PRACTICAL FILE

FULL STACK DEVELOPMENT

(IT432)

Program Name: B. Tech

Semester: 6th

Batch: 2019-23

DEPARTMENT OF INFORMATION TECHNOLOGY

AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY

AMITY UNIVERSITY UTTAR PRADESH

By

Gaurav Singh,

A2305319064

B. Tech 6IT-2X

Faculty:

Ms Anju Mishra

INDEX

Practical No.	Name of Experiment	Date of Allotment of experiment	Date of Evaluation	Max Marks	Marks Obtained	Faculty Signature
1.	Create a simple web page using HTML and style it with CSS.	05/1/22		**1**		
2.	Create a college website and style by applying inline, internal, and external styles.	12/1/22		**1**		
3.	Write a program in JavaScript to implement variables, arrays, loops, functions, objects and to remove duplicates from an array.	19/1/22		**1**		
4.	Write a jQuery program • to show and hide button on click. • to change color on mouse, enter, leave, and click. • to add a row in a table on click. • to demonstrate use of animate() method. • to demonstrate use of DOM manipulation methods.	02/2/22		**1**		
5.	Write a Javascript program: • to make a 3x3 slider puzzle using JQuery • to make a tic tac toe game using JQuery	09/2/22				
6.	Write commands in Mongodb to create a collection and try different operations like insert,delete and update.	02/3/22		**1**		
7.	Write commands in MongoDB to create indexes.	09/3/22		**1**		
8.	Write a program in Nodejs using user defined functions, modules, routers and perform read and write operations.			**1**		
9.	Write a program in Nodejs to connect MongoDB database and fetch data from database.			**1**		
10.	Write a program demonstrating two-way databinding in Angular JS.					
11.	Write a program demonstrating events in Angular JS.			**1**		
12.	Write a program to demonstrate the use of filters in Angular JS.					

Experiment 1

Objective: Create a simple web page using HTML and style it with CSS.

Software Used: Visual Studio Code

Program:

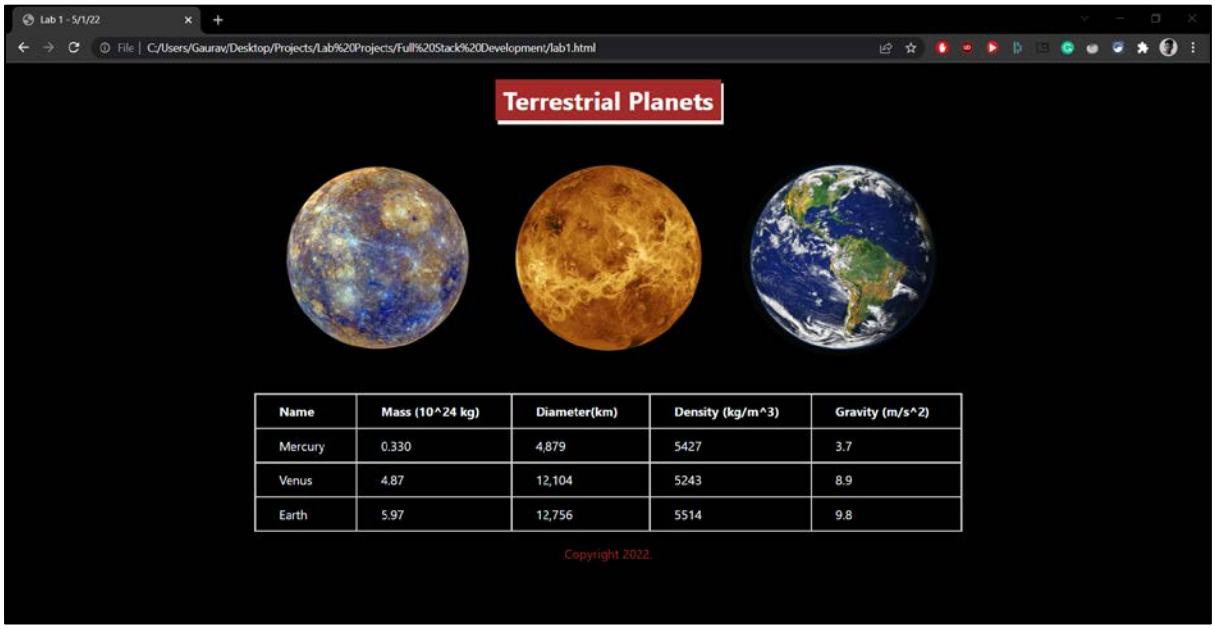
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Lab 1 - 5/1/22</title>
    <style>
        * {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
        }
        img {
            width: 250px;
            height: 250px;
            border-radius: 50%;
            object-fit: cover;
            transition: transform 10s ease-in-out;
            padding: 0 20px;
        }
        img:hover {
            transform: rotate(360deg);
        }

        h1 {
```

```
        box-shadow: 3px 5px white;
        margin-bottom: 50px;
    }

    td {
        border: solid 2px lightgrey;
    }
    th,
    td {
        padding-top: 10px;
        padding-bottom: 10px;
        padding-left: 30px;
        padding-right: 40px;
    }
  </style>
</head>
<body style="background-color: black;">
  <center>
    <h1 style="color: white; background-color: brown; width: max-content; padding: 5px 10px;">Terrestrial Planets
    </h1>
    <img src="https://cdn.mos.cms.futurecdn.net/GA4grWEsUYUqH58cDbRBw8.jpg">
    <img src="https://www.nasa.gov/sites/default/files/thumbnails/image/imagesvenus20191211venus20191211-16.jpeg">
    <img src="https://assets.weforum.org/article/image/25NbfYbkiuMvTW3P_YO8QeE-SxtvNKGRX9Dgr6W-gNE.jpg">
    <br><br><br>
    <table style="border-collapse: collapse; color: aliceblue;">
      <tr>
        <td><b>Name</b></td>
        <td><b>Mass (10^24 kg)</b></td>
```

```
            <td><b>Diameter(km)</b></td>
            <td><b>Density (kg/m^3)</b></td>
            <td><b>Gravity (m/s^2)</b></td>
        </tr>
        <tr>
            <td>Mercury</td>
            <td>0.330</td>
            <td>4,879</td>
            <td>5427</td>
            <td>3.7</td>
        </tr>
        <tr>
            <td>Venus</td>
            <td>4.87</td>
            <td>12,104</td>
            <td>5243</td>
            <td>8.9</td>
        </tr>
        <tr>
            <td>Earth</td>
            <td>5.97</td>
            <td>12,756</td>
            <td>5514</td>
            <td>9.8</td>
        </tr>
    </table>
  </center>
</body>
<footer>
  <center>
    <p style="color: brown;">Copyright 2022.</p>
```

```
    </center>
</footer>
</html>
```

Output:



Experiment 2

Objective: Create a college website and style by applying inline, internal, and external styles.

Software Used: Visual Studio Code

Program:

1) index.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Lab 2 - 12/1/22</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" type="text/css" href="style.css">
    <style>
        h3 {
            margin-top: 20px;
            color: #fff;
            font-size: 25px;
        }

        p {
            margin-top: 10px;
            font-size: 18px;
        }
    </style>
    <script type="text/javascript">
        function sayHello() {
            var name = document.getElementById('name').value;
```

```
        alert("Hey, " + name + ". Amity University, Noida is a private research university located in Noida, India. The university is recognized by University Grants Commission and accredited by the NAAC with grade 'A+'.")
      }
    </script>
</head>
<body>
    <div class="section">
      <div class="container">
        <div class="content-section">
          <div class="title">
            <h1 style="color: #ffd500; font-size: 50px;">About Amity</h1>
          </div>
          <div class="content">
            <h3>Lorem ipsum dolor sit amet, consectetur adipisicing</h3>
            <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
              tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
              quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
              consequat.</p>
            <div class="button">
              <input type="text" name="name" id="name" placeholder="Enter your name: ">
              <input type="button" value="Read More" onclick="sayHello()">
            </div>
          </div>
        </div>
        <div class="image-section">
          <img src="https://www.amity.edu/backoffice/uploads/HomeBanner/1fblg_noida-cam-img.jpg"
            style="border: 10px solid #fff;" width="500px" height="250px">
        </div>
      </div>
```

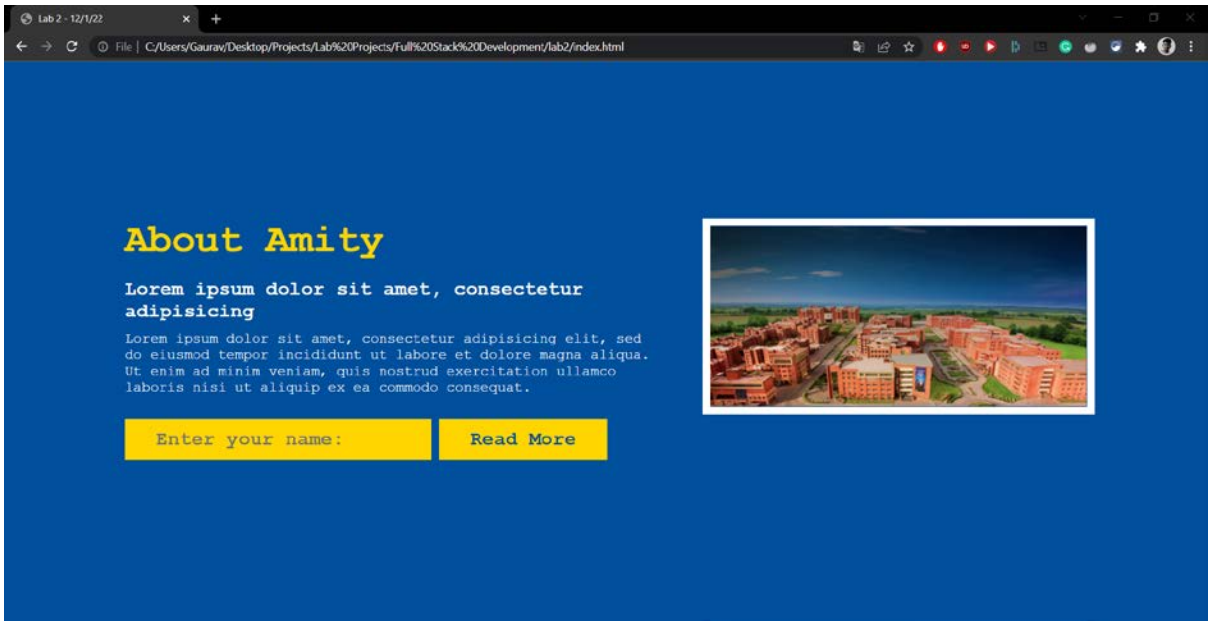
```
    </div>
</body>
</html>
```

2) style.css

```
*{
        margin:0px;
        padding:0px;
        box-sizing: border-box;
        font-family:'Courier New', Courier, monospace , sans-serif;
}
.section{
        width: 100%;
        min-height: 100vh;
        background-color: #00509d;
}
.container{
        width: 80%;
        display: block;
        margin:auto;
        padding-top: 200px;
}
.content-section{
        float: left;
        width: 55%;
}
.image-section{
        float: right;
        width: 40%;
}
.content{
    color: #fff;
```

```
}
.button{
        margin-top: 30px;
}
input{
        background-color: #ffd500;
        padding:12px 40px;
        color:#00509d;
        font-size: 25px;
    font-weight: bold;
    border: 0;
}
```

Output:



Experiment 3

Objective: Write a program in JavaScript to implement variables, arrays, loops, functions, objects and to remove duplicates from an array.

Software Used: Visual Studio Code

Program:

```
<!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">
    <title>FSD Lab 3 - 19/1/22</title>
    <style>
        body div {
            margin: 20px 0;
        }
        div div {
            margin: 0;
        }
        button {
            margin-top: 10px;
        }
        .obj input {
            margin-left: auto;
            margin-bottom: 5px;
        }
    </style>
    <script>
        function varInit() {
```

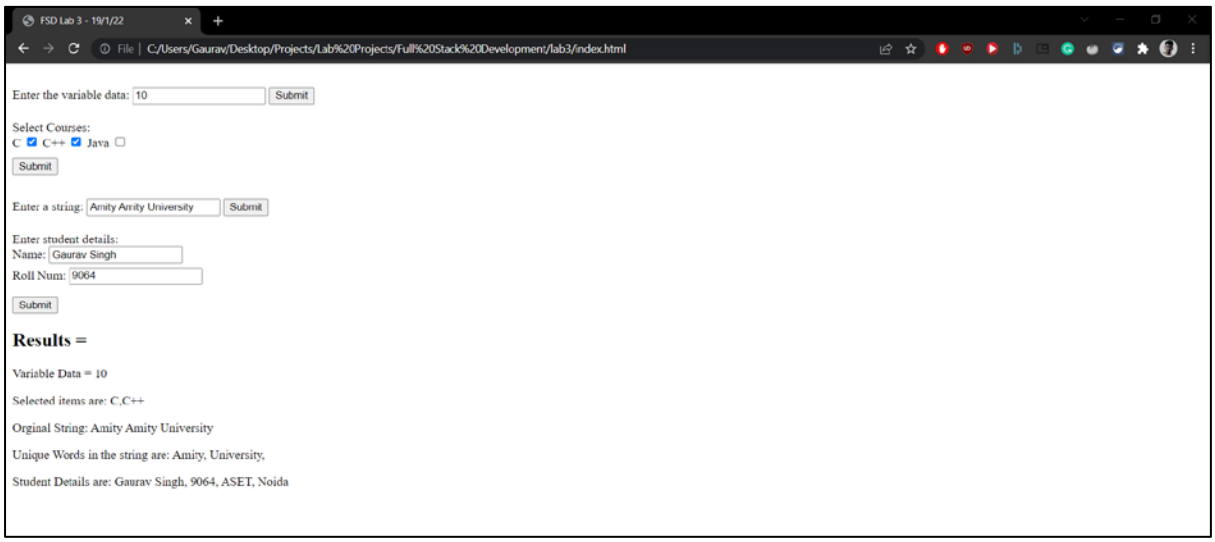
```
    let var1 = document.getElementById("var").value;
    var output = "Variable Data = " + var1;
    document.getElementById("varop").innerHTML = output;
    console.log(var1);
}
function arrInit() {
    var arr = [];
    var checks = document.querySelectorAll('input[type=checkbox]:checked');

    for (let i = 0; i < checks.length; i++) {
        arr.push(checks[i].value);
    }
    var op = "Selected items are: " + arr;
    document.getElementById("varop2").innerHTML = op;

    console.log(arr);
}
function remDup() {
    let str = document.getElementById("str").value;
    const words = str.split(" ")
    const set = new Set(words);
    const org = "Orginal String: " + str;
    let op = "Unique Words in the string are: ";
    for (let item of set)
        op += item + ", ";

    document.getElementById("varop3").innerHTML = org;
    document.getElementById("varop4").innerHTML = op;
    console.log(set);
}
function createObj(i, n) {
```

```
            return {
                id: i,
                name: n,
                college: "ASET, Noida"
            };
        }
        function printObj() {
            const name = document.getElementById("rnum").value;
            const rollnum = document.getElementById("name").value;
            let stu = new createObj(rollnum, name);

            let op = "Student Details are: " + stu.id + ", " + stu.name + ", " + stu.college;
            document.getElementById("varop5").innerHTML = op;

            console.log(stu.id);
            console.log(stu.name);
            console.log(stu.college);
        }
    </script>
</head>

<body>
    <div>
        <label for="var">Enter the variable data: </label>
        <input type="text" name="data" id="var" value="10" required>
        <button onclick="varInit()">Submit</button>
    </div>
    <div>
        <label for="check">Select Courses: </label>
        <div>
            <label for="">C</label>
```

```
        <input type="checkbox" name="" id="check" value="C">
        <label for="">C++</label>
        <input type="checkbox" name="" id="check" value="C++">
        <label for="">Java</label>
        <input type="checkbox" name="" id="check" value="Java">
    </div>
    <button onclick="arrInit()">Submit</button>
</div>
<div>
    <label>Enter a string: </label>
    <input type="text" id="str">
    <button onclick="remDup()">Submit</button>
</div>
<div>
    <label>Enter student details: </label>
    <br>
    <div class="obj">
        <label>Name: </label>
        <input type="text" id="name">
        <br>
        <label>Roll Num: </label>
        <input type="number" id="rnum">
        <br>
    </div>
    <button onclick="printObj()">Submit</button>
</div>
<div>
    <h2>Results =</h2>
    <p id="varop"></p>
    <p id="varop2"></p>
    <p id="varop3"></p>
```

```
        <p id="varop4"></p>
        <p id="varop5"></p>
    </div>
</body>
</html>
```

Output:



Experiment 4

Objective: Write a jQuery program

- to show and hide button on click.
- to change color on mouse, enter, leave, and click.
- to add a row in a table on click.
- to demonstrate use of animate() method.
- to demonstrate use of DOM manipulation methods.

Software Used: Visual Studio Code

Program:

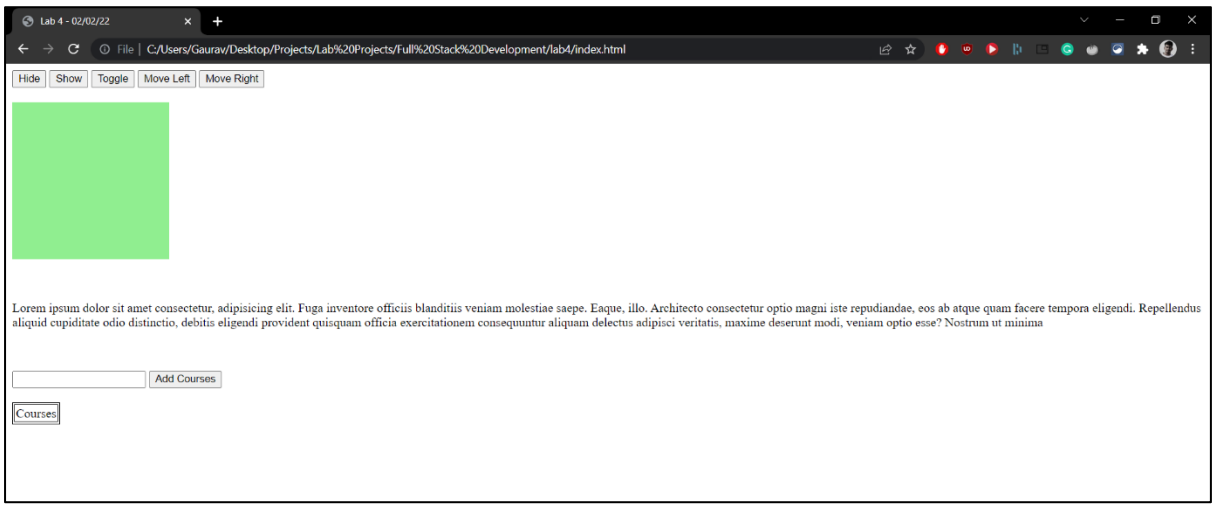
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Lab 4 - 02/02/22</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
    <script>
        $(document).ready(function () {
            $("#btn-1").click(function () {
                $('div').hide();
            });
            $("#btn-2").click(function () {
                $('div').show();
            });
            $("#btn-3").click(function () {
                $('div').toggle();
            });
            $("#box").mouseenter(function () {
                $("#box").css("background-color", "lightgreen");
            });
```

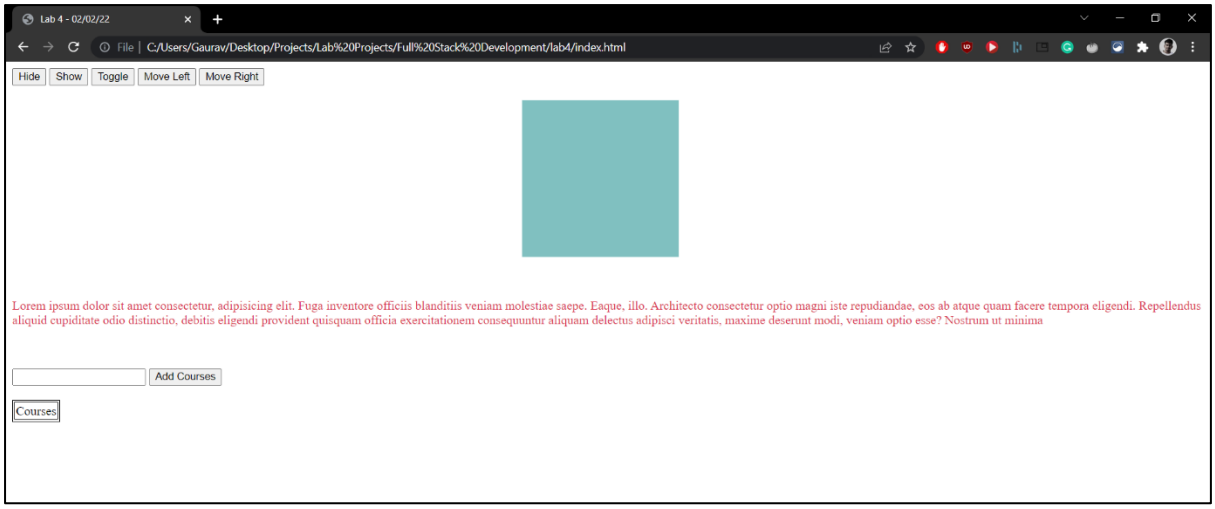
```
        $("#box").mouseleave(function () {
            $("#box").css("background-color", "teal");
        });
        $("#text").on(
            "click", function () {
                $("#text").toggleClass("pink");
            });
        $("#add_c").click(function () {
            var data = "<tr><td>" + $("#course").val() + "</td></tr>";
            console.log(data);
            $("#course_table").append(data);
        });
        $("#btn-4").click(function () {
            $("#box").animate({
                opacity: 0.50,
                left: "+=50"
            });
        });
        $("#btn-5").click(function () {
            $("#box").animate({
                opacity: 1,
                left: "-=50"
            });
        });
    });
</script>
<style>
    table,
    th,
    td {
        border: 1px solid;
```

```
        }
        .pink {
            color: rgb(219, 51, 79);
        }
    </style>
</head>
<body>
    <button id="btn-1">Hide</button>
    <button id="btn-2">Show</button>
    <button id="btn-3">Toggle</button>
    <button id="btn-4">Move Left</button>
    <button id="btn-5">Move Right</button>
    <br><br>
    <div id="box" style="width: 200px; height: 200px; background-color: teal; position: relative;"></div>
    <br><br>
    <p id="text">Lorem ipsum dolor sit amet consectetur, adipisicing elit. Fuga inventore officiis blanditiis veniam
        molestiae
        saepe. Eaque, illo. Architecto consectetur optio magni iste repudiandae, eos ab atque quam facere tempora
        eligendi.
        Repellendus aliquid cupiditate odio distinctio, debitis eligendi provident quisquam officia exercitationem
        consequuntur aliquam delectus adipisci veritatis, maxime deserunt modi, veniam optio esse? Nostrum ut minima</p>
    <br><br>
    <input type="text" name="course" id="course">
    <button id="add_c">Add Courses</button>
    <br><br>
    <table id="course_table">
        <thead>
```

```
            <tr>
                <td>Courses</td>
            </tr>
        </thead>
        <tbody>
        </tbody>
    </table>
</body>
</html>
```

Output:




Lab 4 - 02/02/22
File | C:/Users/Gaurav/Desktop/Projects/Lab%20Projects/Full%20Stack%20Development/lab4/index.html
Hide
Show
Toggle
Move Left
Move Right
Lorem ipsum dolor sit amet consectetur, adipisicing elit. Fuga inventore officiis blanditiis veniam molestiae saepe. Eaque, illo. Architecto consectetur optio magni iste repudiandae, eos ab atque quam facere tempora eligendi. Repellendus aliquid cupiditate odio distinctio, debitis eligendi provident quisquam officia exercitationem consequuntur aliquam delectus adipisci veritatis, maxime deserunt modi, veniam optio esse? Nostrum ut minima
Add Courses
Courses

Experiment 5

Objective: Write a Javascript program

- to make a 3x3 slider puzzle using JQuery
- to make a tic tac toe game using JQuery

Software Used: Visual Studio Code

Program:

1) sliderPuzzle.html

```
<!DOCTYPE html>

<html>

<head>

<title>8-Puzzle</title>

<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
integrity="sha384-
BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">

<script type="text/javascript" src="script.js"></script>

</head>

<body>

<center>

<a style="text-decoration: none; font-size: 5em;" href="/">

<span style="color:#000000;">8</span> <span style="color: #000;">Puzzle Game</span>

</a>

<div id="puzzle">

  <button id="b0" class="btn btn-default" onclick="pushed(this.id)" style="width:100px;
height:100px;font-size: 30px;color:#3377CC;"> </button> <button id="b1" class="btn btn-
default" onclick="pushed(this.id)" style="width:100px; height:100px;font-size:
30px;color:#3377CC;">1</button> <button id="b2" class="btn btn-default"
onclick="pushed(this.id)" style="width:100px; height:100px;font-size:
30px;color:#3377CC;">2</button><br />

  <button id="b3" class="btn btn-default" onclick="pushed(this.id)" style="width:100px;
height:100px;font-size: 30px;color:#3377CC;">3</button> <button id="b4" class="btn btn-
default" onclick="pushed(this.id)" style="width:100px; height:100px;font-size:
30px;color:#3377CC;">4</button> <button id="b5" class="btn btn-default"
```

```
onclick="pushed(this.id)" style="width:100px; height:100px;font-size:
30px;color:#3377CC;">5</button><br />

   <button id="b6" class="btn btn-default" onclick="pushed(this.id)" style="width:100px;
height:100px;font-size: 30px;color:#3377CC;">6</button> <button id="b7"  class="btn btn-
default" onclick="pushed(this.id)" style="width:100px; height:100px;font-size:
30px;color:#3377CC;">7</button> <button id="b8" class="btn btn-default"
onclick="pushed(this.id)" style="width:100px; height:100px;font-size:
30px;color:#3377CC;">8</button>

</div>

<br />

<button class="btn btn-default" style="width:200px; height:50px;font-size:
20px;color:#fc4f3f;" onclick="randomTaslar()">New Pattern</button>

<br /><br />

</center>

</body>

</html>
```

Script.js-

```
let taslar = ["b0","b1","b2","b3","b4","b5","b6","b7","b8"]

function tasBul(val){
   for (i = 0; i < taslar.length; i++) {
         if(document.getElementById(taslar[i]).firstChild.data == val){
            return(taslar[i])}}}
function degisme(id, bosTasId){
   let yakinlar = []
   if([2,5,8].includes(parseInt(bosTasId[1]))){
      yakinlar = [+3,-3,-1]
   }else if([0,3,6].includes(parseInt(bosTasId[1]))){
      yakinlar = [+3,+1,-3]
   }else{
      yakinlar = [+3,+1,-3,-1]
   }
   for(i = 0; i < taslar.length; i++){
      if(parseInt(bosTasId[1])+parseInt(yakinlar[i]) == parseInt(id[1])){
```

```
            return(true); }}
    return(false)}
function pushed(id){
    var btn = document.getElementById(id);
    if (btn.firstChild.data!=" "){
        bosTasId = tasBul(" ")
        if(degisme(id, bosTasId) == false) return;
        document.getElementById(bosTasId).firstChild.data = btn.firstChild.data;
        btn.firstChild.data = " "}}
function solvable(rndList){
    var count = 0;
    for(i=0;i<rndList.length-1;i++){
        if(rndList[i] == 0){
            continue;
        }
        for(j=i+1;j<rndList.length;j++){
            if(rndList[j] == 0){
                continue;
            }else if(rndList[i]>rndList[j]){
                count++;
            }}}
    if(count%2 == 0){
        return(true);
    }else{
        return(false);
    }
}
function randomTaslar(){
    var rndList = []
    while(true){
        rndList = []
```

```
        while(rndList.length < 9){
            var randomnumber = Math.ceil(Math.random()*9)-1
            if(rndList.indexOf(randomnumber) > -1) continue;
            rndList[rndList.length] = randomnumber;
        }
        if(solvable(rndList)){
            break;
        }
    }
    for (i = 0; i < taslar.length; i++) {
        if(rndList[i] == 0){
            val = " "
        }else{
            val = rndList[i].toString()
        }
        document.getElementById(taslar[i]).firstChild.data = val
    }
}
```

2) Tic Tac Toe

(i) index.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">
    <title>Tic Tac Toe - 9/2/22</title>
    <link rel="stylesheet" href="style.css">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
</head>
```

```
<body>
    <center>
        <h1>Tic Tac Toe</h1>
        <h3 id="#result"></h3>
        <div id="tictactoe">
            <table id="board">
                <tr>
                    <td></td>
                    <td></td>
                    <td></td>
                </tr>
                <tr>
                    <td></td>
                    <td></td>
                    <td></td>
                </tr>
                <tr>
                    <td></td>
                    <td></td>
                    <td></td>
                </tr>
            </table>
            <div class="scoreboard">
                <h3 id="result"></h3>
                <br>
                <button class="reset">Reset</button>
            </div>
        </div>
    </center>
    <script src="script.js"></script>
</body>
```

```
</html>
```

(ii) style.css

```
* {
        margin: 0;
        padding: 0;
        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
}
table {
        border-collapse: collapse;
        border-spacing: 0;
        padding: 0px;
        margin: 10px auto;
}
h1{
   text-decoration: underline;
}
table tr td {
        width: 80px;
        height: 80px;
        border: 10px solid black;
        font-size: 30px;
        text-align: center;
}
.x{
   background-color: rgb(245, 19, 11);
}

.o{
   background-color: rgb(0, 60, 255);
}
```

```
#result{
    font-size: 25px;
}
.reset{
    padding: 0 20px;
    font-size: 20px;
}
```

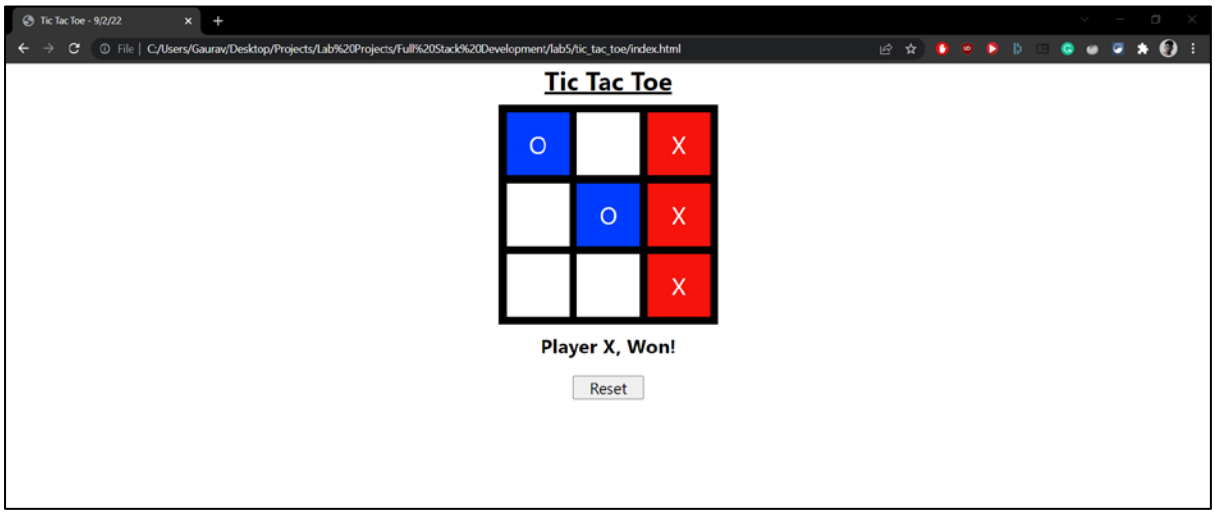
(iii) script.js

```
$(document).ready(function () {
    var move = 1;
    var play = true;
    $("#board tr td").click(function () {
        if ($(this).text() == "" && play) {
            if ((move % 2) == 1) {
                $(this).append("X");
                $(this).css('color', "white");
                $(this).addClass("x");
            } else {
                $(this).append("O");
                $(this).css('color', "white");
                $(this).addClass("o");
            }
            move++;
            if (checkForWinner() != -1 && checkForWinner() != "") {
                if (checkForWinner() == "X") {
                    $('#result').append('Player X, Won!');
                }
                else if (checkForWinner() == "O") {
                    $('#result').append('Player O, Won!');
                }
```

```
            else {
                $('#result').append('Draw, play again!');
            }
            play = false;
        }
    }
});
$(".reset").click(function () {
    location.reload();
});
function checkForWinner() {
    var space1 = $("#board tr:nth-child(1) td:nth-child(1)").text();
    var space2 = $("#board tr:nth-child(1) td:nth-child(2)").text();
    var space3 = $("#board tr:nth-child(1) td:nth-child(3)").text();
    var space4 = $("#board tr:nth-child(2) td:nth-child(1)").text();
    var space5 = $("#board tr:nth-child(2) td:nth-child(2)").text();
    var space6 = $("#board tr:nth-child(2) td:nth-child(3)").text();
    var space7 = $("#board tr:nth-child(3) td:nth-child(1)").text();
    var space8 = $("#board tr:nth-child(3) td:nth-child(2)").text();
    var space9 = $("#board tr:nth-child(3) td:nth-child(3)").text();
    if ((space1 == space2) && (space2 == space3)) {
        return space3;
    } else if ((space4 == space5) && (space5 == space6)) {
        return space6;
    } else if ((space7 == space8) && (space8 == space9)) {
        return space9;
    }
    else if ((space1 == space4) && (space4 == space7)) {
        return space7;
    } else if ((space2 == space5) && (space5 == space8)) {
        return space8;
```

```
    } else if ((space3 == space6) && (space6 == space9)) {
        return space9;
    }
    else if ((space1 == space5) && (space5 == space9)) {
        return space9;
    } else if ((space3 == space5) && (space5 == space7)) {
        return space7;
    }
    return -1;
  }
});
```

Output:



Experiment 6

Objective: Write commands in MongoDB to create a collection and try different operations like insert,delete and update.

Software Used: Visual Studio Code

Program:

```
> use college

switched to db college

> db.createCollection('student_details')

{ "ok" : 1 }

> db.createCollection('course_details')

{ "ok" : 1 }

> db.student_details.insertMany([{"name" : "Gaurav Singh", "roll_num" : 9064}, {"name" : "Armaan Gulia", "roll_num" : 9076}, {"name" : "Pankaj", "roll_num" : 9124}])

{
	"acknowledged" : true,
	"insertedIds" : [
		ObjectId("622a1b73c46d06949b62e273"),
		ObjectId("622a1b73c46d06949b62e274"),
		ObjectId("622a1b73c46d06949b62e275")
	]
}

> db.course_details.insertMany([{"name" : "FSD", "code" : "IT432"}, {"name" : "AI", "code" : "CSE401"}, {"name" : "SPM", "code" : "CSE432"}])

{
	"acknowledged" : true,
	"insertedIds" : [
		ObjectId("622a1befc46d06949b62e276"),
		ObjectId("622a1befc46d06949b62e277"),
		ObjectId("622a1befc46d06949b62e278")
	]
```

```
}
> db.student_details.find({"name" : "Gaurav Singh"}).pretty()
{
        "_id" : ObjectId("622a1b73c46d06949b62e273"),
        "name" : "Gaurav Singh",
        "roll_num" : 9064
}
> db.student_details.find({}).sort({"roll_num" : -1}).pretty()
{
        "_id" : ObjectId("622a1b73c46d06949b62e275"),
        "name" : "Pankaj",
        "roll_num" : 9124
}
{
        "_id" : ObjectId("622a1b73c46d06949b62e274"),
        "name" : "Armaan Gulia",
        "roll_num" : 9076
}
{
        "_id" : ObjectId("622a1b73c46d06949b62e273"),
        "name" : "Gaurav Singh",
        "roll_num" : 9064
}
```

Output:

```
> use college
switched to db college
> db.createCollection('student_details')
{ "ok" : 1 }
> db.createCollection('course_details')
{ "ok" : 1 }
> db.student_details.insertMany([{"name" : "Gaurav Singh", "roll_num" : 9064}, {"name" : "Armaan Gulia", "roll_num" : 9076}, {"name" : "Pankaj", "roll_num" : 9124}])
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("622a1b73c46d06949b62e273"),
                ObjectId("622a1b73c46d06949b62e274"),
                ObjectId("622a1b73c46d06949b62e275")
        ]
}
> db.course_details.insertMany([{"name" : "FSD", "code" : "IT432"}, {"name" : "AI", "code" : "CSE401"}, {"name" : "SPM", "code" : "CSE432"}])
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("622a1befc46d06949b62e276"),
                ObjectId("622a1befc46d06949b62e277"),
                ObjectId("622a1befc46d06949b62e278")
        ]
}
> db.student_details.find({"name" : "Gaurav Singh"}).pretty()
{
        "_id" : ObjectId("622a1b73c46d06949b62e273"),
        "name" : "Gaurav Singh",
        "roll_num" : 9064
}
> db.student_details.find({}).sort({"roll_num" : -1}).pretty()
{
        "_id" : ObjectId("622a1b73c46d06949b62e275"),
        "name" : "Pankaj",
        "roll_num" : 9124
}
{
        "_id" : ObjectId("622a1b73c46d06949b62e274"),
        "name" : "Armaan Gulia",
        "roll_num" : 9076
}
{
        "_id" : ObjectId("622a1b73c46d06949b62e273"),
        "name" : "Gaurav Singh",
        "roll_num" : 9064
```

Experiment 7

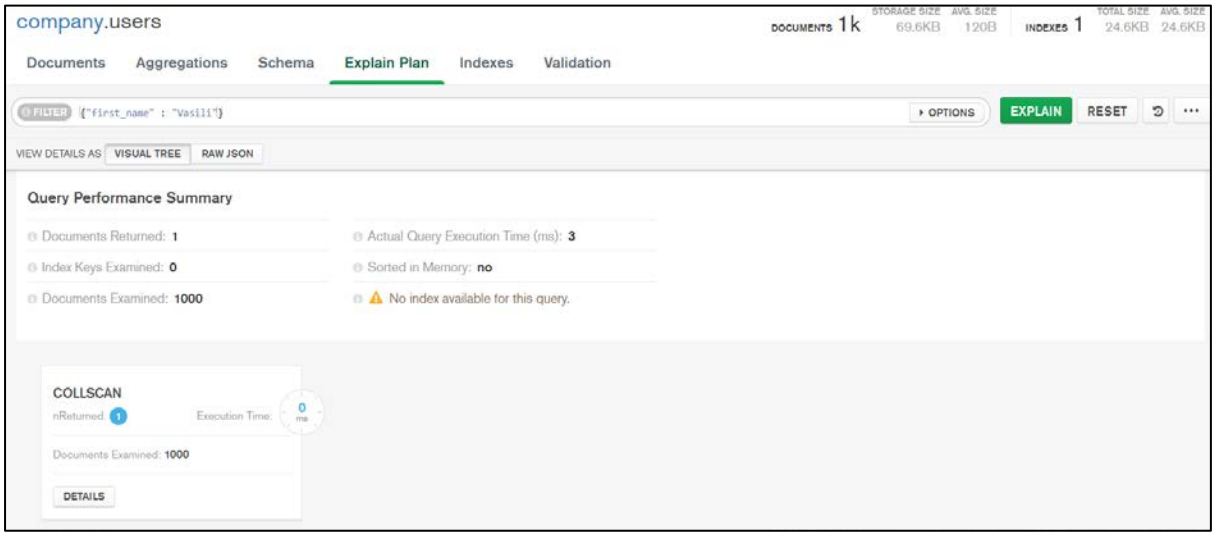
Objective: Write commands in MongoDB to create indexes.

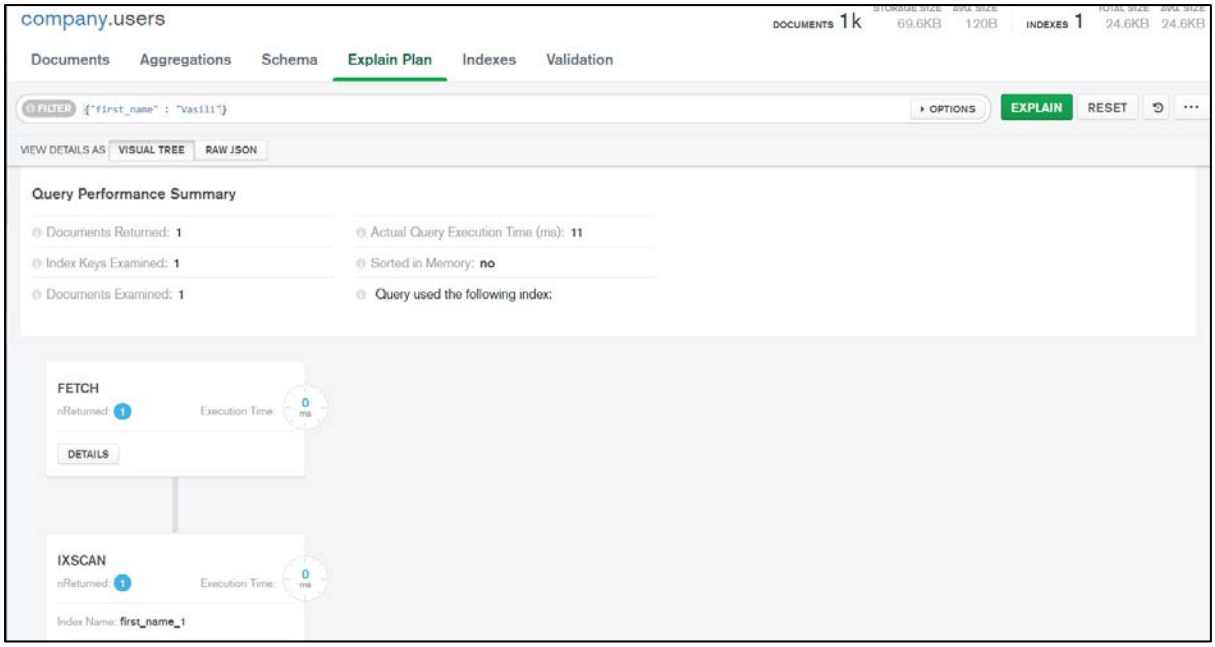
Software Used: Visual Studio Code

Program:

```
> use company
switched to db company
> db.users.createIndex({"first_name" : 1})
{
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "createdCollectionAutomatically" : false,
        "ok" : 1
}
```

Output:




company.users
DOCUMENTS 1k
STORAGE SIZE 69.6KB
AVG. SIZE 120B
INDEXES 1
TOTAL SIZE 24.6KB
AVG. SIZE 24.6KB
Documents
Aggregations
Schema
Explain Plan
Indexes
Validation
FILTER {"first_name" : "Vasili"}
OPTIONS
EXPLAIN
RESET
VIEW DETAILS AS
VISUAL TREE
RAW JSON
Query Performance Summary
Documents Returned: 1
Index Keys Examined: 1
Documents Examined: 1
Actual Query Execution Time (ms): 11
Sorted in Memory: no
Query used the following index:
FETCH
nReturned: 1
Execution Time: 0 ms
DETAILS
IXSCAN
nReturned: 1
Execution Time: 0 ms
Index Name: first_name_1

Experiment 8

Objective: Write a program in Nodejs using user defined functions, modules, routers and perform read and write operations.

Software Used: Visual Studio Code

Program:

1. Run Server

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/html'});
  res.end('<h3>Hello World</h3>');
}).listen(8080);
```

2. User-defined modules

(i) index.js -

```
var http = require('http');
var math = require('./maths.js');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/html'});
  res.write('<h3>Sum of 22 and 100 is: </h3>'+math.sum(22,100));
  res.write('<h3>Difference of 22 and 100 is: </h3>'+math.difference(22,100));
}).listen(8000);
```

(ii) maths.js –

```
sum = (a,b) =>
{
   return (a + b);
}
difference = (a,b) =>
{
   return (a - b);
}
```

```
module.exports.sum = sum
module.exports.difference = difference
```

3. Read HTML File

Index.js

```
var http = require('http');
var fs = require('fs');
http.createServer(function (req, res) {
  fs.readFile('home.html', function(err, data) {
    res.writeHead(200, {'Content-Type': 'text/html'});
    res.write(data);
    return res.end();
  })
}).listen(8000);
```

4. Append HTML File

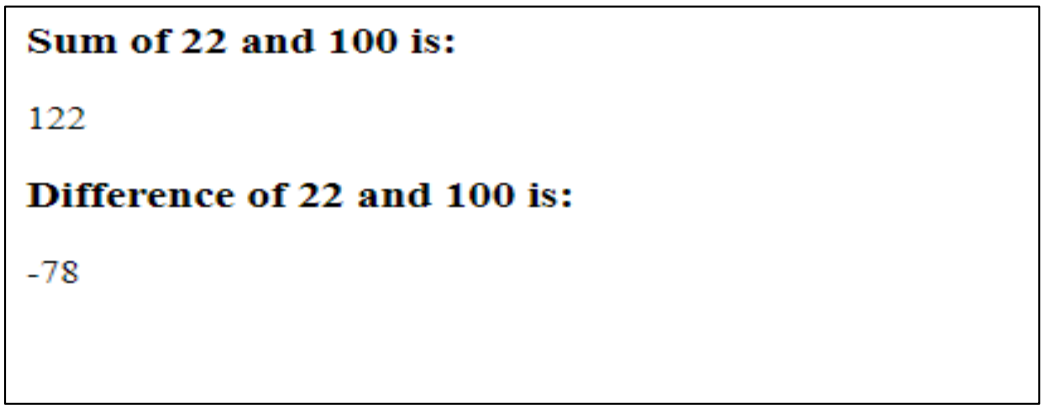
```
var http = require('http');
var fs = require('fs');
http.createServer(function (req, res) {
  fs.readFile('home.html', function(err, data) {
    res.writeHead(200, {'Content-Type': 'text/html'});
    res.write(data);
    return res.end();
  });
  fs.appendFile('home.html', '<h5>This is my text</h5>', function (err) {
    if (err) throw err;
    console.log('Replaced!');
  });
  }).listen(8000);
```

Output:

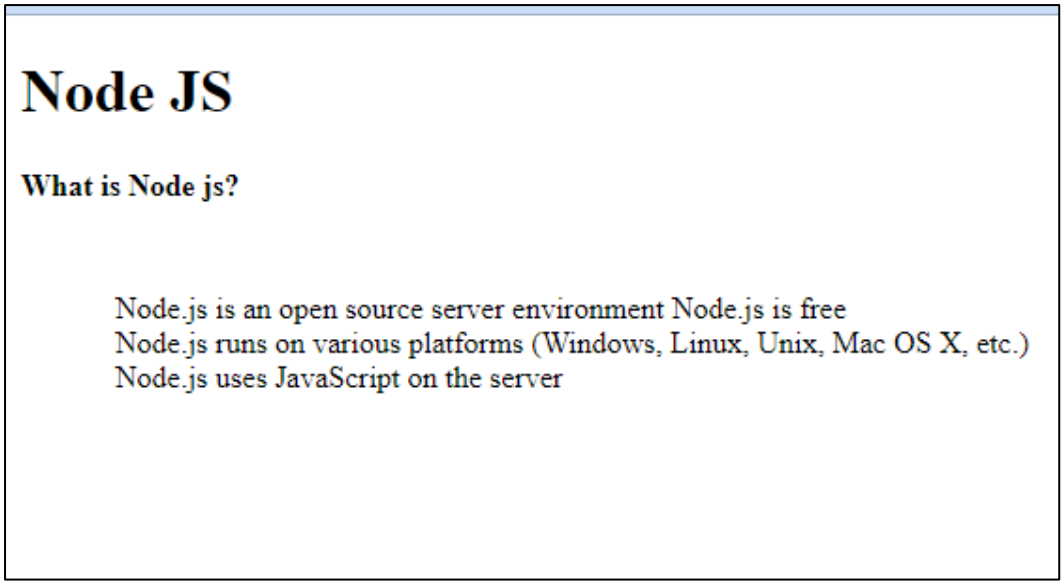
1.


localhost:8080
Gmail
YouTube
Hello World

2.


Sum of 22 and 100 is:
122
Difference of 22 and 100 is:
-78

3.


Node JS
What is Node js?
Node.js is an open source server environment Node.js is free
Node.js runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
Node.js uses JavaScript on the server

4.

Node JS

What is Node js?

Node.js is an open source server environment Node.js is free
Node.js runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
Node.js uses JavaScript on the server

This is my text

This is my text

Experiment 9

Objective: Write a program in Nodejs to connect MongoDB database and fetch data from database.

Software Used: Visual Studio Code

Program:

1. Connecting to MongoDB using Node.js - (File: db.js)

```
var express = require('express')
var MongoClient = require('mongodb').MongoClient;
var app = express()
var database
app.get('/', function(req, res){
  res.sendFile(__dirname+'/home.html');
});
app.listen(8000, () => {
MongoClient.connect('mongodb://localhost:27017',{useNewUrlParser:true},(error,result) =>
{
      if(error) throw error
      database = result.db('college_database')
      console.log('Connected to College database')

  })
});
```

2. Fetching data from database and displaying it. (File Name: index.ejs)

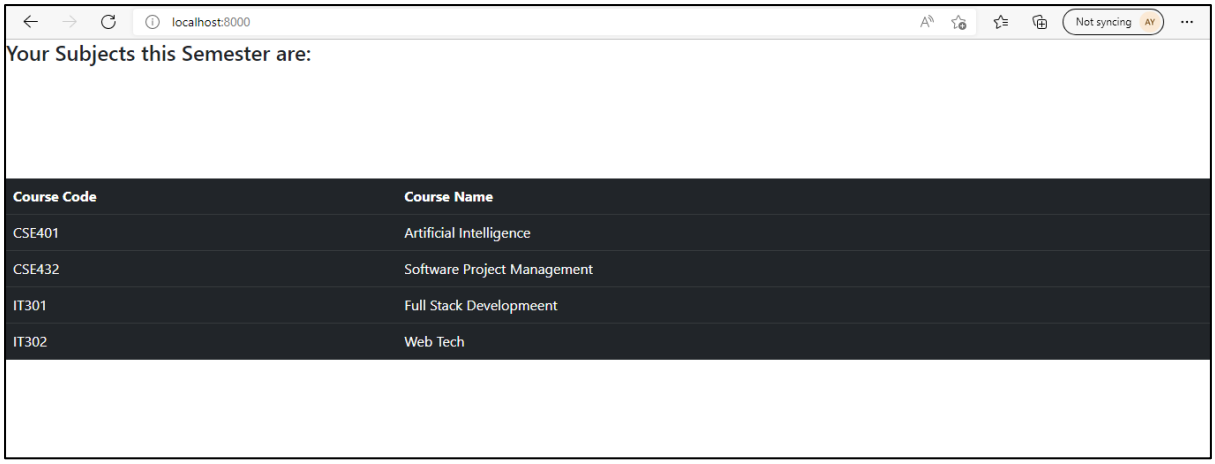
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <!-- CSS only -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
<!-- JavaScript Bundle with Popper -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p" crossorigin="anonymous"></script>
<title>Document</title>
</head>
<body>
    <h4>Your Subjects this Semester are:</h4>
    <table class="table table-dark">
    <tbody>
        <tr>
            <td><b>Course Code</b></td>
            <td><b>Course Name</b></td>
        </tr>
        <br>
        <% records.forEach(function(row) {%>
        <tr>
          <td><%= row.course_id %></td>
          <td><%= row.course_name %></td>
        </tr><br>
        <% }) %>
        </tbody>
    </table>
</body>
</html>
```

(ii) index.js

```
var express = require('express')
var MongoClient = require('mongodb').MongoClient;
const ejs = require('ejs');
var app = express()
var database
app.set('view engine', 'ejs');
app.get('/',(req,res)=>
{
    database.collection('course').find({}).toArray((err,result)=>
    {
        if(err) throw err
                res.render('home',
   {
       name:’Gaura’,
       records:result
   });
    })
})
app.listen(8000, () => {
MongoClient.connect('mongodb://localhost:27017',{useNewUrlParser:true},(error,result) =>
{
        if(error) throw error
        database = result.db('college_database')
        console.log('Connected to College database')
    })
});
```

Output:


localhost:8000
Not syncing
Your Subjects this Semester are:
Course Code
Course Name
CSE401
Artificial Intelligence
CSE432
Software Project Management
IT301
Full Stack Developmeent
IT302
Web Tech

Experiment 10

Objective: Write a program demonstrating two-way databinding in Angular JS.

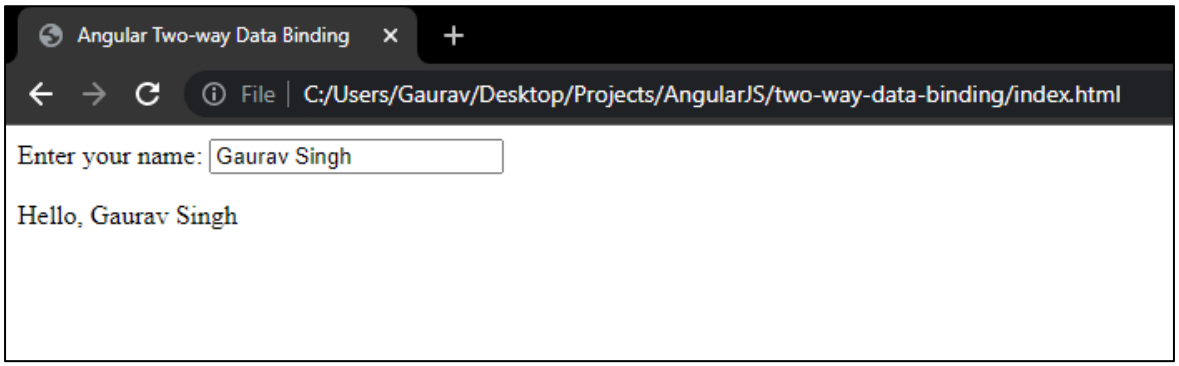
Software Used: Visual Studio Code

Program:

1. index.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">
    <title>Angular Two-way Data Binding</title>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
    <script>
        var app = angular.module("myModule", []).controller("myController", function ($scope)
{
            $scope.user = "";
        });
    </script>
</head>
<body ng-app="myModule">
    <div ng-controller="myController">
        <label for="name">Enter your name: </label>
        <input ng-model="user" id="name">

        <p>Hello, {{user}}</p>
    </div>
</body>
</html>
```

Output:


Angular Two-way Data Binding
File | C:/Users/Gaurav/Desktop/Projects/AngularJS/two-way-data-binding/index.html
Enter your name: Gaurav Singh
Hello, Gaurav Singh

Experiment 11

Objective: Write a program demonstrating events in Angular JS.

Software Used: Visual Studio Code

Program:

1. index.html

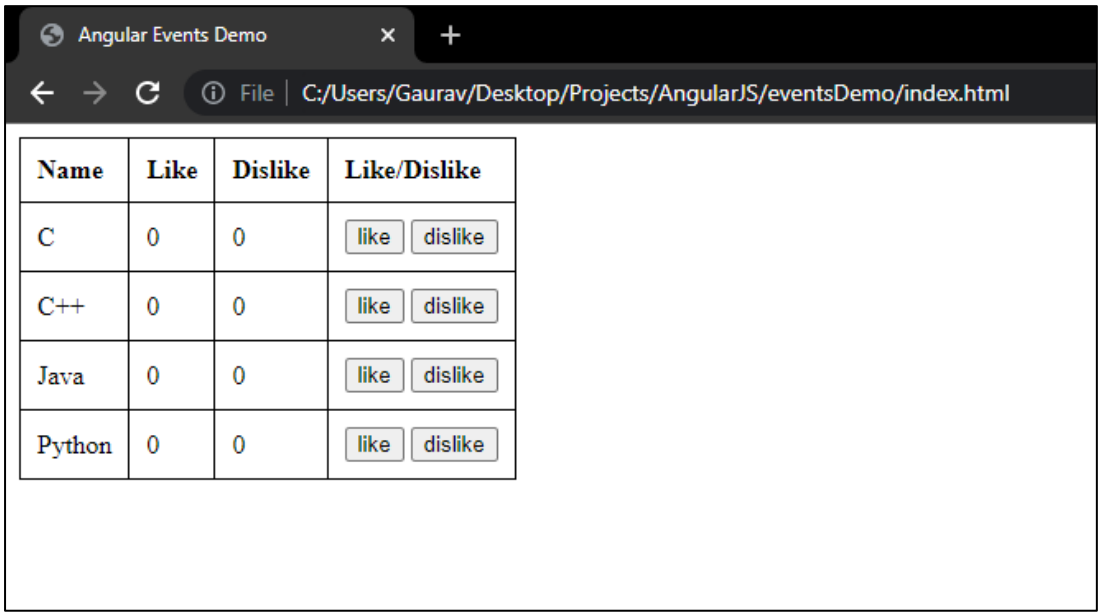
```
<!DOCTYPE html>
<html lang="en" ng-app="myModule">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Angular Events Demo</title>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
    <script src="scripts.js"></script>
    <style>
        table,
        th,
        td {
            border: 1px solid;
            padding: 10px;
        }

        table {
            border-collapse: collapse;
        }

        th {
            text-align: left;
```

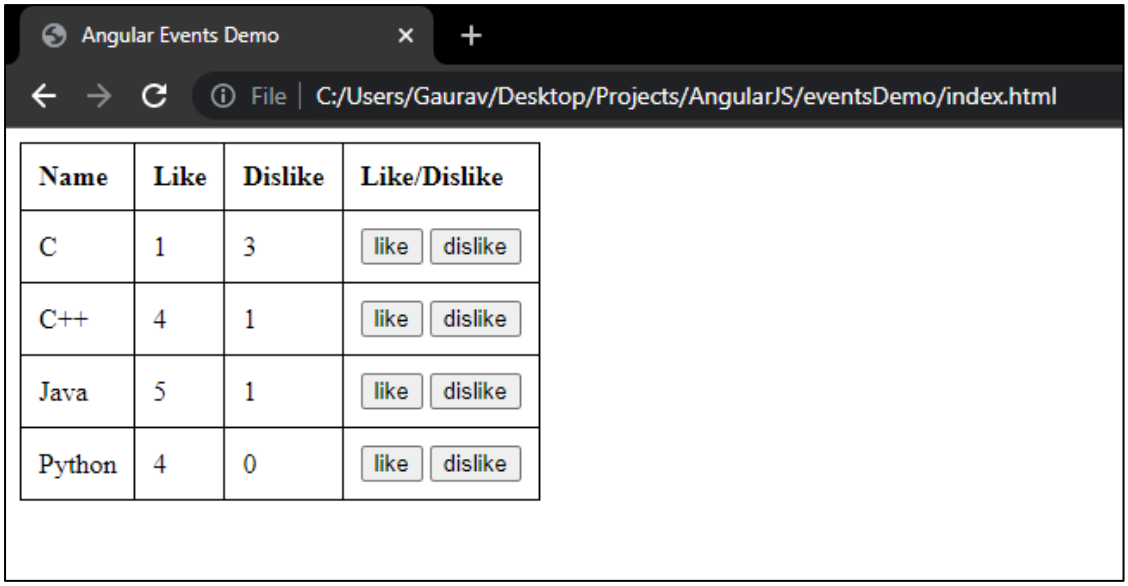
```
        }
    </style>
</head>

<body>
    <table ng-controller="myController">
        <thead>
            <tr>
                <th>Name</th>
                <th>Like</th>
                <th>Dislike</th>
                <th>Like/Dislike</th>
            </tr>
        </thead>
        <tbody>
            <tr ng-repeat="technology in technologies">
                <td>{{technology.name}}</td>
                <td>{{technology.likes}}</td>
                <td>{{technology.dislikes}}</td>
                <td>
                    <input type="button" value="like" ng-click="incrementLikes(technology)">
                    <input type="button" value="dislike" ng-
click="incrementDislikes(technology)">
                </td>
            </tr>
        </tbody>
    </table>
</body>
</html>
```

2. script.js

```
/// <reference path="angular.min.js">
var myController = function ($scope) {
    var technologies = [
        { name: "C", likes: 0, dislikes: 0 },
        { name: "C++", likes: 0, dislikes: 0 },
        { name: "Java", likes: 0, dislikes: 0 },
        { name: "Python", likes: 0, dislikes: 0 },
    ];
    $scope.technologies = technologies;
    $scope.incrementLikes = function (technology) {
        technology.likes++;
    };
    $scope.incrementDislikes = function (technology) {
        technology.dislikes++;
    };
};
var app = angular.module("myModule", []).controller("myController", myController);
```

Output:




Angular Events Demo
File
C:/Users/Gaurav/Desktop/Projects/AngularJS/eventsDemo/index.html
Name
Like
Dislike
Like/Dislike
C
1
3
like
dislike
C++
4
1
like
dislike
Java
5
1
like
dislike
Python
4
0
like
dislike

Experiment 12

Objective: Write a program to demonstrate the use of filters in Angular JS.

Software Used: Visual Studio Code

Program:

1. index.html

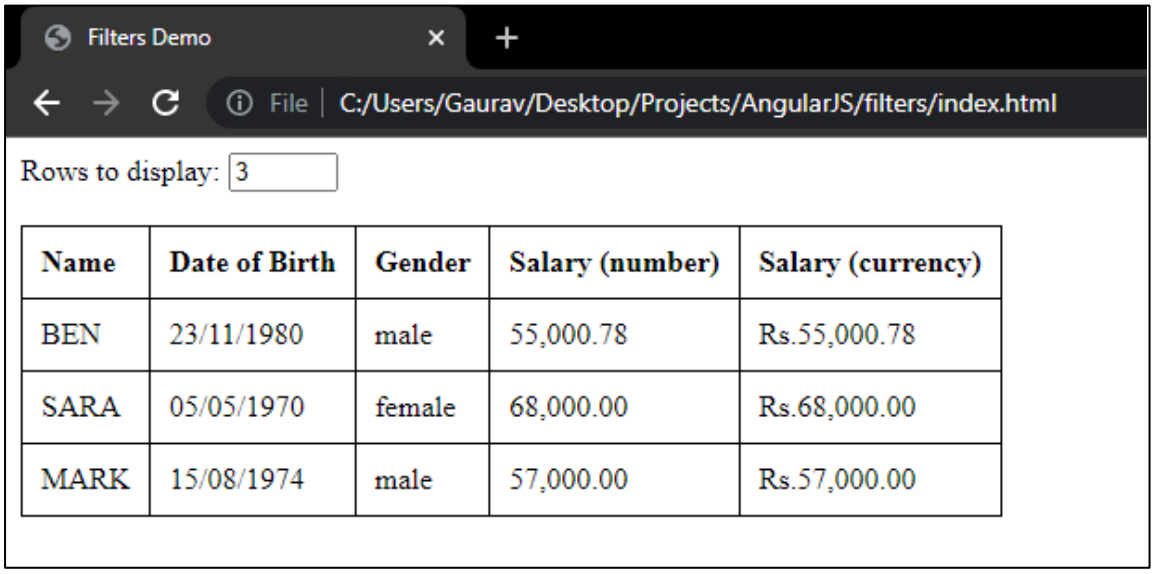
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Filters Demo</title>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
    <script src="./scripts.js"></script>
    <style>
        table,
        th,
        td {
            border: 1px solid;
            padding: 10px;
        }
        table {
            border-collapse: collapse;
        }
        th {
            text-align: left;
        }
    </style>
```

```
</head>
<body ng-app="myModule">
  <div ng-controller="myController">
    Rows to display:
    <input type="number" step="1" min="0" max="5" ng-model="rowLimit" />
    <br />
    <br />
    <table>
      <thead>
        <tr>
          <th>Name</th>
          <th>Date of Birth</th>
          <th>Gender</th>
          <th>Salary (number)</th>
          <th>Salary (currency) </th>
        </tr>
      </thead>
      <tbody>
        <tr ng-repeat="employee in employees | limitTo:rowLimit">
          <td>{{employee.name | uppercase}}</td>
          <td>{{employee.dob | date:"dd/MM/yyyy"}}</td>
          <td>{{employee.gender | lowercase}}</td>
          <td>{{employee.salary | number:2}}</td>
          <td>{{employee.salary | currency:"Rs.":2}}</td>
        </tr>
      </tbody>
    </table>
  </div>
</body>
</html>
```

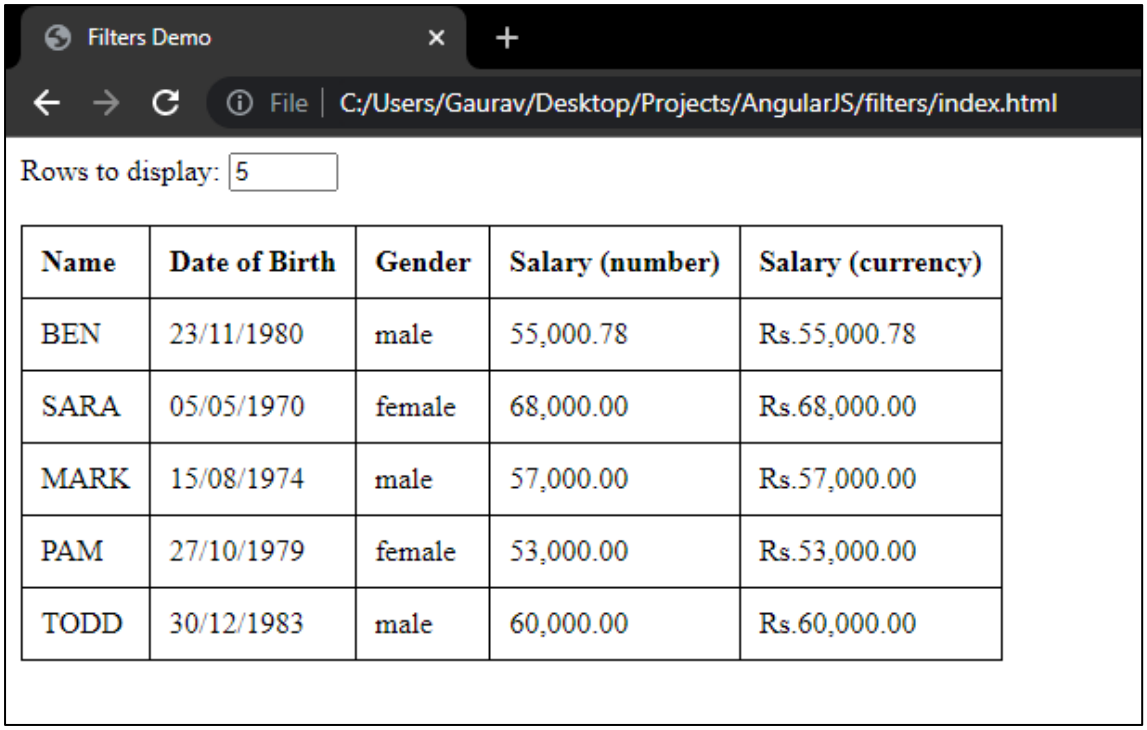
2. script.js

```
/// <reference path="angular.min.js">
var myController = function ($scope) {
    var employees = [
        { name: "Ben", dob: new Date("November 23, 1980"), gender: "Male", salary: 55000.78
},
        { name: "Sara", dob: new Date("May 05, 1970"), gender: "Female", salary: 68000 },
        { name: "Mark", dob: new Date("August 15, 1974"), gender: "Male", salary: 57000 },
        { name: "Pam", dob: new Date("October 27, 1979"), gender: "Female", salary: 53000 },
        { name: "Todd", dob: new Date("December 30, 1983"), gender: "Male", salary: 60000 },
    ];
    $scope.employees = employees;
    $scope.rowLimit = 3;
};
var app = angular.module("myModule", []).controller("myController", myController);
```

Output:



Name	Date of Birth	Gender	Salary (number)	Salary (currency)
BEN	23/11/1980	male	55,000.78	Rs.55,000.78
SARA	05/05/1970	female	68,000.00	Rs.68,000.00
MARK	15/08/1974	male	57,000.00	Rs.57,000.00


Filters Demo
File
C:/Users/Gaurav/Desktop/Projects/AngularJS/filters/index.html
Rows to display: 5
Name
Date of Birth
Gender
Salary (number)
Salary (currency)
BEN
23/11/1980
male
55,000.78
Rs.55,000.78
SARA
05/05/1970
female
68,000.00
Rs.68,000.00
MARK
15/08/1974
male
57,000.00
Rs.57,000.00
PAM
27/10/1979
female
53,000.00
Rs.53,000.00
TODD
30/12/1983
male
60,000.00
Rs.60,000.00