**Design/create static website include hyper link, formatting, images,multimedia,list**

<!DOCTYPE html>

<html lang="en">

<head>

<style>

th, td {

border-style: solid;

}

</style>

<meta charset="UTF-8">

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

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

<title>Document</title>

</head>

<body bgcolor="powderblue">

<center>

<h1>XAVIER INSTITUTE OF ENGINEERING</h1><br>

<img src="college 1.jpg" align="center"><br>

<a href="https://www.xavier.ac.in/" align="center">click here to visit college website</a><br><br>

<h2><u>BE-IT TIME TABLE</u></h2><br>

<table>

<tr>

<td align="center" bgcolor="skyblue">DAY\TIME</td>

<td align="center" bgcolor="skyblue">8:45-9:45</td>

<td align="center" bgcolor="skyblue">9:45-10:45</td>

<td align="center" bgcolor="skyblue">10:45-11:00</td>

<td align="center" bgcolor="skyblue">11:00-12:00</td>

<td align="center" bgcolor="skyblue">12:00-1:00</td>

<td align="center" bgcolor="skyblue">1:00-1:30</td>

<td align="center" bgcolor="skyblue">1:30-2:30</td>

<td align="center" bgcolor="skyblue">2:30-3:30</td>

<td align="center" bgcolor="skyblue">3:30-430</td>

</tr>

<tr>

<td align="center" bgcolor="skyblue">MONDAY</td>

<td align="center">LAB</td>

<td align="center">LAB</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">IOE</td>

<td align="center">IRS</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">AI & DS-2</td>

<td align="center">STQA</td>

<td align="center">Mentoring</td>

</tr>

<tr>

<td align="center" bgcolor="skyblue">TUESDAY</td>

<td align="center">LAB</td>

<td align="center">LAB</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">IRS</td>

<td align="center">ILOC</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">STQA</td>

<td align="center">IOE</td>

</tr>

<tr>

<td align="center" bgcolor="skyblue">WEDNESDAY</td>

<td align="center">LAB</td>

<td align="center">LAB</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">IRS</td>

<td align="center">IROS</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">AI & DS-2</td>

<td align="center">STQA</td>

</tr>

<tr>

<td align="center" bgcolor="skyblue">THURSDAY</td>

<td align="center">LAB</td>

<td align="center">LAB</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">AI & DS-2</td>

<td align="center">IOE</td>

<td align="center" bgcolor="yellow">BREAK</td>

<td align="center">ILOC</td>

<td align="center">TPO ACTIVITY</td>

</tr>

<tr>

<td align="center" bgcolor="skyblue">FRIDAY</td>

<td align="center">PROJECTS</td>

</tr>

</table>

</center>

<br><br><br><br>

<ol>

<li>IOE - Internet Of Everything</li>

<li>AI & DS 2 - Artifical intelligence and Data Science-2 </li>

<li>STQA - Software Testing and Quality Assurance </li>

<li>IRS - Information Retrieval System </li>

</ol></body> </html>

**Design and create webpages including CSS 3 Syntax,Background,Fonts,Tables,Lists,CSS3 selectors**

<!DOCTYPE html>

<html lang "en”>

<head>

<style>

table,

th,

td{

border: Ipx solid rgb(57, 4, 4);

border-collapse: collapse;

}

p{

border: 4px rgb(6, 6, 12);

font-size: 20px;

</style>

meta charset="UTF-8"/>

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

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

<ink rel"stylesheet" href-"styles.css" >

<title> Document</title>

<head>

<body bgcolor="powderblue">

<center>

<h1>XAVIER INSTITUTE OF ENGINEERING<h1>

<br>

<img sr="college 1jpg" align"center"/> <br/>

<a href-"https://www.xavier.ac.in/" align"center">

<click here to visit college website-</a>

<br> <br>

<h2><u>BE-IT TIME TABLE</u></h2>

<table>

<tr>

<td align"center" bgcolor"skyblue">DAYITIME</td>

<td align"center" bgcolorskyblue>8:45-9:45</td>

<td align"center" bgcolor"skyblue">9:45-1045<td>

<td align-"center" bgcolor"skyblue>10:45-11:00</td>

ad align"center" bgcolor"skyblue>-11:00-12:00-<hd

ad align"center" bgcolor"skyblue">-12:00-1:00-hd

<d align"center" bgcolorskyblue">1:00-1:30<hd

ad align"center" bgcolor"skyblue">-1:30-2:30<hd

ad align"center" bgcolor"skyblue">2:30-3:30-<hd

ad align"center" bgcolor"skyblue>3:30-430<hd

<tr>

<td align="center" bgcolor-"skyblue">MONDAY</td>

ad align"center>LAB<hd

ad align"cnter>-LAB<d

ad align"oenter" bgcolor"yellow>BREAK<d

qd align"center">-1OE<hd

ad align"center">IRS<hd

ad align"center" bgcolor"yellow">BREAK<d

qd align"center">Al & DS-2<hd

ad align"center>-STQA<hd

ad align"center>Mentoring</td>

</tr>

<tr>

<td align oenter" bgcolor-="skyblue”> TUESDAY</td>

ad align="center">-LAB<td

<ad align="center"-LAB<nd

td align= "center" bgcolor "yellow">BREAK<d

d align "center">IRSnd

td align= oenter-lLOCd

d align="center" bgcolor "yellow'BREAKtd

ld align= "center"> STQA</td>

<td align=”center”>IOE</td>

</tr>

<tr>

ld align="center bgcolor- "skyblue>WEDNESDAY<1d>

dd align="center -LABtd

<d align- "center>LAB<Nd

td align= "oenter bgcolor "yellow>BREAK<Ad

d allgn- oenler >1RSnd

d aligne"center>-IROS<hd

d align= "center bgcolor="yellow'>BREAK<1d:

a algn= center Al & DS-2<td>

ld algn- "center51Ad

</tr>

<tr>

<d align="center" bgcolor "skyblue>THURSDAY<td>

ld align= center>LAB td>

td align= oenter">LAB<nd>

<td align= "center" bgcolor yellow>BREAK<Hd>

td algn= "center>Al & DS-2<nd>

qd align= "center">-10E<td>

<td align= "center" bgcolor-"yellow"> BREAK<td>

<ad align "center">ILOC</td>

ad align "center">TPO ACTIVITY<td>

</tr>

<tr>

<td align="center" bgcolor "skyblue">FRIDAY</td>

<td align="center">PROJECTS</d>

</tr>

</table>

</center>

<br ><br><br ><br>

<ol>

<li>lOE - Internet Of Everything<li>

<li>AI & DS 2 - Artifical intelligence and Data Science-2</li>

di>STQA- Software Testing and Quality Assurance</li>

di>IRS- Information Retrieval System</li>

</ol>

</body>

</html>

**Write a program to display CSS border properties and color properties**

**Write a program in JS to validate registration form including email validation**

HTML

<!DOCTYPE html>

<html lang="'en ">

<head>

<meta charset="UTF-8">

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

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

<link rel="stylesheet" href-"/exp4.css">

<title> Experiment 4</title>

<script defer src="/EXP4.js"></script>

</head>

<body>

<Div class="container">

<h2>Login</h2>

<p>Enter your username and password</p>

<br>

<input type="text" id="user" placeholder="Username">

<br>

<br>

<input type"password" id="password" placeholder="Password”>

<br>

<span><button id-"show'">Show Password</button></span>

<br>

<br>

<br>

<button id="'submit" onclick="validate()">Sign in</button>

<br>

<p>Dont have an Account? <a href-"#">Sign up</a></p>

</div>

<body>

<html>

CSS

body {

display: flex;

justify-content: center;

Align-items: center;

Background-color: #4158D0;

background-image: linear-gradient(43deg, #4158DO 0%, #C850C0 46%, #FFCC70 1006)

background-repeat: no-repeat;

height: 100vh;

}

.container{

background: white;

width: 22vw;

Height: max-content;

justify-items: center;

justify-content: center;

text-align: center;

}

#user{

width: 20em;

height: 3em

}

#password {

width: 20em;

height: 3em;

}

#submit{

width: 12vw;

height: 6vh;

font-size: 15px;

font-family: Times New Roman', Times, serif;

border-width: 1px;

color: rgb(17, 16, 16);

cursor: pointer;

border-color: #0b0e07;

border-top-left-radius: 2px;

border-top-right-radius: 2px;

border-bottom-left-radius: 2px;

border-bottom-right-radius: 2px;

box-shadow:inset Opx -3px 7px Opx #29bbff;

Background: linear-gradient(#2dabf9, #0688fa);

}

#submit:hover{

background: linear-gradient(#0688fa, #2dabf9);

}

JAVASCRIPT

let Pass=document.getElementByld("password");

let show=document.getE lementByld("show");

letuser=document.getElementByld("'user");

show.onclick=0=> { if (Pass.value.length >0)

{if (Pass.type==="password”) {

Pass.setAttribute("type", "text");

}else{

Pass.setAttribute("type", "password");

}

}};

function validate0{

let pswrd="^(?=\*?[A-Z])(?=\*?[a-Z])(?=\*?[0-9])(?=\*2[#?!@$%^&\*~-]).(8,)$";

if (user.value.length<=0 || Pass.value.length<=0) { console.log("no input");

alert("Please enter all the fields")

}

if (Pass.value.match(pswrd)){

console.log("valid");

alert("Welcome");

}

if (!Pass.value.match(pswrd)) {

console.log("not valid");

alert("Password is not valid");

}}

**Iterators and generators**

Iterators

function iterators (list) {

var index 0;

return fanction () {

return list [index++];

}

}

Var v;

var str="falguni joshi";

var itStr=iterators(str);

console.log(“Begin to iterate the string”);

while (v=itStr())

{

console.log(v);

}

var arr=[‘j’, ‘o’, ‘s’, ‘h’, ‘i’];

var itArr=iterators(arr);

console.log(‘Begin to iterate the array');

while (v=itArr()){

console.log(v);

}

Generator

function\* nextNatural()

{

var naturalNumber = 1;

while (true) {

yield naturalNumbert++;

}

}

var gen = nextNatural();

for (var i=0; i< 15; i++)

{

console.log(gen.next().value);

}

**Promise in JS**

const delay= (seconds)=>new Promise((resolve,reject)=>{

if(seconds>3){

reject(new Error( `${seconds} is too long`))

}

setTimeout(0=>{

resolve('the long delay has ended')

}, seconds \* 1000)

})

delay(6).then(console.log).catch(err =>console.log(err.message))

delay(4).then(0=>

{

console.log(the delay has ended')

})

delay(6).catch(err =>{

console.log(err.message)

})

**Inheritance**

**Class and function component**

FUNCTIONAL

App.js

import'/App.css;

import img 1 from'/img 1.png'

const App=0=>{

return(

<h1>This is a React functional component</h1>

<div className="image">

<img src={img1} /}

</div>

</>

}

}

export default App;

App.css

\*{

margin: 0;

padding: 0;

box-sizing: border-box;

}

h1{

max-width: 100vw;

height: 15vh;

display: flex;

justify-content: center;

align-items: cente;

text-transform: uppercase;

color: #FBEAEB;

background-color: #9e20ac;

.image{

display: flex;

justify-content: center;

align-items: center;

padding: 40px;

background-color: #FBEAEB;

}

.image>img{

border-radius: 30px;

height: 400px;

}

CLASS:

import '/App.css';

import img1 from './img1.png'

import React, {Component} from 'react’

class App extends React.Component{

render(){

return(

<hl>This is a React Class component </h1>

<div class Name="image">

<img sre{img1} >

</div>

</>

)}

}

export default App;

App.css

\*{

margin: 0;

padding: 0;

box-sizing: border-box;

}

h1{

max-width: 100vw;

height: 15vh;

display: flex;

justify-content: center;

align-items: center;

text-transform: uppercase;

color: #FBEAEB;

background-color: #1192e2;

}

.image{

display: flex;

justify-content: center;

align-items: center;

padding: 40px;

background-color: #FBEAEB;

}

.image> img{

border-radius: 30px;

height: 400px;

}

**State and props**

STATE.JS

import React, {Component} from "react";

class StateExp extends Component{

state={

name: "State",

clg: "XIE",

city: "Mahim"

}

render(){

return(

<>

<h2>This is Example of <span>{this.state.name} </span> in React</h2>

<h3>Location: {this.state.clg} , {this.state.city}</h3>

</>

)

}

}

export default State;

PROP.JS

import React, {Component} from "react";

class PropsExp extends Component{

state={

name: this.props.name,

clg: "XIE",

city: "Mahim"

}

render(){

return(

<>

<h2>This is Example of <span>{this.state.name} </span> in React</h2>

<h3>Location: {this.state.clg} , {this.state.city}</h3>

</>

)

}

}

export default PropsExp;

APP.JS

import PropsExp from "./components/PropsExp";

import StateExp from "./components/StateExp";

function App() {

return (

<div className="App">

<div className="pt1">

<StateExp/>

</div>

<div className="pt2">

<PropsExp name="Props" />

</div>

</div>

);

}

export default App;

APP.CSS

pt1{

background-color: #dfdfdf;

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

}

.pt2{

background-color: #ff8e8e;

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

}

**Form landing in Reactjs**

APP.JS:

import { useState, useEffect } from "react";

import './App.css';

function App() {

const initialValues = { username: "", email: "", password: "" };

const [formValues, setFormValues] = useState(initialValues);

const [formErrors, setFormErrors] = useState({});

const [isSubmit, setIsSubmit] = useState(false);

const handleChange = (e) => {

const { name, value } = e.target;

setFormValues({ ...formValues, [name]: value });

};

const handleSubmit = (e) => {

e.preventDefault();

setFormErrors(validate(formValues));

setIsSubmit(true);

};

useEffect(() => {

console.log(formErrors);

if (Object.keys(formErrors).length === 0 && isSubmit) {

console.log(formValues);

}

}, [formErrors]);

const validate = (values) => {

const errors = {};

const regex = /^[^\s@]+@[^\s@]+\.[^\s@]{2,}$/i;

if (!values.username) {

errors.username = "Username is required!";

}

if (!values.email) {

errors.email = "Email is required!";

} else if (!regex.test(values.email)) {

errors.email = "This is not a valid email format!";

}

if (!values.password) {

errors.password = "Password is required";

} else if (values.password.length < 4) {

errors.password = "Password must be more than 4 characters";

} else if (values.password.length > 10) {

errors.password = "Password cannot exceed more than 10 characters";

}

return errors;

};

return (

<div className="App">

<div className="container">

{Object.keys(formErrors).length === 0 && isSubmit ? (

<div className="msgSuc">Signed in successfully</div>

) : (

""

)}

<form onSubmit={handleSubmit}>

<h1>Login Form</h1>

<div className="ui-divider"></div>

<div className="ui-form">

<div className="field">

<label>Username</label>

<input

type="text"

name="username"

placeholder="Username"

value={formValues.username}

onChange={handleChange}

/>

</div>

<p>{formErrors.username}</p>

<div className="field">

<label>Email</label>

<input

type="email"

name="email"

placeholder="Email"

value={formValues.email}

onChange={handleChange}

/>

</div>

<p>{formErrors.email}</p>

<div className="field">

<label>Password</label>

<input

type="password"

name="password"

placeholder="Password"

value={formValues.password}

onChange={handleChange}

/>

</div>

<p>{formErrors.password}</p>

<button className="button">Submit</button>

<button className="button2">Clear</button>

</div>

</form>

</div>

</div>

);

}

export default App;

APP.CSS

body {

background-repeat: no-repeat;

background-size: auto;

background-size: 1980px 1080px;

background-image:

url("https://www.hdwallpapers.in/download/blur\_abstract\_lights\_4k\_hd\_abstract-1366x768.jpg");}

.container {

padding: 15px;

display: flex;

flex-direction: column;

justify-content: center;

align-items: center;

height: 100vh;

max-width: 550px;

width: 100%;

margin: auto;}

.container > form {

width: 60%;

border: 1px solid #dfdfdf;

padding: 20px;

border-radius: 5px;

background: #fff;}

pre {

width: 70%;

color: #fff;

font-size: larger;}

.button2 {

background: #ffffff;

margin-left: 13px;

border: 2px solid black;

padding: 5px;

width: 150px;

border-radius: 15px;

align-items: center;

color: #0563b4;

font-size: 18px;}

button {

margin: 4px;

background: #0563b4;

border: 2px solid #0563b4;;

padding: 5px;

width: 150px;

border-radius: 15px;

color: white;

font-size: 18px;

align-items: center;

height: 45px;}

button:hover{

opacity: 0.5;}

button:active {

opacity: 0.2;}

p {

color: red;}

.field{

font-size: large;

font-weight: 500;

margin-right: 250px;}

.field > input{

width: 310px;

height: 30px;

font-size: 18px;

border-radius: 3px;

border: 1px solid rgb(82, 82, 99);

padding: 6px;}

.msgSuc{

color: red;

font-size: 30px;

position: absolute;

top: 100px;}

**Fibonacci series in REPL**

let s = 0

let g = 1

console.log('Fibonacci series: ‘)

console.log(s)

console. log(g)

next= S+ global

while (next <=10){

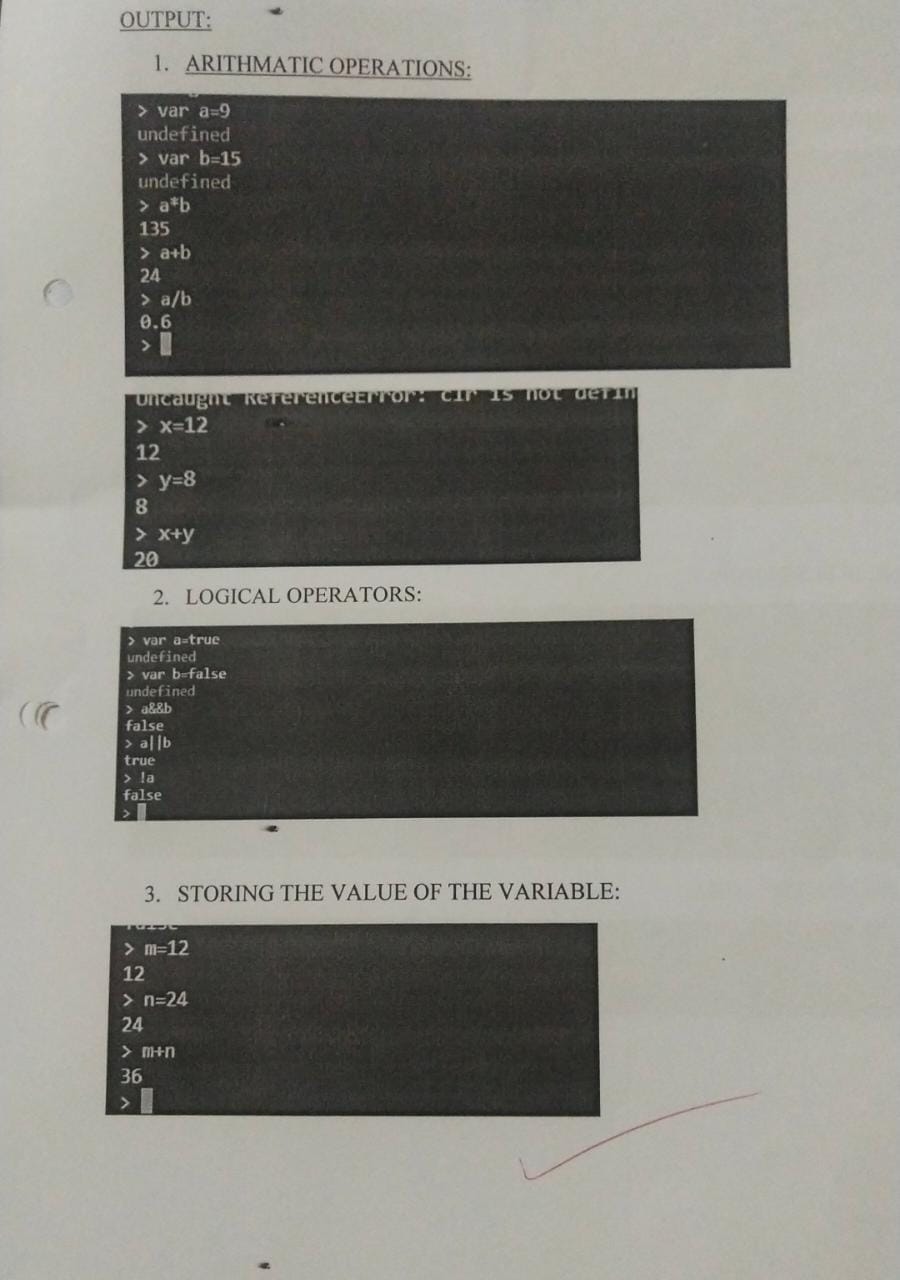
console.log(next)

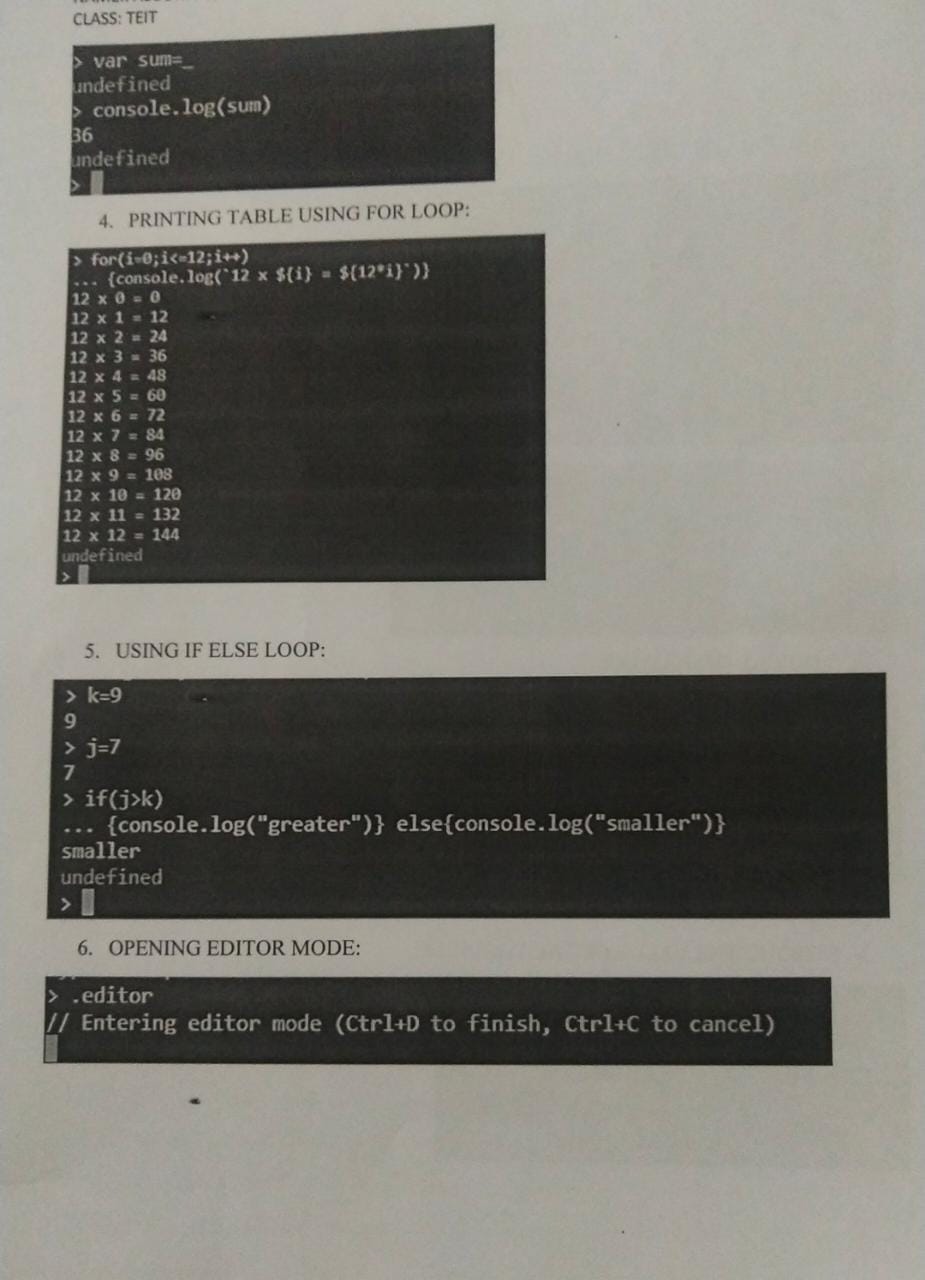
S= g

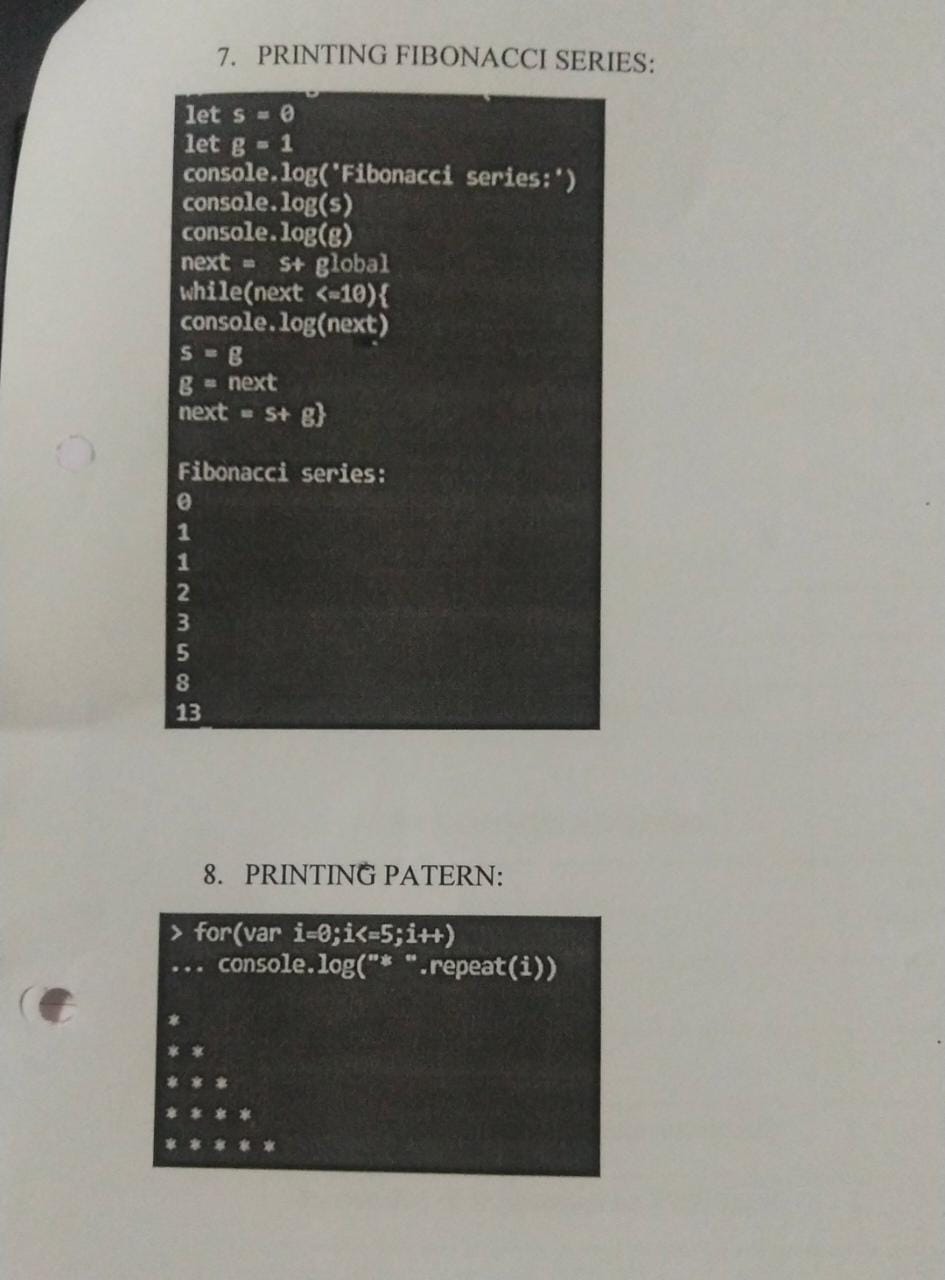
g=next

next= S+ g}

**REPL environment demonstrate variables and multiline expression**

****

****

****

**Nodejs read and write piping stream**

READ

var fs = require("fs");

var data =’ ’';

|// Create a readable stream

var readerStream = fs.createReadStream( ‘input. txt');

|// Set the encoding to be utf8.

readerStream.setEncoding( ‘UTF8’);

// Handle stream events --> data, end, and error

readerStream.on ("data', function (chunk) {

data+= chunk;

});

readerStream.on ( ‘end', function (){

console. log (data);

});

readerStream.on ( ‘error', function (err){

console. log (err. stack) ;

});

console. log ("Read Stream Completed Successfully");

WRITE

var fs = require("fs") ;

var data ='A Solution of all Technology written in text file';

|// Create a writable stream

var writerStream = fs.createWriteStream( "output.txt");

// Write the data to stream with encoding to be utf8

writerStream. write (data, 'UTF8");

// Mark the end of file

writerStream.end ();

|// Handle stream events --> finish, and error

writerStream.on(‘finish', function () {

console. log ( "Write Stream Completed. ");

});

writerStream. on ("'error', function (err){

console. log (err. stack) ;

});

console. log ("Program Ended Successfully");

PIPING

var fs= require("fs") ;

// Create a readable stream

var readerStream = fs.createReadStream ( ‘readStream . js’);

// Create a writable stream

var writerStream= fs.createWriteStream( "writeStream. js');

// Pipe the read and write operations

|// read input. txt and write data to output.txt

readerStream. pipe (writerStream);

console.log("Successfully piped the read and write operations ") ;

**Nodejs web module**

CLIENT.JS

var http= require('http’);

var options={

host: 'localhost’,

port: '8080’,

path:'EXP14 \index. Html’

};

var callback = (res)=>{

var body =””;

res.on( 'data', (data)=>{

body=+ data;

});

res.on('end',0=>{

console.log(body);

});

var req = http.request(options,.callback);

req.end();

SERVER,JS

var http = require('http');

var fs= require('fs');

var url = require('url);

http.createServer( function (req, res) {

var pathname = url.parse(req.url).pathname;

console.log("Request for " + pathname+ " received.");

fs.readFile(pathname.substr(1), function (err, data) {

if (err) {

console.log(err);

res.writeHead(404, {'Content-Type': 'text/html'});

}else{

res.writeHead(200, {'Content-Type': 'text/html'});

res.write(data.toString();

}

res.end();

});

}).listen(8080);

console.log('Server running at <http://127.0.0.1:8080/>);

**Express router**

const express= require("'express");

const app=express0;

app.get(", (req, res) =>{

res.send("<h>This is the HOME page.<h 1>");

});

app-get(about, (req, res)=>{

res.send(“<h1>This is the ABOUT page. </h1>");

});

app.get(“/contact", (req, res)=>{

res.send(“<h>This is the CONTACT page.<h1>");

app.listen(3000, 0=>{

console.log"got connected!!!!");

})’

**Class inheritance and apply styles on it**

**Even and odd numbers**

**Nodejs file system**

var fs = require("fs");

var buf = new Buffer(1024);

console.log("opening an existing file");

fs.open('input.txt', 'r+', function(err, fd) {

if (err) {

return console.error(err);

}

console.log("File opened successfully!");

console.log("reading the file");

fs.read(fd, buf, 0, buf.length, 0, function(err, bytes){

if (err){

console.log(err);

}

console.log(bytes + " bytes read");

// Print only read bytes to avoid junk.

if(bytes > 0){

console.log(buf.slice(0, bytes).toString());

}

});

});

**React router**