**06/11/2022**

**Hitesh sir**

1, Fast coding in html-!+tab

2.Head is not visible this is internal attributes

3. H1,H2in body

4. P –lorem200

5.<img src=’’”>

6. download image unsplash.com

7.Link another page=<a href=” <https://google.com>.”> go to google</a>

8. for breack the linked line-<h2><a href=”<https://google.com>”.> go to google</a>

</h2>

9. <a href =”<https://google.com>”.> <h2>go to google</h2></a>

10. for comment on multiple line(select all ctrl +/)

11. create multiple curson(option key+alt)

12. right click on output and open source code

13.to breack paragraph (<pre></pre>)

Page-heading,paragraph,formating,citation,image

14. Formating tag= <Strong></strong>,Small,em, <p>H<.sub>2</Sub>O</p>.,ins,del,mark

15.Citation tag= blockquote, q,cite

16 UAE website=<.bdo dir=”rtl”> human</bdo>

17.image(image-map.net)

18.Documentation-MDN

**Anurag sir**

1.Tables-school time table-tr,th,td

Assignment-time table, signleborder

2.Iframe=link youtube videos

3.research on audeo and video

4.forms(name,password,email,dob,submit,reset,radio,checkbox,button)

5.Input types

6.semantic (header,footer,section) more on semantic

**12-11-2022**

1. Css use to change background color, body ,paragraph, header
2. Selectors

a>Universal selector

b>Individual selector

c>ID selector

d>and selector (li.bg-black, text-white)

e>combined selector (span, li{background –color:burlywood})

f>inside selector (div ulli {background-color:4d4d4d})

g>Direct child selector

h>sibling selector

i>Pseudo selctor-::Before-::After

>> uicolorpicker.com for color

>> warning, Dangers

>>getbootstrap.com

>>css unminify

**13-11-2022**

1.Pseudo selector :hover::befor, :: after

2.Content,display,width,hirght,border radius, border

3.button, padding ,border radius,hover,border,bgc color,color

**Anurag sir**

4.:hover,:visited,:link

5.input:focus

6.li:nth - child (even,odd,5n+1,2n){

Bgc color

}

7.li:first –child, last-child

8.<li style=”color:yellow”>ten</li>

9.Margin is from outside and padding is from inside(Padding sequence is –top,,right,bottom,left)

10.list-style-type:circle,square;

11.Position-Static, relative,fixed,absolute,sticky

12.Float-right,left,none.inherit

Homework

1.when clicking on input box then size should be change

2.Padding

**19-11-2022**

1.write article as much as possible

2.Position (static,fixed,sticky,relative,absolute), flex, play games

3.CSS tricks.com

Pos{

Border:

Display:

Height:

Width:

Text-align:

Border-radius:

}

#three{

Pos: relative;

Top: 50px

Right:10px

}

4.excalidraw

**Flex**

**5. .container{**

Display:flex

Height:

Border:

Flex-direction:(row|row-reverse|column|column-reverse)

Justify-content: (flex-start|flex-end|center|space-between|space-around|space-evenly)

Flex-wrap: (nowrap|wrap|wrap-reverse)

Align-item: (flex-start| flex-end | center | stretch)

Gap: (10px)

**}**

**6. #flex {**

Flex-grow:2

Align-self: center

**}**

**7**. flexbox froggy.com (complete all 24 of 24)

Home work

1. Write article on flex box and its properties

**20-11-2022**

1.Position: Relative,Absolute,sticky,static,fixed

3.Grid,

4.media

@media (max-width:500px){

Body{

Background-color:

Display:

Flex-direction:

Justify-content:

Align-item:}

}

5. Grid: Grid container, grid cell, grid line

6. grid-template-column, grid-template-row

7. 1fr (1 fraction of entire width)

8.row-gap, column-gap

9. grid-row-start: 1

Grid-row-end: 3

Grid-column-start: 1

Grid-column-end-3

10. grid games

Assignment: Position, flex, grid, media

Next session topic: a>tailwind

b>bootstrap

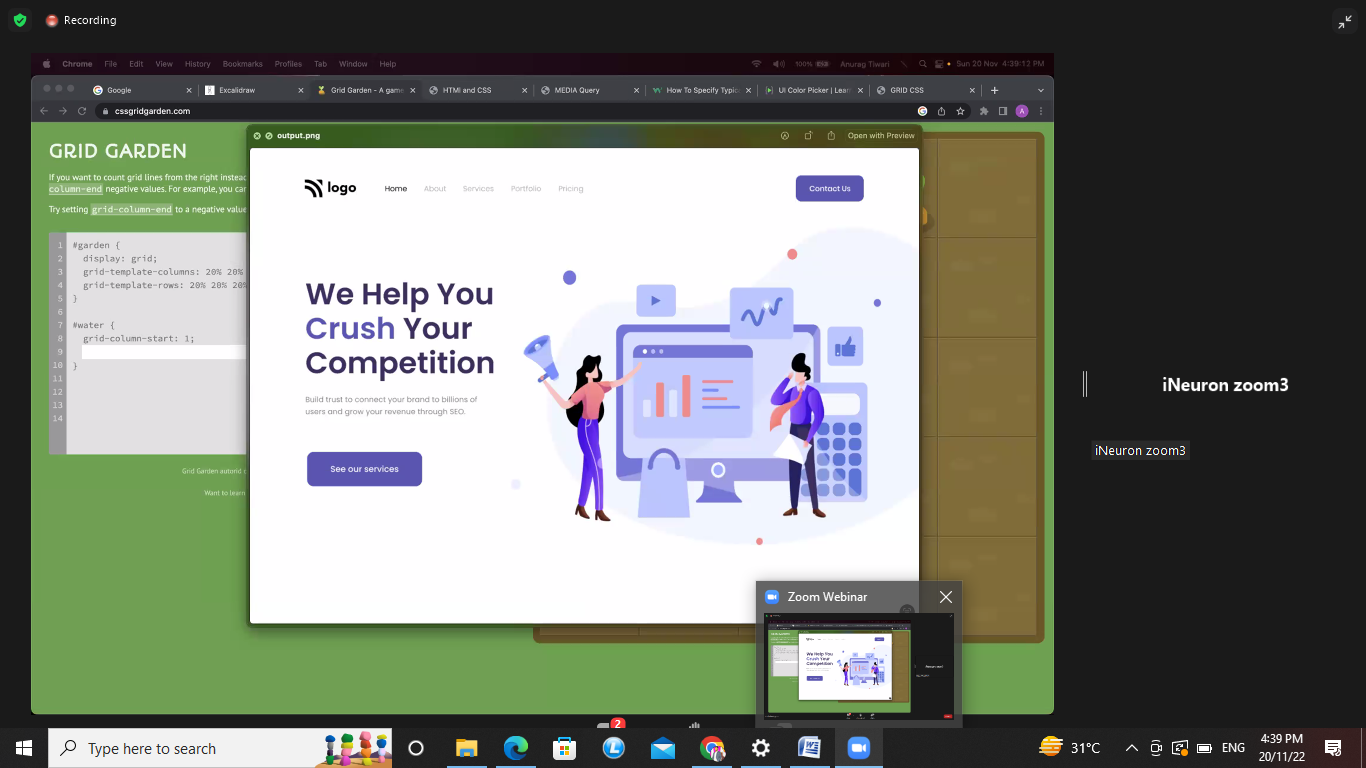
c>git and github

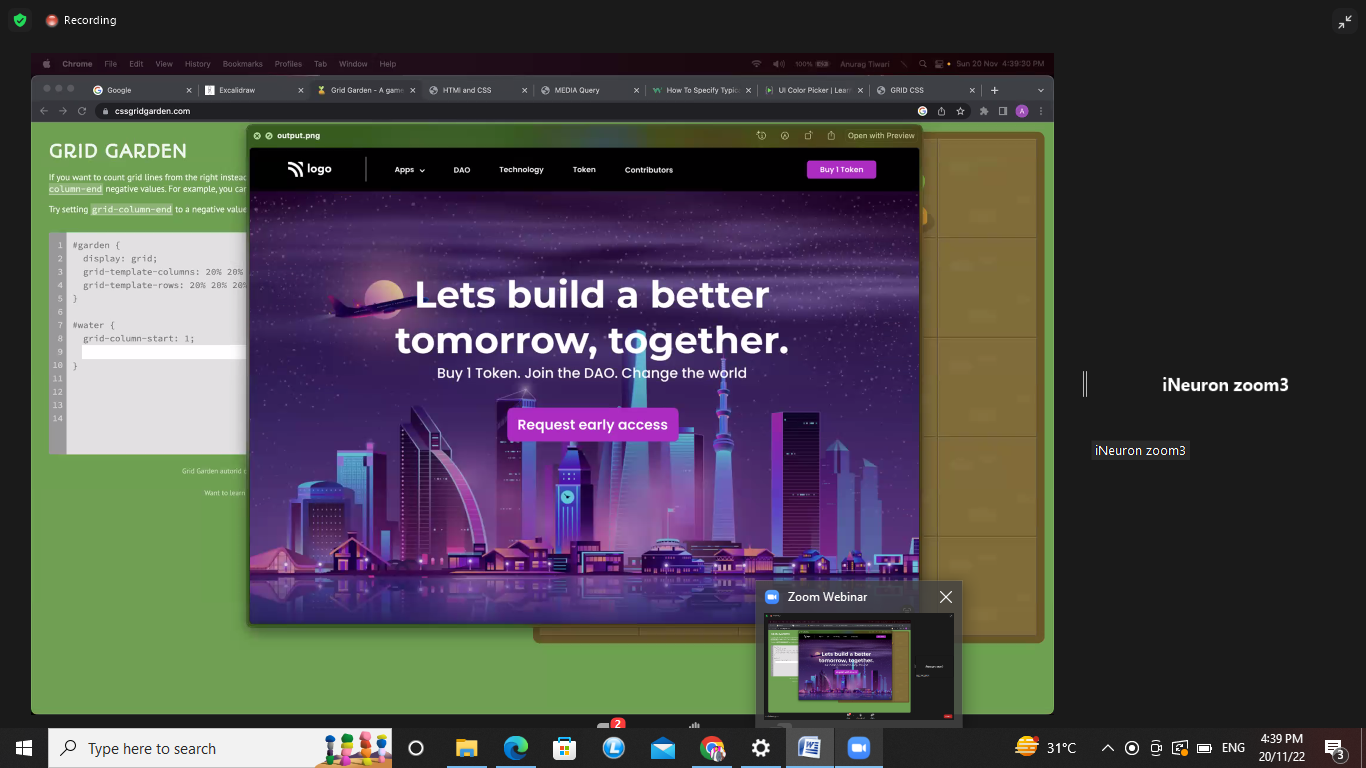
d>markdown

11.Rivision of all html and CSS topic covered

**Project**

Project: complete three project

****

****

**26-11-2022**

1.create whole website after complete this course

2. Project-1

Semantic – nav bar

Style.css – it will be linked to html file

Link.css

Font weight-500

Home work- margin, padding between the image

Project-2

Navbar

Main section

Left section

Right section

\*{

Margin:0

Padding:0

Box-sizing : border-box?

}

Body{

Margin: auto?

Overflow :hidden

}

(z-index:-1): it will keep the page behind the page

**27-11-2022**

Tailwind

1.cdn link

2.mterial css

3. Tailwind utility first CSS

4.Tailwindcss.com

5. Transition?

6.Core concept –read

7.Code.visualstudio.com –clone assignment

8.Netlify.com/ Vercel.com

**03-12-2022**

1.Readme.md= conclusion- This file is restored on repository and used to communicate related to your project

2.Md=markdown – it is information related to project

3.Repo-repository

# is pound symbol

Italic (-Batch-)

Bold (\*\*Batch\*\*)

~ is called till day ~~9999~~ Rs.10000.0/-

Underline

[] (put link here)

[] (link ‘title’)

Image

![] (image path)

Tables

||||

` ` is called back tick

```javascript

```

>Is called quotes

List 1. 2. 3.

* For bullet point

Githb is used for storing your files

1. Git is version controlling system
2. Github is simple tool which work on git

First install git and config

~ git –version (enter)

~ git config –global user.name “2azahar786”

~ git config –global user.email “[2azahar786@gmail.com](mailto:2azahar786@gmail.com)”

~ git config –global –edit

Esc : wq( for back)

First time in terminal push

Git –version

Git init

Git ls-a

Clear

Git status

Git add.

Git commit –m “file name”

Git branch –M main

git remote add origin https://github.com/2azahar786/FSJS2.git

git push –u origin main

when file added

git status

git add.

Git commit –m “file name”

Git push

When changes in file

Git status

Git add.

Git commit –m “changes name”

Git push

**10-12-2022**

History and basic of the javascript

// History

// Javascript came in 1995 name as "mocha"

// Its changed in 1996 name as "live script"

// against its changes in 1999 name as javascript and this as till Date

// why need java Script

// that time there were two browser first is "internet explorer" and 2nd is "netskip"

// which is supported only html and css File

// the browser get some shorts of functinality so for the solutions of that they created scriting langues for that

// programmer were creating own script so it was very dificullt inderstand for other programer.

// Echmascript organisation came into the pictures who set certain rules for the scripting langues which will be

// applied to all scripting langues. it set the rules and becomes midified and in 2015 gets satndard rules name as ES6

// 6 pillors of any programming langues

// 1. Values 2. operation 3. variables

// 4. conditions 5. loops6. functions

// Values

// 9;

// 3.5;

// these are the number values

// Azhar;

// Ansari;

//  These are the string values

// true; (1)

// False; (0)

// these are the Boolean values(Itwas implemented by the famous sceintis name as george boole)

// null;

// undefined;

// These are te empty values

// These all are the premitive values (Premitive has a single value)

// Array And Object

// [1, 'azhar', 'ansari', 85, true, 65, 8] (in squqre brackets are aaray and it should be Number, string, boolean)

//  0,  1,         2,      3,   4,   5,  6 - index value tells the place of that aaray value

// {

//  firstname:Azhar

//  lastname:Ansari

// }

// here are the key and values, key will be your choice.these are the object and it should be always in curly brackets

// this is the non premitive data type= has more than one value

// Variables

// var (this will be taught later)

// let

// const

// let x=20; (this will be cahned any time)

// const bankaccountno='45236178933155' (this cant't be changed never)

// let firstname='azhar';

// let lastname='ansari';

// const accountno=45789611233;

// console.log( 'My first name is'+firstname);

// console.log('My last name is',lastname);

// console.log(accountno,firstname,lastname);

// Template Literal

//  console.log(`My current first name is ${firstname} ${lastname} ${accountno}`);

// Operator

// Assignment operator

// let x=10;

// let y=12;

// Arithmatic operator

// +, -, \*, /, (%=> remainder)

// let z=x+y;

// console.log(z);

// console.log(x+y);

// console.log(x-y);

// console.log(x\*y);

// console.log(x/y);

// console.log(x%y);

// comparision operator

// let abc=20;

// let xyz=25;

// console.log(abc>xyz);

// console.log(abc<xyz);

// console.log(abc<=xyz);

// console.log(abc>=xyz);

// == or ===

// let az=10;

// let an='10';

// console.log(az==an); (this will compare only value)

// console.log(az===an); (this will compare data types)

// Interview question

// let y;

// y=10;

// console.log(y);

// const account=456321;

// // account=56324;

// console.log(account);

// conditions

// let age=31;

// if ( age==18)

// {

// console.log('you are allowed to the park');

// }

// else if(age<=21){

//     console.log('You are welcome');

// }

// else if(age>=30){

//     console.log('you are most welcome');

// }

// else{

    // console.log('you are not allowed in the park')

// }

**11-12-2022**

Basic of the Javascript

// Conditions

// if else

// switch case

// ternary operator/ tarnary conditions

// Rating mini app

// let rating=4;

// if (rating==5){

//     console.log('5 stars');

// }

// else if (rating==4){

//     console.log('4 stars');

// }

// else if(rating==3){

//     console.log('3 stars');

// }

// else if (rating==2){

//     console.log('2 stars');

// }

// else {

//     console.log('I am else');

// }

// switch

// let user ='admin';

// switch (user) {

//     case 'admin':

//     console.log('Redirect him/her to admin page');

//     // break;

//     case 'mentor':

//         console.log('Redirect him/her to mentor page');

//         // break;

//     case 'principal':

//         console.log('Redirect him/her to principal page');

//         // break;

//         default:

//             console.log('I am simple user');

// }

// Ternary Operator / Ternary conditions

// conditions ? true : false

// let userlogin = false;

// userlogin ? console.log('logout') : console.log('login');

// condition ?  true  : false

// let name = [

//     'Azhar',

//     'deepak',

//     'Sumant',

//     'Serajul',

//     'Rakesh',

//     'Dhananajay',

//     'Nooralam',

//     'Tipu',

// ];

// console.log(name);

// console.log(name[7]);

// Object

// let bankaccount = {

//     firstname : 'Azhar',

//     Lastname : 'Ansari',

//     mobileno : 7506976882,

//     balance : 9999999,

// };

// console.log(bankaccount);

// console.log(bankaccount.balance);

// Date and Math

const now = new Date();

// console.log(now);

// console.log(now.toString());

// console.log(now.getFullYear());

// console.log(now.getMonth());

// console.log(now.getDay());

// console.log(now.getUTCMonth());

// console.log(now.getTime());

// console.log(now.getHours());

// console.log(now.getMinutes());

// console.log(now.getSeconds());

// Math

// const  PI = Math;

// console.log(PI);

// console.log(Math.PI);

// console.log(Math.round(PI));

// console.log(Math.round(11.2));

// console.log(Math.floor(10.5));

// console.log(Math.ceil(10.5));

// Min and Max

// console.log(Math.min(5,3,10,8,44,10,11,6,));

// console.log(Math.max(44,22,5,6,55,88,21,28,29,101));

// Random

// console.log(Math.random());  // 0 - 0.999999999 / (0 - 1)

// console.log(Math.round(Math.random()));

// console.log(Math.round(Math.random()\*51));

// console.log(Math.pow(3, 3));

// console.log(Math.pow(2, 4));

// console.log(Math.log(3));

// // console.log(Math.log(9));

// console.log(Math.sin(90));

// console.log(Math.cos(45));

// Loops

// Do while, While, for

// let i=5;

// do {

//     console.log('value of i is :', i);

//     i++;

// } while (i<=8);

// let j=8;

// while (j<=12){

//     console.log('value of j is', j);

//     j++;

// }

//  for (let i=2; i<=8; i++){

//     console.log('value of i is ', i);

//     console.log(i);

//  }

17-12-2022

// Array in depth and for of loops

// Array method

// let names =['Azhar', 'Deepak', 'Imran', 'Sumant', 'Tipu'];

// console.log(names);

// console.log(names.length);

// console.log(names[2+2]);

// console.log(names[names.length-2]);

// names[2] ='Ansari';

// console.log(names);

// let namess = new Array('name1', 'name2');  //a way to declare the array// not record

// console.log(namess);

// Push => insert new values into the array

// names.push('Cash');

// console.log(names);

// slice

// let names =['Azhar', 'Deepak', 'Imran', 'Sumant', 'Tipu'];

// console.log(names.slice(1, 3)); //3 is the excluded value not included value

// // // splice

// let fruit = ['Apple', 'badaapple', 'chota Apple', 'double apple'];

// fruit.splice(1, 3 ,'accha apple', 'kharab Apple');

// console.log(fruit);

// Concatenation

// let arr1 = [1,2,3,4,5];

// let arr2 = [4,5,6,7,8];

// let arr3 = [7,7,7,7,7,7,7,7,7,]

// console.log(arr1.concat(arr2));

// console.log(arr1.concat(arr2, arr3));

// fill

// let arr4 = [1,2,3,4,5,6,7,8,9];

// arr4.fill('Azhar', 1, 6);

// console.log(arr4);

// Include

// let num =[1,2,3,4,5,6,7,8];

// console.log(num.includes(7,));

// indexof

// let num =[1,2,3,'Azhar', 5,8,9,7,'Azhar', 'Azhar'];

// console.log(num.indexOf('Azhar'));

// isarray

// let num =[1,2,3,'Azhar', 4,5,8, 'Azhar'];

// let num1 ='Azhar';

// console.log(Array.isArray(num));

// console.log(Array.isArray(num1));

// Join

// let arr1 =[1,2,3,4,5,6];

// let var1 =arr1.join('');

// // console.log(var1);

// console.log(arr1.join(''));

// console.log(typeof arr1);

// keys

// For of

// lastindexof

// let num =[1,2,'Azhar',3,4,5,6,'Azhar'];

// console.log(num.lastIndexOf('Azhar'));

// console.log(num.indexOf('Azhar'));

// console.log(num.indexOf(5));

// Map

// let maths =[1, 4, 9, 16, 25,];

// console.log(maths.map(Math.sqrt));

// Pop

// let maths =[1,4,9,16,25];

// console.log(maths.pop());

// console.log(maths);

// reverse

// let maths =[1,4,9,16,25];

// console.log(maths.reverse());

// Shift

// let maths = [88,'Azhar','Ansari',1,4,9,16,25];

// console.log(maths.shift());

// sort

// let names =['Azhar','Sumant', 'Imran', 'Deepak','Tipu'];

// console.log(names.sort());

// console.log(names.reverse(names.sort()));

// unshift

// let fruit =['Apple','badaapple','chhota apple', 'doubleapple'];

// fruit.unshift('apple01','apple02');

// console.log(fruit);

// converting to array

// let name ='Azhar';

// let array1= name.split("");

// console.log(array1);

// for of

// let fruit =['Apple','badaapple','chhota apple', 'doubleapple'];

// let upperfruit =[]; //empty array

// for (let Azhar of fruit){

//     upperfruit.push(Azhar.toUpperCase());

// }

// console.log(upperfruit);

// Break and continue

// continue

// for (let i = 0; i<=5; i++){

//     if (i==3){

//         break;  //skip omit

//     }

//     console.log(i);

// }

18-12-2022

// WelCome to JavaScript Lecture!

// Stage One will Be completed Today!

// Datatypes => number, string, boolean, Null and undefined.

// Array and Object

// OPerators

// Condtions

// If, If else, Switch and Ternary Cond

// Loops => do while, while loop and For Loop

// Math and Date

// Array and method => Assignment to write article

// for of  => Array and for in => Object

// Function=> block of function , when require we call that and that perform the action and return the result

// Topic : Function

// function sum(value1, value2) {

    // definition of function

    // let num1;

    // let num2;

    //  console.log('addition of the two value', value1 + value2);

// }

// sum(120, 25); // calling of the function

// sum(12, 20);

// sum(45, 40);

// sum(21, 30);

// sum(26, 35);

// sum(18, 31);

// sum(28, 39);

// sum(15, 45);

// Normal function and funtion with parameter

// function with return type

// function sum(val1, val2){

//     let result= (val1 + val2);

//     return result;

// }

// let add = sum(15, 30);

// console.log(add);

// function subtract(val1, val2){

//     let result = (val1 - val2);

//     return result;

// }

// let minus = subtract(20, 10);

// console.log(minus)

// function multiplication(val1, val2){

//     let result = ( val1\*val2);

//     return result;

// }

// let multiply = multiplication(10, 5);

// console.log(multiply);

// function division (val1, val2){

//     let result = (val1 / val2);

//     return result;

// }

// let devide = division(50, 5);

// console.log(devide);

// function module(val1, val2){

// let result = (val1 % val2);

// return result;

// }

// let mod = module(100, 10);

// console.log(mod);

// String

// function URL (url, domain){

//     let con = 'https://';

//     let result = (con + url + domain);

//     return result;

// }

// let site = URL('azharfirstproject', '.netlify.app');

// console.log(URL('azharfirstproject', '.netlify.app'));

// console.log(site);

// function sum(arr){

//     let sum =0;

//     for (let i=0; i<arr.length; i++){

//         sum = sum + arr[i];

//     }

//     return sum;

// }

// let sumarray=[1,2,3,4,5,6,7,8];

// let add = sum(sumarray);

// console.log(add);

// function azhar (){

//     // arguments

//    let sum =0;

//    for (let i=0; i<arguments.length; i++){

//     sum = sum+arguments[i];

//    }

// return sum;

// }

// console.log(azhar(10,11,45,23,88,75,));

// arrow function

// let azhar = (val1, val2) => {

//     console.log('mai bhagwan nhi hun',val1 , val2);

// };

// azhar(1, 2);

// let azhar = () => {

//     console.log('kuch bhi');

// };

// azhar();

// object

let username ={

    firstname :'azhar',

    lastname:'ansari',

    role: 'admin',

    logincount:25,

};

// console.log(username);

// console.log(username.role);

// username.role = 'metor';

// console.log(username.role);

// console.log(username);

// for in

for(let x in username){

    console.log(username[x]);

}

07-01-2023

1.Object in javascript

2.execution Engin

3. Rendering engine

4. Object MDN read only description

5. Revise object mdn

6.Enumereates

7.Browser inspect console

08-01-2023

1.Prototypes

2.Inheritence

3.DOM manipulation

4.Array can inject from prototypes

5.Inheritence

6. \_\_Proto\_\_

7.modern system

8.object.setprototypes

9object mdn in script

10.create free account on AWS

11 To find true length of the string

String.prototype.truelength= function(){

    console.log(`True length is : ${this.trim().length}`);

}

12. for adapting other object to another object

Object.setPrototypeOf(TeachingSupport, Teacher)

21-01-2023

1.assignment create the blank webpage and on click should be create circle

2. Event in javascript

3.en.m.wikipedia

4.document.queryselector(H3)

5.Nodelist – not a javascript API/ It is a browser API

6.innertext/innerHTMl

7.allh3.map(=>INNERtEXT)

8.Alllh3[0].prototype.innerHTML

9.Convert node to array

10.array.form(allh3)

11. […allh3]

12.Dom –Document object Model

13. By tag name/ By tag/ QuerySelector

14. Alert | confirm | Prompt

22-01-2023

1.document.window

2.` ` Template Literal

3. document.body.appendChild()

4.all addEventListener (click,dblclick,mouseenter, moveover, input, keydown, keyup, keypress, onload)

5. document.designMode = “on”

6.Change your with your name on google (document.queryselector(“ #...”).innerText = “Azhar”)

28-01-2023

HOF AND CALL BACK

1. function is calling  a function

1. Higher order functions are functions which take other function as a parameter or return a function as a value. The function passed as a parameter is called callback.

1.. forEach() is HOF, the () => {} inside it is callback

2.map – map something to each value

3.filter – filter something from the array

4 reduce - Reduce takes a callback function. The call back function takes accumulator, current, and optional initial value as a parameter and returns a single value. It is a good practice to define an initial value for the accumulator value. If we do not specify this parameter, by default accumulator will get array first value. If our array is an empty array, then Javascript will throw an error.

5. foreach, map, filter, reduce, sort, find, some, every

6. desctructuring and spread and rest (...spread)(...rest)

29-01-2023

1.Rest and spread(both are denoted by (...))

2.chunk – multiple data together

3.(...args) holding rest operator

4.spread mostly used in react

5. Try and catch

6. throw

7. there are three types error in javascript – reference error, syntax error, type error

8. Promises devided in three category – (Pending, resolved , Rejected)

9. event loop

10.games =www..com

04-02-2023

1.Revision of basic javascript

2.Node.js is a runtime to write javascript

3. ryan Dahl invented node.js(runtime)

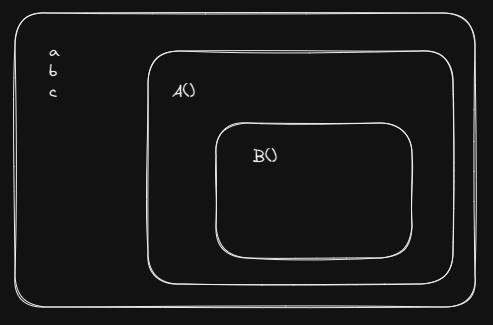
4.javascript engine = v8

5.v8 is called also Libuv

6. Scope and callstack

7. var alway store in global

8. let and const alway go to script (block level) (temporal dead zone)

9.Lexical scope 

10.Datatype : Premetive and nonpremetive

11.Premetive = Single value

12.nonpremetive = multiple value (Array and objects)

13.Use strict = this is use at first line of code to aoid the mistakes in javascript coding

14.typeof

15.Template literal (` ` ) ( Template string)(javascript interpolation)

16.operator (arithmetic operator, assign operator, comparision operator,)

17. condition (if alse, switch and ternary condition)

18.tarnary condition(? True : false)

var hello = “hey”;

hello = “hey” ? console.log(“FSJS”) : console.log(“Azhar”)

19.array= index and value (fixed index)

20.object = key and value (key can change)

21.let names = [ ]

Let namesss = new array ( )

22. all method coved =(push, pop, slice splice, shift, unshift, fill, includes, indexof, join, keys, lastindexof, reverse, concat, sort, split, trim, reduce, filter, math, date, reduce, rest, spread)

23. loops (while, do while, for loop, for of and for in)

24.function = (normal function,

Function with return type and parameter

Anonymous function

Expression function

Self invoking function)

// Variable

// Datatype

// Primitve  => Single Value

// Number, String, Boolean, null , undefined

// Non-primitive => Multiple Values

// Array and Object

// Everything in Javascript is OBject

// "use strict";

// typeof

// template LIteral(Template String)

// Javascript Interpolation

// let nameofvaribale = [1, 2, 3, 4];

// console.log(` Anurag ${nameofvaribale}`);

// console.log(typeof null);

// OPerators

// =>

// Conditions

// if else

// switch case

// Ternary Con

// let Bankaccount = 500000000;

// if (Bankaccount == 500000000) {

//   console.log("Material thingss");

// } else if (Bankaccount){

//     console.log("Alto");

// }

//  else {

//   console.log("I am Happy");

// }

// if else

// Switch Case

let user = "Human";

// switch (user) {

//   case "Human":

//     console.log("Hello World");

//     break;

//   case "insaan":

//     console.log("Hey World");

//

// }

// Ternay Operators and Few Call it as Ternay Conditon

// ? True : False

var hello = "he";

hello == "hey" ? console.log("FSJS") : console.log("Anurag");

// Array

// index and Values (Fixed Index)

// Object

// Key and value (Key Can Changed)

let names = [

  "Anurag",

  "Anirudh",

  "VYOM",

  "Varun",

  "NIshand",

  "Jaswant",

  "Pradip",

];

// console.log(names);

// let namesss = new Array(

//   "Anurag",

//   "Anirudh",

//   "VYOM",

//   "Varun",

//   "NIshand",

//   "Jaswant",

//   "Pradip"

// );

// console.log(namesss);

// length

// console.log(names.length);

// Push

// Pop

// Slice

//Splice

// shift

// unshift

// fill

// includes

// indexof

// join

// Keys

// lastindexof

// reverse

// concat

// sort

// trim and split

// Math

// Date

// loops

// while, do while, for loop

// for of and for in

// const number = { 1: "1", 2: 2, 3: 3, 4: 4, 5: 5 };

// for (const num in number) {

//   console.log(number.num);

// }

// Function

// Normal Function

// function with Return type and Parameter

// Anonymouse Function\

// Expresssion Function

// Self Invoking Function

let userDDetails = {

  firstName: "Anurag",

  lastName: "Tiwari",

  courseList: [],

  website: "anuragtiwari.me",

  buyCourse: function (courseName) {

    this.courseList.push(courseName);

  },

  getCourseCount: function () {

    return `${this.firstName} is enrolled in ${this.courseList.length}`;

  },

};

// console.log(userDDetails);

userDDetails.buyCourse("FSJS");

console.log(userDDetails.getCourseCount());

05-02-2023

1.GEC-Global Execution Context

2.Hoisting (javascript engine first start the scanning and after start the execution this process called the Hoisting)

3.call stack (Play to keep everything in memeory)

4.Lexical scope/environment (function depend on function is lexical environment)

5.Lexico graphy = kisi ke sath

6. Clouser = it outer function accessed by the inner function in that case outer function is clouser

7.HOF (higher order function is function which take other parameter and return a function)

Function A (a){

HOF

}

Function a(){

Callback }

8. functional programming(definition – we don’t have to write regular loop and function)

arr.foreach( () => { } )

9. map, filter, reduce

11/02/2023

1.Weather App

2. map:interate over array and perform a action

3.filter:iterate over the array filter out the element.

4. reduce: accumulate of the data currect initiat.

5.promise . then() , Cathc()

7. const u=URL = “https:...........”;

Fetch (URL)

.then(() =>)

.then(() =>)

Catch(() =>)

8. go and createweather API

9.weatherqpi.com

12/02/2023

1.React : Library which having few thing

2.Nodejs : it is framework which is having everything

3.react is a javascript library for building user interface. React is created by Jardon walke < implemented in 2013 and handelled by facebook

4.JSX : it htmlelement in javascript with variable ( const names = <p>welcome to World</p>)

5.babel : it is Transpiler which convert JSX to javascript and return new version of js over oldest

6. (const names = <p>Azhar Ansari</p> =>JSX)=> jsx element

7.react fragment ( <div> </div>)

8.In react project there will be only one single file as index.html

9.babel cdn

10.reactDOM.render(what, where)

11. component based

12. classis reserved in javascript so we use className in javascript

13.NPM : It is installed globally

14.NPX : It can be installed particularly place

15. Node module : It holds all the thing related react app

16. web screepling

18/02/2023

1.npm –(node package manager)

2.npx –(node package executer)

3.react app terminal install package

4.npx create-react-app ppt-basics (for creating new react app)

5. npm start (it is used after npx run)

19/02/2023

1.Create new react project(npx create-react-app ppt-basics)

2.ReactDOM.createroot(Document.getelementbyid(“...”))

3.React.strike-mode

4.react documentation

5.www.npmjs.com/avios

6.randomuser.me/api

7.JSON Formator

8.useeffect and usestate

9.try grab API assumption

25/02/2023

1.React three cdn – babel , react and react DOM

2.ReactDOM.render(what, where)

3.create react file (npx create-react-app ppt)

4.RFC

5.componenet will be always capital

Preplacement Team

Resume preparation

Soft skills training

Tech skills training

Problem solving techniques

Mock interview and feedback

Ineuron internship project

Building resume

26/02/2023

1. SPA – Single page application

2. React-router-Dom

3. component pro

4. props(something going to change props)

5.use state and useeffect

6Homework – make pwskills website

04/03/2023

quick revision

1.why come react

2.componenet & props

3.data & state has been changed by using react

4.hooks in react

5.usestate: which is going tohold something

6.for new react app (mkdir saturday

cd saturday)

7.use context router

8.npx create-react-app appname

9.import react,{usestate} from “react”;

05/03/2023

1.mkdir conapi | cd conapi

2.npx create-react-app cont

3..Context API : (Pass message directly)

4.context has three thing ( create context, provider, consumer)

5.rfce

6.react-router-DOM

7.react js, usestate, useeffect, context api, router DOM

11/03/2023

1.usestate management ( Just manage the state of the component)

2.Redux Toolkit

3.React Redux

4.npx create-react-app

5.Tailwind commond

6.npm 1 -D tailwind / npx tailwindcss init

7. rfce

8.npm install

9.payload

10.increment/ decrement

12/03/2023

1.Redux toolkit

2.todo app

3.add todo

4.remove todo

5.read more documentation of redux

18/03/2023

\*\*Backend of JS

1.How internet works?

Request

Response

URL query (<https://google.com/serach>)

3. ? = query string begins

4.q= ineuron (key value pair)

5.&= separator

6.% = also separator

\*\* What is API?

1. Application programingInterface
2. Domain.com/login?username = azhar & pass =....
3. API= string (which will communicate between client & server)
4. API = It is link which carry certain information or data

\*\* NodeJS Architecture

requests

NodeJS server

Event Queue

Event Loop

Thread Pool

External Resources

\*\* server

1. Listen
2. Local Host (Domain)
3. PORT

65535 last server PORT

1. Appache – web server /PHP

\*\*Application Workflow

\*\* Database

SQL No SQL

MySQL MongoDB

1.MySQL = relational Database

2.MongoDB = Non relational Database

3. React Node

Axios

URl connection string

MongoDB

Netlify axios Railway MongoDB

(URL)

\*\*MVC Architecture

Mode view & Controller

Middlware

Password Hosting

Ebgbrctde14252638495

bcrypt

Abcd2345

Password has li function Has H password

(hide information)

\*\*Storing Media

1. Cloudnary/ link/ link store to database

19/03/2023

HTTP1 | HTTP1.1 | HTTP2.0 | HTTP3

1996 | 1997 |2015 |2022 JUN

CLIENT SERVER

NON PERSISTENT

CONNECTION

\*\* PERSISTNET COONNECTION IS IDEAL TILL ALL THE REQUEST IS COMPLETED

KEEP ALIVE

HTTPS2 connection

3

6

4

2

STREANG

CLIENT server

**HTTP3**

TCP QUICK PROTOCOL

4G 5G

\*\* IT GIVES YOU SMOTH CONNECTION WITH DIFFERENT NETWORK

**LINUX**

PWD

ls

ls-a (a=hidden files)

ls-l (L= much more data)

ls-al (al= hidden & non hidden)

clear

gitbase

apt –get update apt-get upgrade

uname

Darwin

Uname-r

Uname-a

cd.desktop/ (cd= change directory)

pwd

desktop mkdir

FSJS202319 TOUCH FILENAME.TXT (TOUCH = ADD)

FSJS202319 RM FILE.TXT (RM =REMOVE)

FSJS202319 RM FROM WHERE

**3 PORT**

Admin (root user)

Group

Publice

Admin group publice

Chmod rwx rwx

1= Execute

2= write

4 = read

(~) root directory

(chmod 777 azhar.txt)

Heroku

Render

Railway / express

Cyclic

NodeJs

Hapi

Koa

Expressjs NodeSJ / Explore

25/03/2023

\*\* Design Backend

admin

seller

product

user

payment

order

return

review

**user**

f.name

l.name

mobile.

Email.

Password

DOB

User id

**inevntory**

Name

Phone

Email

Password

GST No.

Brand name

Pan

Location

Seller code

**seller**

**admin**

Product

p.name

p.price

p.cafio

p.category

p.quantity

p.code

seller code

Name

Password

Email

Phone

Access

Pp(optional)

\*\* Category

C.name

Features

Schem a = design

\*\*Database

Tables (User , Seller, Product, inventory, Admin)

Design (That all things will go under each tables)

Round robin algorithm

Low level design

Payment Gatway

Payment Id

User id

Status = success | failed

Product id

Mode of payment (card, UPI)

Card id

UPI number

Payment Date

Amount

Transaction Id

Payment

\*\* Home Work

Ecom App design

Mind mapping tool

Excelidraw

Drawnio

Design any (Ecom, Music, Ola, Edtch, ERP)

Read : Normalization

1NF, 2NF, 3NF, BCNF (35)

26/03/2023

**Social Media App**

Super admin

Admin

Sub admin

Id

F.name

l.name

Email

Username (unique)

Password

Created at

Profile Pic

Is Active

Gender

Last Active

Bio

Location

user

Admin

Loc id

Location (l1 l)

User id

Date & Times

Email

Mobile

Password

Name

Superadmin

COMMENT

LIKE

POST

Device

Device id

User id

Location id

Device lost

\*\* Assignment

* Read about Database keys
* Normalization
* Database Design for social media
* Project take at up to Heroku debug social appin production

01/04/2023

1.MongoDB Atlas

2. ctrl+shift+p

3. dotenv, mongoose, Hooks

4. Package npm i express mongoose

5. dotenv

6. database, alway try/ catch and keep another continent

7. asynce if function

8. npm run start

9. mind map / miro

10. env : you can select only these option only

11. bcrypts

02/04/2023

1. Middleware => to check something and do something
2. ( req, res, next)
3. App.js => initialize everything
4. Server.js =>listening
5. Configure => details about dababse connection
6. Model => schema for database
7. Controller =>
8. Routes => app.set () (req, res)
9. Crud =>read
10. Folder

File => app.js

File => server.js

Folder =>configure, Model, Controller, Routes

08/04/2023

Mega Project Part -1

1.JsonwebToken

2.document ID mongoDb \_id

3. secure \_URL (https url)

09/04/2023

1. Mega Project Part -2
2. JWT.io
3. Moon model mongoDB
4. Controller
5. Custome error in nodejs
6. Express, cors, cokies parser

15/04/2023

<https://randomuser.me/api>

<https://google.com>

1. Cloudnery
2. Npm aws
3. Multer , nodeJS

16/04/2023

1. (/)
2. Passport.js
3. Frontend+backend
4. Internship
5. Hotel booking
6. WISTIA

22/04/2023

1.github/ankit9835

2.jay\_bhosayata

3. npm install node mailer

4. expressjs.com

5. Js docs

6. req.params

23/04/2023

1. Github/ankitdevelops
2. Razorpay/ sdk
3. Braintree
4. Nonce cryptography
5. Localhost/api/Hitesh/
6. React API
7. mongoDB aggregation
8. React quill

06/05/2023

1.render

2.railway

3.Digital ocean

4. cyclic

07/05/2023

1. Wireframes
2. Balsamiq
3. Buildresume.in

**Doubt Question**

1. How can I link video saved in my PC
2. Link image kept in image folder issue
3. Flexbox froggy level of 24 not resolved
4. Grid garden pending elevel cover
5. How to keep option with list
6. Whats use of style.css folder in project
7. How to find exact color codes
8. At the time of website hosting how can we remove netlify from the end
9. Stethoscope ,project-2 ,z-index:-1 not working
10. How I will get font-family link
11. How will I get search icon svg link, path (project css-3)
12. How can I input the underline of the content
13. Project summary repository not working when i clicked on linked given in readme file
14. How many value can we define in array
15. Console.log(Math.round(PI)); is not printing at terminal
16. Vs code assignment –nav barc is fixed but when doing scroll vs home page image is going upper of nav bar why?
17. How set image map
18. How to set logo image with page title
19. Project 12 top section background color?
20. Emoji shortcut key in javascript

**Pending Task**

1.2nd session assignment

2.2nd session note down

3.2nd session video note down

4.see the video of anurag sir (<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />

    <link

)

7. revision of all topics covered(20-nov second class file)

8.07-01-2023 practice

9. 08-01-2023 practice

10. Read object mdn and do practice as much as possible

11.30assignmnet(18-29 practice)

12. 9 project(Practice self)

13.18- feb practice notedown

14. Revise 18 feb lecture

15. Revise 11 feb lecture

16.WRITE PENDING ARTICLES

17.19-feb -note down and practice

18. React assignment

19.25-feb Notdown

20.Resume building as per pre placement(make project from inneuron intership for mock interview)

21.26-feb- recording notdown practice