**Day 1 : 05-03-2021**

**Git :**

Application Or Project

A Person Login module

B Person Customer module

C Person Account module

Merge code

Version Control tool

Git : Git is a version control system for tracking changes in file or folder or application or project and coordinates work of those file or projects among the multiple people.

Create folder : then create one or more than one file with simple message.

Git init : to create local repository

Git status :

Git add filename.extension

Git add a1.txt a2.txt a3.txt

Git add \*.txt

Git add .

Git commit –m “File created”

Git status

Git init

Then create the file

Git add .

Git commit –m “message”

Git remote add origin URL

Git push –u origin HEAD

Or

Git push – origin master

**Day 2 : 08-03-2021**

1. **Git init** : This command is use to create local repository.
2. **Git status** : This command is use to check the last command status.
3. **Git add . :** This command is use to add file or folder or project in staging area.
4. **Git commit –m “Message”** : This command is use to pass the files or folder from staging area to local repository.
5. **Git remote add origin URL :** This command is use to link local repository to remote repository.
6. **Git push –u origin HEAD/branchName**: This command use to pass the value from local repository to remote repository.
7. **Git clone URL** : This command is use to download the remote repository to local machine.

**Manager** 🡪 **Simple.txt**

**File created by Manager**

**Git init**

**Git add .**

**Git commit –m “File created”**

**Git remote add origin URL**

**Git push –u origin HEAD/BranchName**

**Default branchName –master/main**

**Raj**

Git clone URL

**Master/main**

**Ajay**

Git clone URL

**Master/Main**

**Branch** : Branch is like a pointer which hold more than one commit details.

1. **Git branch :** This command is use to check the all branch details.
2. **Git branch branchName :** This command is use to create the branch.
3. **Git checkout branchName :** This command is use to switch from one branch to anther branch.
4. **Git checkout –b branchName:** This command is use to create the branch and switch to created branch.
5. **Git branch –D branchName :** This command is use to delete the branch locally.

**Manager -- > Create main file**

**And push to remote repository**

**Raj Developer -🡪 clone remote repository**

**Create the file in Raj branch**

**Push raj branch to remote repository**

**Ajay Developer 🡪 Clone remote repository**

**Create the file in Ajay branch**

**Push ajay branch to remote repository**

**Create Folder TCSMEANStackTraining**

**Test file**

**Your Details**

**Remote Repository : EmpId\_YourName\_TCSMEANStackTraining**

**Web Technologies.**

**https://**[**www.google.com**](http://www.google.com) **🡪 URL**

**http/https🡪Req---🡪**

**Client Server**

**🡨-res(http/https)---**

**http/https:** Hyper text transfer protocol.

Protocol : set of rules which help to communicate more than one machine or device.

**www:** world wide web

**google :** domain or search engine.

**com** : commercial

**Uniform Resource Locator**

**HTML : Hyper Text Mark up Language.**

**It is use to display the contents.**

**CSS : Cascading Style sheet**

**Look and Feel or Presentation on Contents.**

**JS : JavaScript**

**Action on Contents**

**HTML : Hyper Text Mark Up Language : it is use to create the web page.**

**Web Page : It is use to display the contents in different format like normal, bold, italics, video, audio, clips etc.**

**Web Application : Collection of more than one web page.**

**Using HTML we can create static as well as dynamic web page.**

**Static : Display contents as it is on browser.**

**Dynamic : When user interact with web page event(action performed) generate.**

**Version**

**1, 2, 3, 4 and 5 : HTML5**

**Tags or elements.**

**Syntax**

**<tagName> opening tag**

**</tagName> closing tag**

**<tagName/> self closing tag**

**HTML is case insensitive.**

**HTML tags**

1. **Html**
2. **Head**
3. **Body**
4. **Title : This tag is use to display the message in title bar.**
5. **Paragraph tag : This tag is use to display the contents in browsing area.**

**This tag must in in between body tag.**

**<p> </p>**

1. **Break or br tag : This tag is use to break the content in next line. Break tag doesn’t contains closing tag.**

**<br>**

**<br/>**

1. **Heading tag : This tag is use to write the heading for paragraph or any contents.**

**There totally 6 heading tag**

**H1 : largest**

**To**

**H6 : smallest**

**Day 3 : 09-03-2021**

**Attribute : attribute is know as properties of a tags.**

**Syntax**

**<tagName name1=”value1” name2=’valule2’ name3=value3></tagName>**

**Attribute must in opening tag**

**In the form of key-value pairs.**

**Key may in single quote or double quote or without quote.**

**If value may be more than one world then it must be in single or double quote.**

**<p align=”center”></p>**

**<h1 align=right></h1>**

**Hyperlink :**

**This tag is use to connect external as well as internal (bookmark) contents.**

1. **External hyper link**
2. **Internal hyper link (book mark)**

**External hyper link**

**<a href=”pathOfFile.html”></a>**

**a 🡪 anchor**

**href 🡪 hyper reference.**

**Hr : horizontal line**

**<hr/>**

**Internal hyper link or book mark**

**<a href=”#a1”></a>**

**<a href=”#a2”></a>**

**<a name=”a1”></a>**

**<a name=”a2”></a>**

**Image tag**

**<img src=”NameOfImage.extension”/>**

**List Tag :**

1. **Unorder List**

**<ul>**

**<li>Raj</li>**

**<li>Ravi</li>**

**<li>Ramesh</li>**

**</ul>**

**UL : Unorder List**

**Li : List item**

1. **Order list**

**<ol>**

**<li>Java</li>**

**<li>Python></li>**

**<li>JavaScript</li>**

**</ol>**

**Ol : order list**

**Li : list item**

**Definition list**

**DL : definition list**

**DT : Definition term**

**DD : definition description**

**Table Tag :**

**Employee Details**

**Id Name Salary**

**100 Raj 12000**

**101 Ramesh 14000**

**<table>**

**<tr>**

**<th>Id</th>**

**<th>Name</th>**

**<th>Salary</th>**

**</tr>**

**<tr>**

**<td>1</td>**

**<td>Ravi</td>**

**<td>12000</td>**

**</tr>**

**</table>**

**Tr 🡪 table row**

**Th 🡪 table heading**

**Td 🡪 table data**

**Form Tag**

**Login, Application, Feedback Page, etc**

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

**By default HTML form method is Get consider.**

**If method is get information send through URL using URL rewrite technique.**

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

**Get is not a secure.**

**If you want secure then use method is POST**

**Get faster than post method.**

**VS Code :**

**Day 4 : 10-03-2021**

**HTML 4 version**

**<html public =”URL .dtd”>**

**<head>**

**</head>**

**<body>**

**</body>**

**</html>**

**Document type definition**

**HTML 5**

**<!doctype HTML> : Given the instruction to browser writing HTML5 version features.**

**VS Code**

**: Visual Studio Code :**

**MEAN Stack**

**Mongo DB Express Angular Node JS**

**CSS :**

**If we want to apply any formatting style for the contents we have to depending upon other tags.**

**As well HTML provided only few tags with the help of those tags we can’t apply good look and feel for web page.**

**Using HTML if we apply formatting style.**

**The content and formatting style combined together.**

**CSS : Cascading style sheet. CSS provide lot property in the form of key-value pairs which help to apply good look and feel for the web page.**

**With help of CSS we can achieve separation of concern.**

**Actual Content Formatting style**

**CSS file divided into 3 types**

1. **Inline css**
2. **Internal or Embedded CSS**
3. **External CSS**

**Inline CSS**

**Syntax**

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

**</tagName>**

**P,h1,h6,b,form,input,table, tr, td etc**

**Internal or Embedded CSS**

**In between head tag we have to write style tag**

**<style type=”text/css”>**

**Selector {property : value;property:value}**

**</style>**

**Types of selectors**

1. **Universal selector : \***

 \* {color: skyblue;}

1. **Specific selector**

**tagName {property : value}**

**p,h1,h6,div,span, b etc**

1. **Multi specific selector**

**tagName,tagName,tagName {property:value;}**

1. **Local Class selector :**

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

1. **Global class selector**

**.className{property:vaulu;}**

1. **Id selector :**

**#idName {property:value;}**

**Class selector Vs Id selector**

**More than one tag must be part of same class**

**Or**

**Class is a group of more than one tags.**

**Id is use to give unique ness for the tags.**

**<p class=”abc” id=”a1”>First Para</p>**

**<p class=”xyz” id=”a2”>Second Para</p>**

**<p class=”abc” id=”a3”>Third Para</p>**

**<p class=”xyz” id=”a4”>Fourth Para</p>**

**Div tag : Div tag is known as container tag. Which contains more than one other tags like p, h1, span, as well as another div.**

**External CSS File : CSS rules globally available for all web pages.**

**Font related property**

**Day 5 : 11-03-2021**

**CSS3 Property**

**CSS3 Transformation : Transformation allow you to translate, rotate, scale and skew html element (DOM (Document Object Model)).**

**CSS3 Transition : CSS3 Transition allow you to change CSS property value smoothly over a period of a time.**

1. **CSS property you want to add an effect to like width, height, color, size etc**
2. **Duration (time).**

**CSS3 Animation :**

**CSS3 animation allow most of HTML tags with JavaScript or Flash.**

**@keyframes : Which contains set of rules to execute the animation (ie start and end position for animation).**

**JavaScript**

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

**ES5 ECMA Script : European Computer Manufacture Association.**

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

**JavaScript tags**

**Syntax**

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

**</script> closing tag**

**This script tag we can write in between head tag or body tag or without any tags.**

**In one html page we can write more than one script tags.**

**Variable and data types.**

**Variable is a name which hold the value.**

**In JavaScript to declare the variable we have to use the keyword as var(up to ES5 JavaScript).**

**var variableName;**

**Data types : Data type is a type of data which tells that type of value it can hold.**

**JavaScript data types**

1. **Number (with or without decimal)**
2. **String : more than one character single or double quote.**
3. **Boolean : true or false.**
4. **Object reference.**

**Operator**

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

**2 Relational operator >, >=, <, <=, ==, !=**

1. **Assignment operator : =**
2. **=== :**
3. **Logical operator : &&, ||, !**
4. **Typeof operator or function**

**If statement**

1. **Simple if**

**if(condition) {**

**}**

1. **if else**

**if(condition) {**

**}else {**

**}**

1. **if else if**

**if(condition){**

**}else if(condition) {**

**}else if(condition) {**

**}else {**

**}**

**Switch statement**

**Syntax**

**switch(variableName) {**

**case label1: block1;**

**break;**

**case label2: block2**

**break;**

**case lable3:block3**

**break;**

**default : defaultblock**

**break;**

**}**

**Avg =80**

**if(avg>90) {**

**a++**

**}else if(avg>80) {**

**a**

**}else if(avg>70) {**

**b**

**}else {**

**c**

**}**

**Looping : It use to execute the statement again and again till the condition become false.**

**While loop**

**Do while loop**

**For loop**

**Initialization : start / end**

**Conditions true**

**Increment / decrement**

**styles.css (external CSS)**

**.className : global class selector**

**tagName.className : local class selector**

**.divFontClass{**

**font-size:**

**font-style:**

**font-family:**

**}**

**<div class=”divFontClass”>**

**</div>**

**<p></p>**

**<input type=”button”/>**

**<h1></h1>**

**<table>**

**Min 3 tag for 3 class selector (CSS class selector which contains min 3 css property).**

**Function and events**

**Function : it use to write set of instruction to perform a specific task.**

**2 types**

1. **pre-defined function**

**document.write(“Welcome to JS”);**

1. **alert(“welcome to JS”);**
2. **prompt(“Msg”): This function is use to receive the value from keyboards.**
3. **eval() : This function is use to convert string to number.**

**parseInt(): string to int**

**parseFloat(): string to float**

**eval() = parseInt() + parseFloat()**

1. **user-defined function**

**syntax**

**function functionName(parameterList) {**

**}**

**Event : Event is a interaction between user and html tags ( components) or DOM (Document Object Model).**

**DOM : All html tag is known as DOM elements. Like html, p, h1, b, form etc.**

**Event provide the bridge between html and JS code.**

**Type of events**

**In JavaScript all event start with prefix on followed by event name**

**onClick**

**onDblClick**

**onMouseOver**

**onMouseOut**

**onKeyUp**

**onKeyDown**

**onSubmit**

**onChange**

**onBlur**

**onFocus**

**etc**

**External JavaScript file**

**JSON : Java Script Object Notation**

**JSON is use to share the data between one application to anther application.**

**JSON is use to store the data in the form of key-value pairs. Where key is string type and value may be number, string, boolean, object reference type.**

**Object / JSON to String conversation**

**JSON.stringify()**

**String to JSON conversation**

**JSON.parse()**

**Storage objects**

**HTML5/JavaScript provide two types of storage object**

1. **sessionStorage : It hold the value till application close. Once application close the value get destroy from session.**
2. **localStorage : It hold the value in secondary memory after close the application also we can get next time open the application.**

**Using storage object we can share the value between more than one page or application.**

**One.js**

**sessionStorage.setItem(“key”,value);**

**localStorage.setItem(“key”,value);**

**two.js**

**sessionStorage.getItem(“key”)**

**localStorage.getItem(“key”);**

**sessionStorage.removeItem(“key”);**

**localStorage.removeItem(“key”);**