Eating, wiring

(javascript)

Bricks, paing

(CSS)

House Plan

(HTML)

house

Use of Firefox inspect tools

To clear the console clear() and enter

**Javascript code to change background at every click**

const htmlBody = document.querySelector('body');

const randomClickFunction = function(){

const colors = ["green", "orange", "grey", "red", "blue" , "green"];

const randomIndex = Math.floor(Math.random() \* colors.length);

const randomColor = colors[randomIndex];

htmlBody.style.backgroundColor = randomColor;

console.log('The user clicked and the color is set to ' + randomColor);

}

htmlBody.onclick = randomClickFunction

Simple button modifications and application of simple javascript

<!DOCTYPE html>

<html lang="en">

<head>

 <meta charset="UTF-8">

 <style>

  button{

   background-color: transparent;

   border: 1px solid navy;

   padding: 20px;

   font-size: 1.4rem;

   border-radius: 10px;

  }

  button:hover{

   background-color: navy;

   border: none;

   color: white;

  }

 </style>

 <meta http-equiv="X-UA-Compatible" content="IE=edge">

 <meta name="viewport" content="width=device-width, initial-scale=1.0">

 <title>Example</title>

</head>

<body>

 <button>CLICK ME</button>

 <div class="container"></div>

 <script>

  function onClickEvent(){

   const el = document.createElement('p');

   el.innerText = 'Clicked the button';

   document.querySelector('.container').appendChild(el);

  }

  document.querySelector('button').onclick = onClickEvent;

 </script>

</body>

</html>

**JavaScript Variables and Data Types**

const myArray = ["10", "20", "30", "40", "50"];

myArray

Array(5) [ "10", "20", "30", "40", "50" ]

myArray.push("70")

6

myArray

Array(6) [ "10", "20", "30", "40", "50", "70" ]

myArray.reverse()

Array(6) [ "70", "50", "40", "30", "20", "10" ]

Use console.log() to print something in console

One line comment //

Multiline comment /\* \*/

**Variables and datatypes**

Variable declaration and assignment

const firstVar = 'Amitha Shehan';

const secondVar = 50;

const thirdVar = {firstProperty: 'hello world'};

console.log(firstVar)

console.log(secondVar)

console.log(thirdVar)

declaring variable first and then assigning value to it

let newVar;

newVar = 50;

console.log(newVar)

**Accepted declaration and assigning**

Const var1 = 50; //cannot be reassigned/redeclared

Let var2 = 90; //used for reassignment

Var var3 = 50; // this is a legacy method. Do not recommend using this. Var can be redeclared

Use const when we have values not to change as the program runs

Use let when we assign new values to variables in the long run

Use capslock when defining const variables

const TAX\_RATE = 0.08;

**Use of const and function**

const variable1 = 10;

//const variable2 = variable1;

const variable3 = "Amitha";

const variable4 = {variableType:"object",variableValue:"somevalue"};

const variable5 = (function(){

return "Hello , my name is ";})();

const variable6 = variable5 + variable3;

**Boolean variables**

Const variable7 = false

**To get to know the datatype of a declared variable**

typeof variable1;

javascript is a dynamically typed language which means we do not need to define the variable type (number, string, Boolean) when we declare a variable.

Converting a string number and add it to a number

Both single quotes and double quotes can be used in string declaration. It is recommended to use single quotes

When assigning values to a number do not use commas

**Arrays**

Varieties of data can be included into arrays’

const array2 = [10, 'string 1', {prop:'sdkfjd'}, [10,20,330]];

console.log(array2[0]);

console.log(array2[1]);

console.log(array2[2]);

console.log(array2[3]);

console.log(array2[3][2]);

**Objects**