr etc many be optional.**

**HTML5**

**<!DOCTYPE html> giving the instruction to browser we are writing HTML 5 features(optional).**

**CSS : Cascading Style Sheet : CSS provide set of property and value which help to apply good look and feel for the web page.**

**Using CSS we can achieve separation of concern.**

**Actual content and formatting style separate.**

**3 type of CSS**

1. **Inline CSS**
2. **Internal CSS of embedded CSS**
3. **External CSS**

**Inline CSS**

**Syntax**

**<tagName style=”property:value;property:value;”></tagName>**

**Body, p, div, span, form, table etc**

**Internal CSS or embedded CSS**

**In head tag**

**<style>**

**selector {property : value;property: value}**

**</style>**

**Type of Selector**

**\* : Universal selector**

**\* { property : value}**

**tagName : specific selector**

**p {color:red}**

**tagName,tagName,tagName :**

**h1,p{color:red}**

**local class selector**

**tagName.className { property : value}**

**p.abc{color:red}**

**global class selector**

**.className {property : value}**

**.mno{color:green}**

**Id selector :**

**class selector Vs id selector**

**class : group of tags.**

**id : uniqueness for that tag.**

**Pease don’t provide two tag same id.**

**<div id=”d1”>**

**<p class=”abc” id=”p1”>First para</p>**

**<p class=”xyz” id=”p2”> Second para </p>**

**<p class=”abc” id=”p3”> Third para </p>**

**<p class=”xyz” id=”p4”> Fourth para </p>**

**</div>**

**DOM : Document Object Model :**

**External CSS**

**style.css**

**Selector { property : value}**

**Then in html page write this tag to connect the external css file.**

<link rel="stylesheet" href="style.css">

**CSS and CSS3 property**

**Font and text family property**

**Background property**

**Border :**

**border-top**

**border-bottom**

**border-left**

**border-right**

**border-width: 2px;**

**border-color : red**

**border-style : solid/dotted/**

**Padding :**

**padding-left : 5px**

**padding-right :10px**

**padding-top :5px**

**padding-bottom :10px**

**padding : 4px**

**Margin**

**Margin-left :**

**Margin-right:**

**Margin-top :**

**Margin-bottom :**

**Margin :**

**In html every tag or dom(Document Objet Model) element interlay follow box model concept.**

**Html, body, p, h1 to h6 tag.**

**Div tag default maring is 5px**

**Planning apply margin for p tag is 1px, 2px or parent tag margin then we have to inheritance.**

**<div >**

**<p></p>**

**</div>**

**Box Model**

**CSS3**

**Transform : rotate, skew, scale, translate etc**

**Transition property : CSS transition property allow us to change css property smoothly.**

**Margin,**

**Width,**

**Height,**

**Padding,**

**Font-size**

**Font-color**

**Etc**

**Transition :**

**transition-property : property-name/all**

**transition-duration : N s/ms**

**transition-delay : N S/ms**

**transition-timing-function : pre-defined-function-name**

**JavaScript Using ES5**

**var keyword to declare the variable**

**JavaScript using ES6**

**var, let and const**

**JavaScript ES5 and ES6**

**Babel**

**Day 4 : 16—5-2021**

**JavaScript was object based interpreter scripting language.**

**Object based(Prototype base) Vs Object oriented**

**JavaScript contains lot of pre-defined object or we can create user-defined object but not class concept till ES5.**

**Interpreter Vs Compiler**

**Scripting Vs Programming**

**Using JavaScript we can do programming on web page.**

**Syntax of JavaScript**

**<script type=”text/JavaScript”> opening tag**

**</script> closing tag**

**Type attribute is optional by default every script tag code JavaScript consider.**

**This script tag we can write in between head tag or body tag of html web page.**

**We can write more than one script tag in head or body part of html web page.**

**variable : variable is a name which hold the value.**

**To declare the variable in JavaScript we have to use the keyword as**

**var**

**syntax**

**var variableName;**

**DataType**

**var a; // default value is undefined**

**var n=10; // it consider number type**

**var m=10.10 //it consider number type**

**var name=”Ravi” // it consider string type**

**var res = true; //it consider Boolean type**

**var obj = new Date(); // it consider object type**

**Operator :**

**Arithmetic Operator : +, - , \*, /, %**

**Conditional Operator : >, >=, <, <= , ==, !=, ===**

**Logical operator : &&, ||, !**

**Assignment operator : =**

**Shorthand operator : +=, -=, \*=, /=**

**Increment and decrement operator : ++ , --**

**Pre and post increment or decrement**

**typeof operator or functions**

**if statement : it use to execute one or set of statement depending upon the conditions.**

**simple if**

**if(condition) {**

**}**

**if else**

**if(condition) {**

**}else {**

**}**

**Nested if**

**if(condition) {**

**if(condition){**

**}else {**

**}**

**}else {**

**if(condition) {**

**}**

**}**

**If else if or if ladder**

**if(condition) {**

**}else if(condition) {**

**}else if(condition) {**

**}else {**

**}**

**Switch statement : use can decide which block or set of code we want to execute.**

**Syntax**

**switch(variableName) {**

**case label1: block1;**

**break;**

**case label2: block2;**

**break;**

**case label3: block3;**

**break;**

**default : wrongBlock**

**break;**

**}**

**switch,case, break and default are keywords.**

**Looping : it is use to execute set of statement continuously till the condition become false.**

**While loop**

**Do while loop**

**For loop (classical for loop)**

**Initialization start and end**

**Condition true**

**Do the task**

**Increment or decrement**

**ES5, ES6**

**ECMA Script**

**European Computer Manufacture Association**

**ES is a concept.**

**JavaScript is a one of the implementation of ES5 or ES6 or ES7 etc.**

**Programming or OOPS Concept.**

**C,**

**C++, Java, Python**

**functions : ES5 style**

**function is use to write a set of instruction to perform a specific task.**

**2 types of function**

1. **Pre-defined function or global function**
2. **User-defined functions**

**Pre-defined function**

1. **alert(“Msg”) : This function is use to display alert pop message box.**
2. **prompt() :This function is use to receive the value from keyboard.**
3. **parseInt() : covert string to integer (without decimal)**
4. **parseFloat() : covert string to float ( with decimal)**
5. **eval() : convert string to number.**
6. **confirm() : it contains two button if we click ok return true and if click cancel return false.**

**do {**

**alert 1: Add, 2:sub, 3: Mul, 4: Div**

**using prompt receive the choice value.**

**eval for conversion**

**switch(variable) {**

**case 1: addition using x and y**

**break**

**case 2: sub using x and y**

**break**

**case 3: mul using x and y**

**break**

**case 4: div using x and y**

**break**

**default : wrong choice**

**break**

**}**

**confirm : take confirmation do u want to continue y or n**

**}while(variableName)**

**alert ( “thank you”)**

**Creating User-Defined functions**

**Normal Function style**

**Syntax**

**function functionName(paramterList) {**

**function body;**

**}**

1. **function no passing parameter and no return type.**

**Function declaration**

**function info() {**

**coding….**

**}**

**Calling function**

**info();**

1. **function passing parameter and no return type.**

**function add(a,b) {**

**}**

**add(100,200);**

1. **function passing parameter and return value.**
2. **no passing parameter but return the value.**

**Events**

**Event is an interaction between user and html component (DOM).**

**typeOfEvent**

**In JavaScript all event start with pre-fix on followed by event name.**

**on\***

**onclick**

**ondblclick : button or p,h1 etc.**

**onmouseover**

**onmouseout : div, image**

**onkeyup : a, b, c**

**onkeydown : key press in textfield , textfield, passwordfield, textarea**

**onfocus UPPERCASE, FORMAT OF DATA**

**onblur : textfied, passwordfiled, textarea**

**onchange : drowndown box.**

**Onload : when page loaded body tag, start service**

**Onunload : close the resources closing application or specific web page.**

**onsubmit : validation using JavaScript**

**etc**

**if we want to use the event we have to register the even on dom(html tags).**

**Event provide the bridge between HTML (Dom) and JavaScript code.**

**DOM : Document Object Model**

**DOM for HTML Web Page**

**DOM Hierarchy**

**index.html**

**<html>**

**<head>**

**</head>**

**<body>**

**<p>Welcome to Web Page</p>**

**</body>**

**</html>**

**Html**

**head body**

**p textNode**

**DOM API (Document Objet Model ) Application Programming interface.**

**Java, Python, C++, C# and JavaScript provided API means function or methods or classes or package or modules.**

**Which help to read, write and update the DOM element dynamically.**

**Reading text value using DOM API.**

**1st way : document.formname.componentname.value.**

**Day 5 and Day 6**

**JavaScript using ES5 and ES6**