INFORMATION TECHNOLOGIES

SUBMITTED BY :-

DEV GUPTA

10839

B.Sc (Hons) COMPUTER SCIENCE

IIIrd YEAR (A)

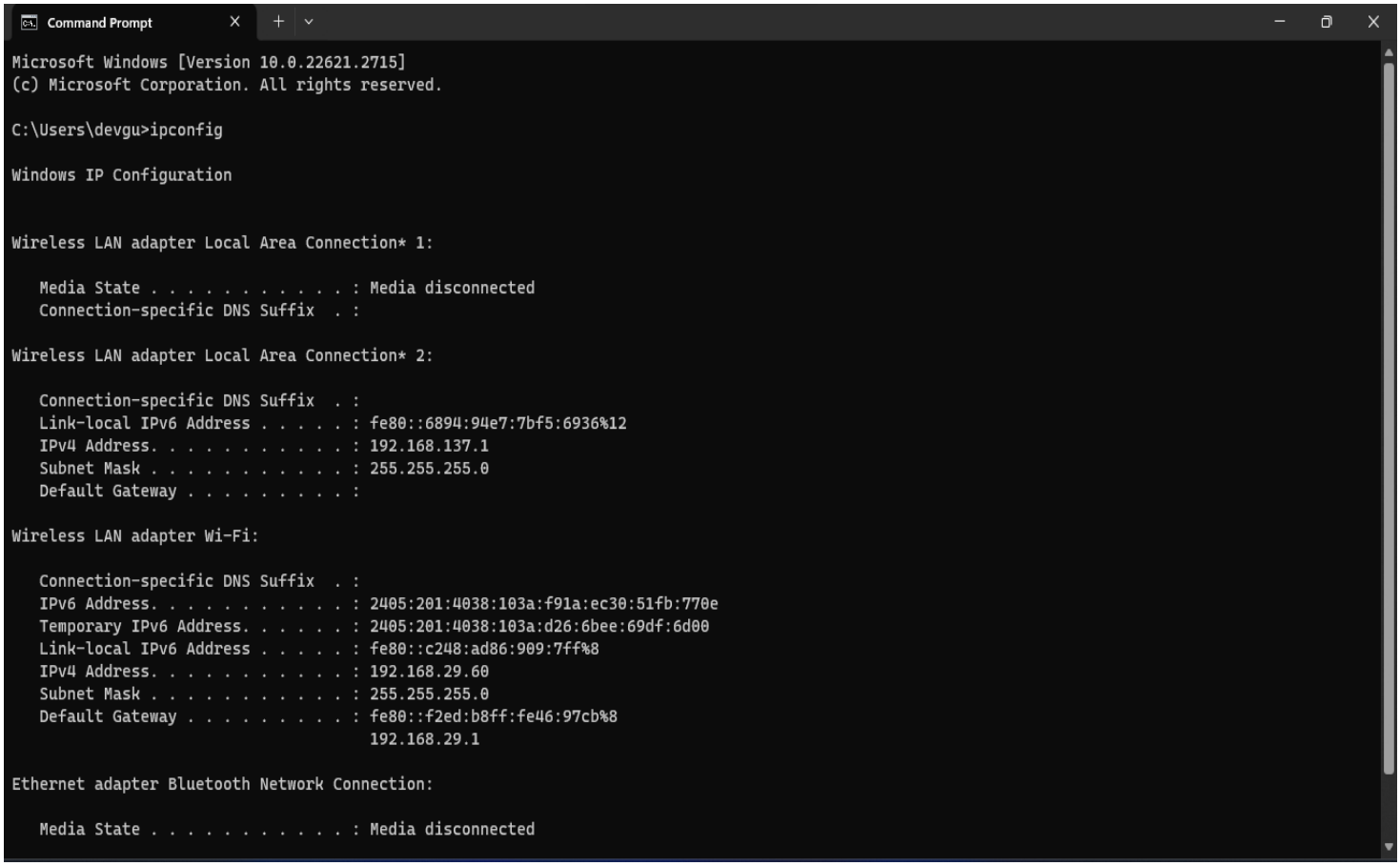
PRACTICAL

FILE

INDEX

PRACTICAL 1

Display your systems IP Address, Subnet mask using ipconfig, and find out the network address and the maximum number of systems possible on your network and range of IP addresses available to these systems.



IP ADDRESS : 192.168.29.60

SUBNET MASK : 255.255.255.0

→ NETWORK ADDRESS : 192.168.29.0

BROADCAST ADDRESS : 192.168.29.255

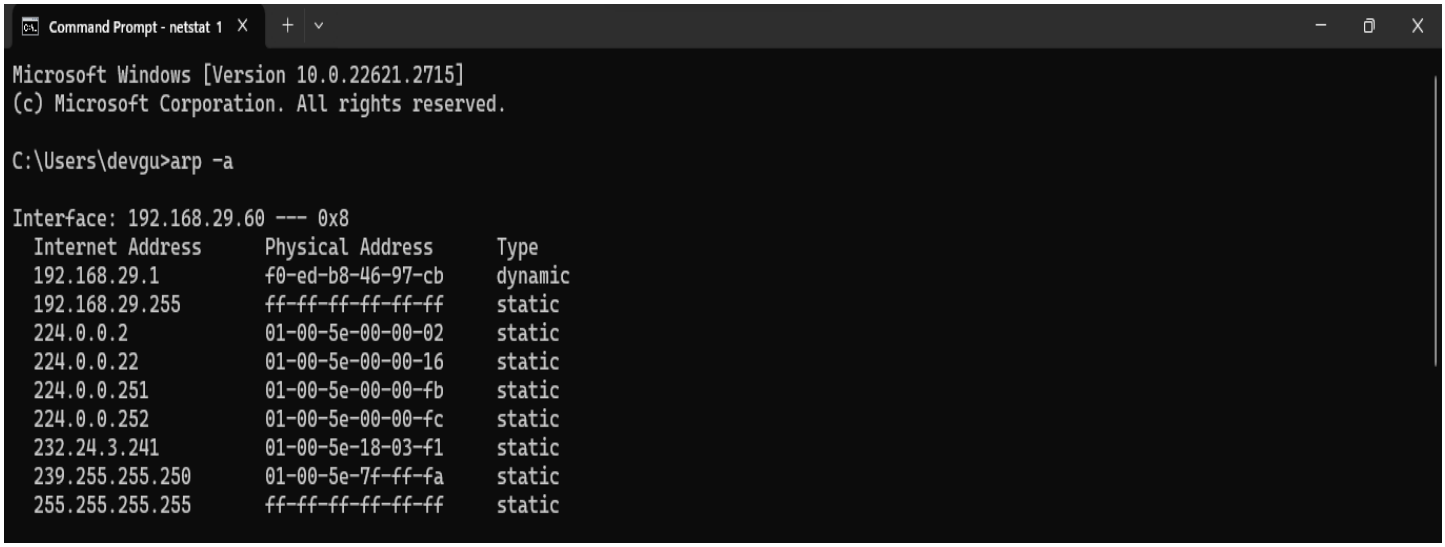
→ RANGE OF IP ADDRESSES AVAILABLE :

192.168.29.1 TO 192.168.29.254

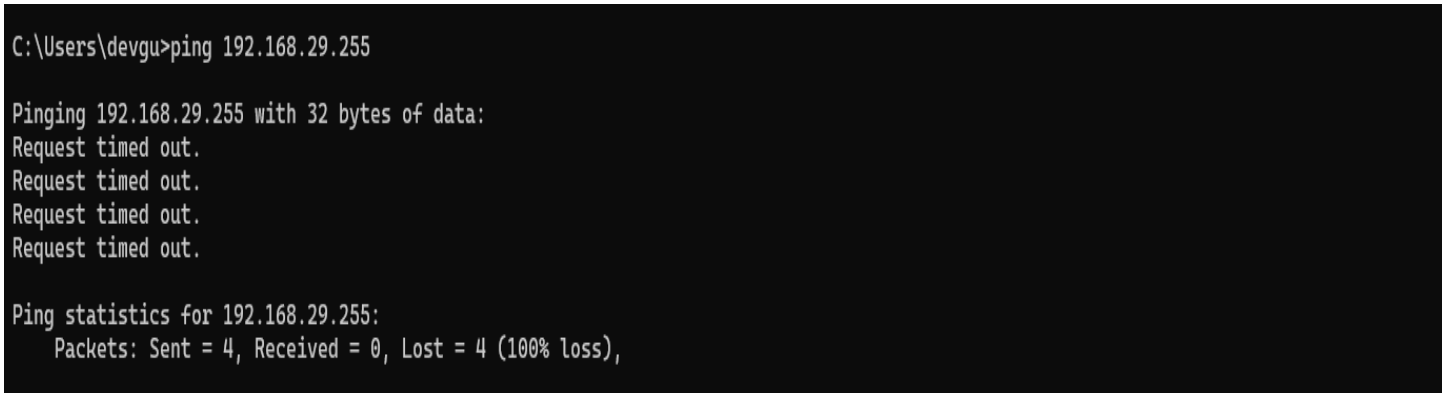
PRACTICAL 2

With help of ping, check if you are connected to other systems of your network and find the route to connect to that system using tracert. List all the processes which are using ports for TCP protocol.

→ arp -a



→ ping 192.168.29.255



→ tracert 192.168.29.255

```
C:\Users\devgu>tracert 192.168.29.255

Tracing route to 192.168.29.255 over a maximum of 30 hops

  1     *        *        *     Request timed out.
  2     *        *        *     Request timed out.
  3     *        *        *     Request timed out.
  4     *        *        *     Request timed out.
  5     *        *        *     Request timed out.
  6     *        *        *     Request timed out.
  7     *        *        *     Request timed out.
  8     *        *        *     Request timed out.
  9     *        *        *     Request timed out.
 10     *        *        *     Request timed out.
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15     *        *        *     Request timed out.
 16     *        *        *     Request timed out.
 17     *        *        *     Request timed out.
 18     *        *        *     Request timed out.
 19     *        *        *     Request timed out.
 20     *        *        *     Request timed out.
 21     *        *        *     Request timed out.
 22     *        *        *     Request timed out.
 23     *        *        *     Request timed out.
 24     *        *        *     Request timed out.
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
 30     *        *        *     Request timed out.

Trace complete.
```

→ netstat 192.168.29.255

```
C:\Users\devgu>netstat 192.168.29.255

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:49671        DEVTA:49672            ESTABLISHED
  TCP    127.0.0.1:49672        DEVTA:49671            ESTABLISHED
  TCP    127.0.0.1:49673        DEVTA:49674            ESTABLISHED
  TCP    127.0.0.1:49674        DEVTA:49673            ESTABLISHED
  TCP    192.168.29.60:49411    20.198.118.190:https   ESTABLISHED
  TCP    192.168.29.60:49815    152.195.38.76:http     CLOSE_WAIT
  TCP    192.168.29.60:50213    20.198.119.84:https    ESTABLISHED
  TCP    192.168.29.60:50217    20.49.99.116:8883      ESTABLISHED
  TCP    192.168.29.60:50235    1drv:https             ESTABLISHED
  TCP    192.168.29.60:50243    a184-85-220-57:https   ESTABLISHED
  TCP    192.168.29.60:50244    ec2-34-208-129-82:https  TIME_WAIT
  TCP    192.168.29.60:50245    52.152.90.172:https    ESTABLISHED
  TCP    192.168.29.60:50246    ec2-34-208-129-82:https  ESTABLISHED
  TCP    192.168.29.60:50251    ec2-34-208-129-82:https  TIME_WAIT
  TCP    192.168.29.60:50253    20.231.121.79:http     SYN_SENT
  TCP    192.168.29.60:50255    20.69.137.228:https    ESTABLISHED
  TCP    192.168.29.60:50256    40.65.170.106:https    ESTABLISHED
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49812  [2600:1f14:62a:de81:b848:82ee:2416:447e]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49813  [2620:108:700f::3420:b173]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49816  [2405:200:1604:600:49:44:142:8a]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49817  [2405:200:1604:600:49:44:142:8a]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49818  [2405:200:1604:600:49:44:142:8a]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49819  [2405:200:1604:600:49:44:142:8a]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49820  [2405:200:1604:600:49:44:142:8a]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49821  [2405:200:1604:600:49:44:142:8a]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49822  [2620:108:700f::3420:b173]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49823  [2620:108:700f::3420:b173]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49824  [2405:200:160f:600:49:44:142:b2]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49825  [2405:200:160f:600:49:44:142:b2]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49826  [2405:200:160f:600:49:44:142:b2]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:49828  [2405:200:160f:600:49:44:142:b2]:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:50067  g2600-140f-0005-0000-0000-0000-17d9-6e73:https  CLOSE_WAIT
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:50082  [2620:1ec:90c::254]:https  ESTABLISHED
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:50218  [2603:1063:2202:14::3]:https  ESTABLISHED
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:50250  whatsapp-chatd-edge6-shv-02-del1:http  ESTABLISHED
  TCP    [2405:201:4038:103a:8d92:fdb9:414a:ceae]:50254  [2620:1ec:42::132]:https  ESTABLISHED
```

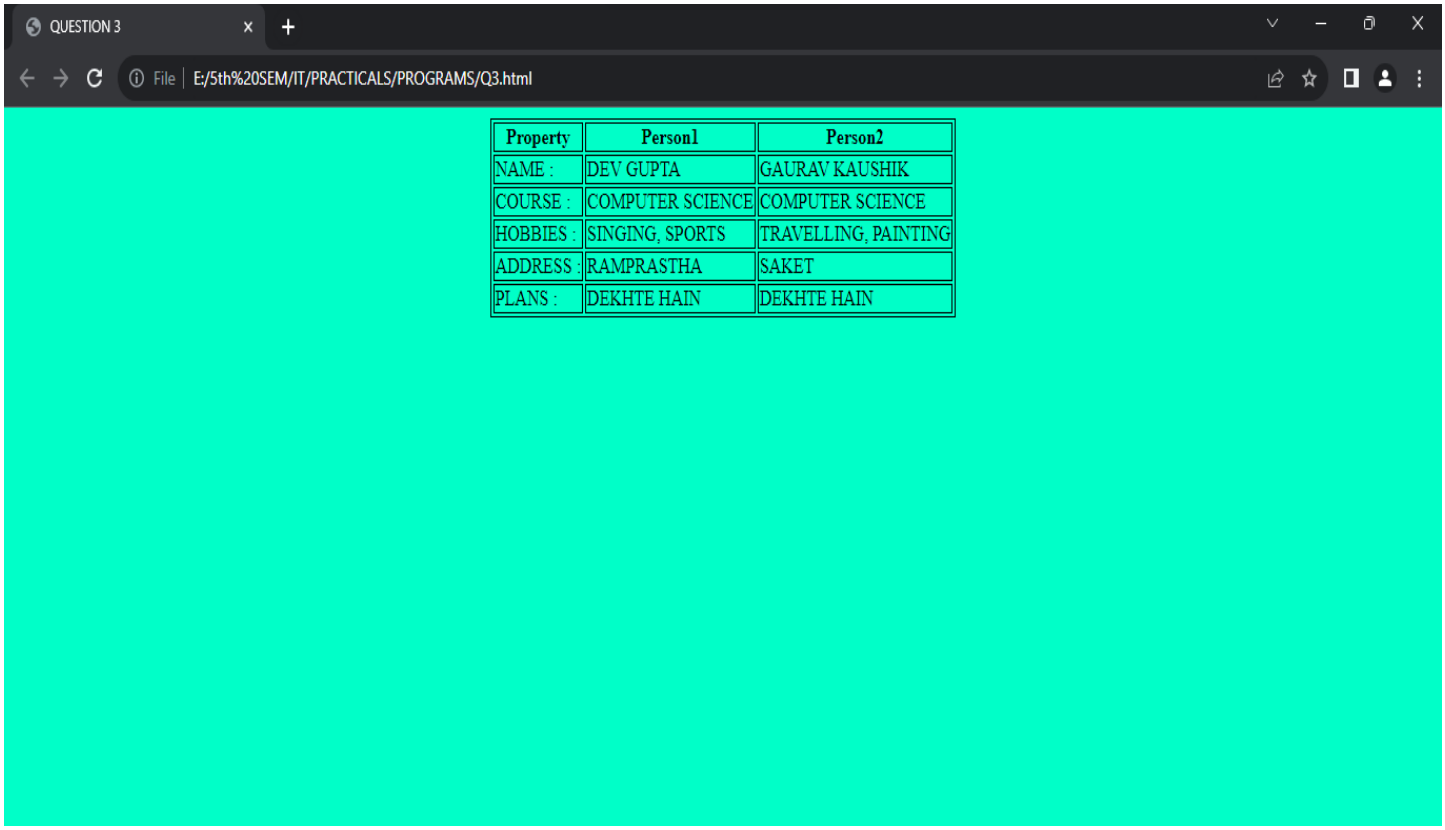
PRACTICAL 3

Create an HTML page that shows information about you, your course, hobbies, address, and your plans. Use CSS for styling of HTML page so that looks nice.

CODE

```
<html>
    <head>
        <title>
            QUESTION 3
        </title>
    </head>
    <style>
        table,th,td{border: 1px solid black;}
        body{background-color: rgb(0, 255, 200);}
    </style>
    <center>
    <table>
        <tr>
            <th>Property</th>
            <th>Person1</th>
            <th>Person2</th>
        </tr>
        <tr>
        <td>NAME :</td>
        <td> DEV GUPTA </td>
        <td> GAURAV KAUSHIK </td>
        </tr>
        <tr>
            <td>COURSE :</td>
            <td>COMPUTER SCIENCE</td>
            <td>COMPUTER SCIENCE</td>
        </tr>
        <tr>
            <td>HOBBIES :</td>
            <td>SINGING, SPORTS</td>
            <td>TRAVELLING, PAINTING</td>
        </tr>
        <tr>
            <td>ADDRESS :</td>
            <td> RAMPRASTHA </td>
            <td> SAKET </td>
        </tr>
        <tr>
            <td>PLANS :</td>
            <td>DEKHTE HAIN</td>
            <td>DEKHTE HAIN</td>
        </tr>
    </table>
</center>
</html>
```

OUTPUT



Property	Person1	Person2
NAME :	DEV GUPTA	GAURAV KAUSHIK
COURSE :	COMPUTER SCIENCE	COMPUTER SCIENCE
HOBBIES :	SINGING, SPORTS	TRAVELLING, PAINTING
ADDRESS :	RAMPRASTHA	SAKET
PLANS :	DEKHTE HAIN	DEKHTE HAIN

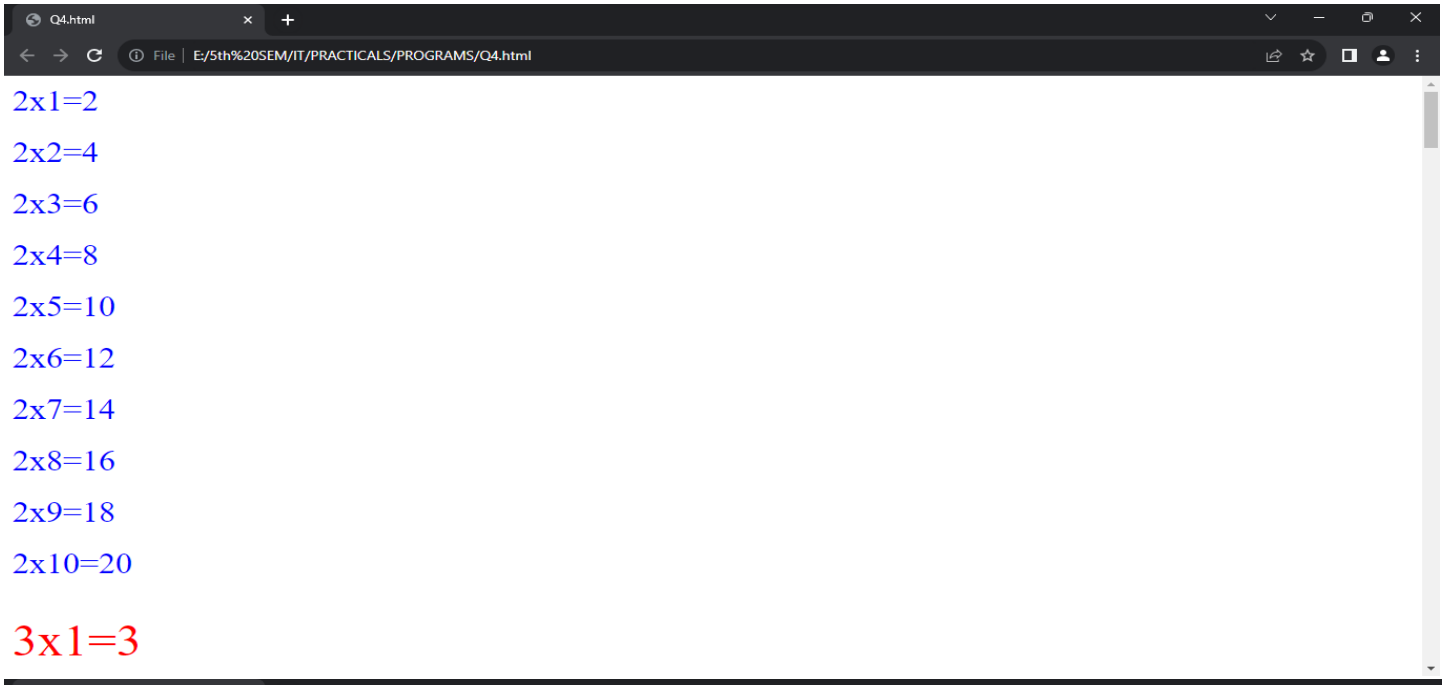
PRACTICAL 4

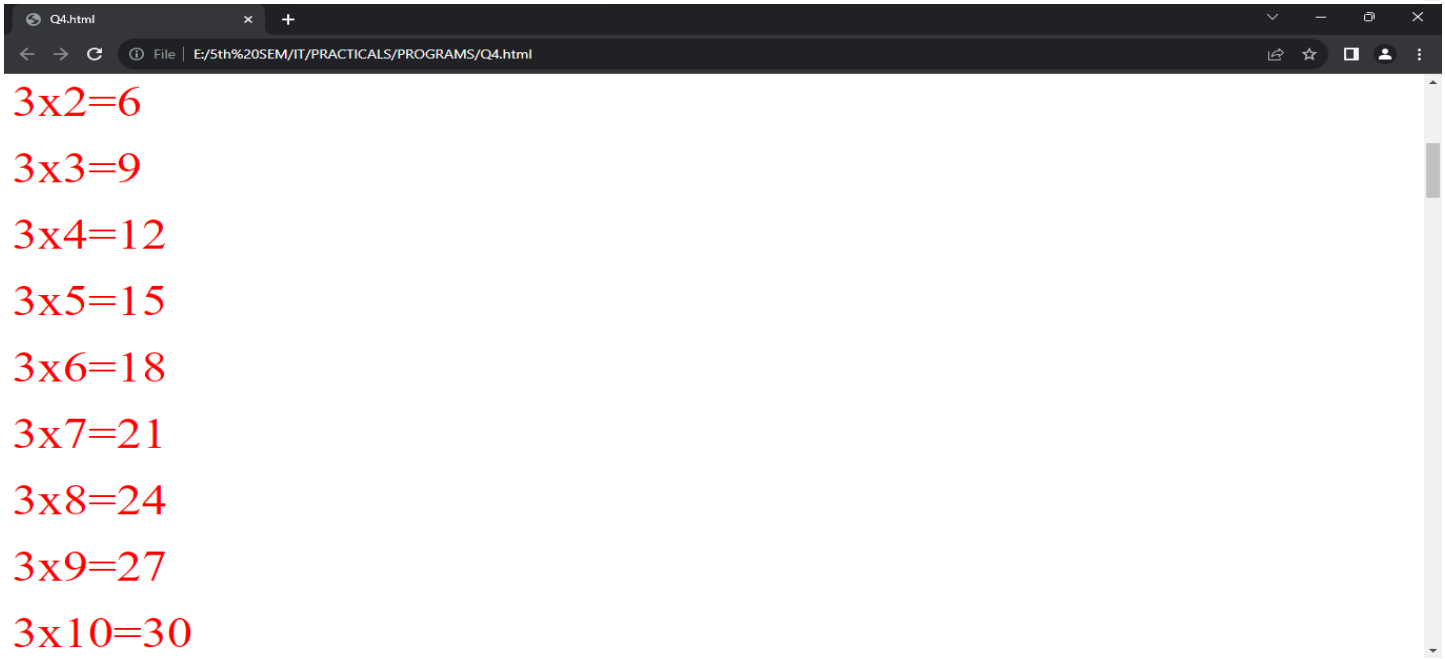
Create an HTML page with the sole purpose to show multiplication tables of 2 to 10 (row-wise) created by JavaScript. Initially, the page is blank. With help of setInterval function print a row every 5 seconds in different colors and increasing font size.

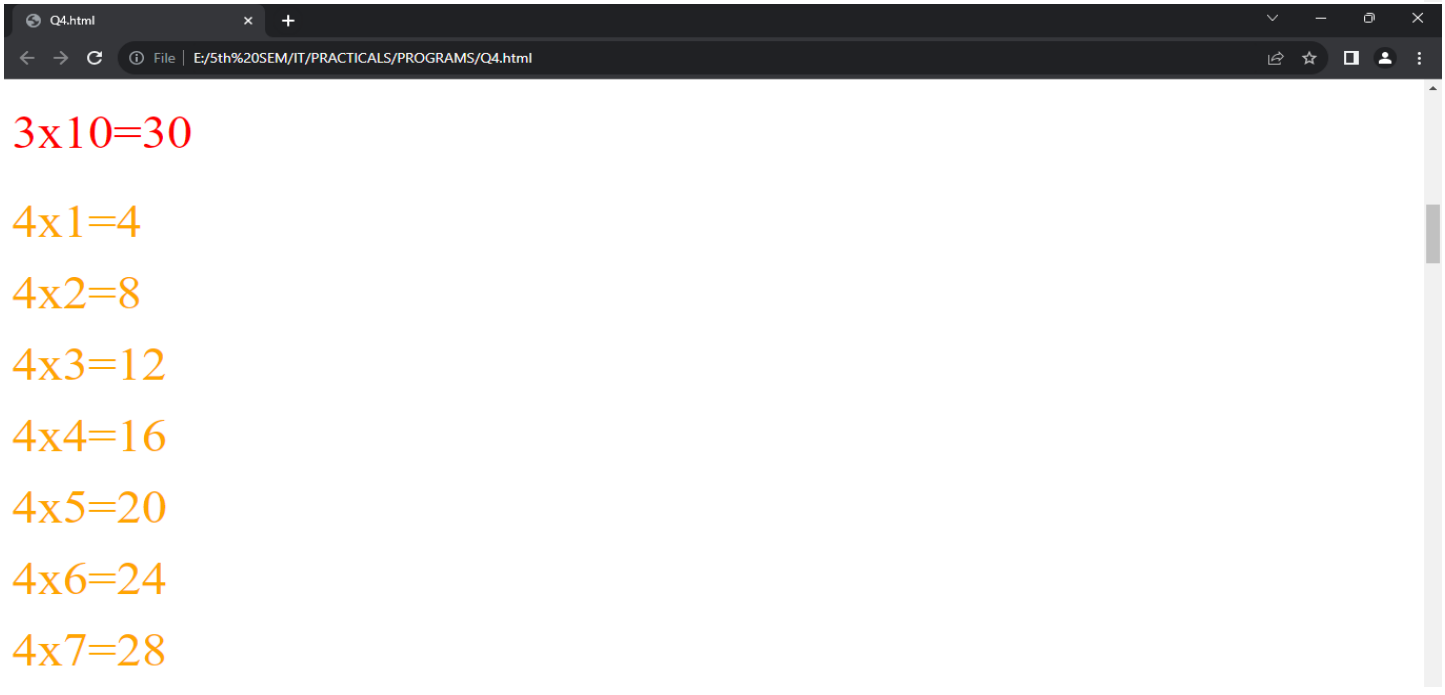
CODE

```
<html>
    <head>
        <title>
            QUESTION 4
        </title>
    </head>
    <style>
        body{background-color: aquamarine;
        }
    </style>
    <body>
        <h1>TABLES</h1>
        <button>START</button>
    <script>
        var time;
        var size=5;
        var count=2;
        var btn=document.querySelector('button');
        btn.addEventListener('click',def);
        function def(){
            time=setInterval(abc,1000);
        }
        function abc(){
            var color=["red","blue","green","maroon","orange","cyan","black","sky
blue","hotpink","babypink","oral"];
            var rand=Math.floor((Math.random()*10));
            for(var i=1;i<=10;i++){
                var table=(count+"x"+i+"="+count*i);
                document.write(table.fontcolor(color[rand]).fontsize(size+1));
                document.write("<br><br>");
            }
            document.write("<br>");
            size+=1;
            if(count==10){
                clearInterval(time);
            }
            count+=1;
        }
    </script>
    </body>
</html>
```

OUTPUT


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
2x1=2
2x2=4
2x3=6
2x4=8
2x5=10
2x6=12
2x7=14
2x8=16
2x9=18
2x10=20
3x1=3


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
3x2=6
3x3=9
3x4=12
3x5=15
3x6=18
3x7=21
3x8=24
3x9=27
3x10=30


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
3x10=30
4x1=4
4x2=8
4x3=12
4x4=16
4x5=20
4x6=24
4x7=28

4x8=32

4x9=36

4x10=40

5x1=5

5x2=10

5x3=15

5x4=20

5x5=25

5x5=25

5x6=30

5x7=35

5x8=40

5x9=45

5x10=50

6x1=6

6x2=12

6x3=18

6x4=24

6x5=30

6x6=36

6x7=42

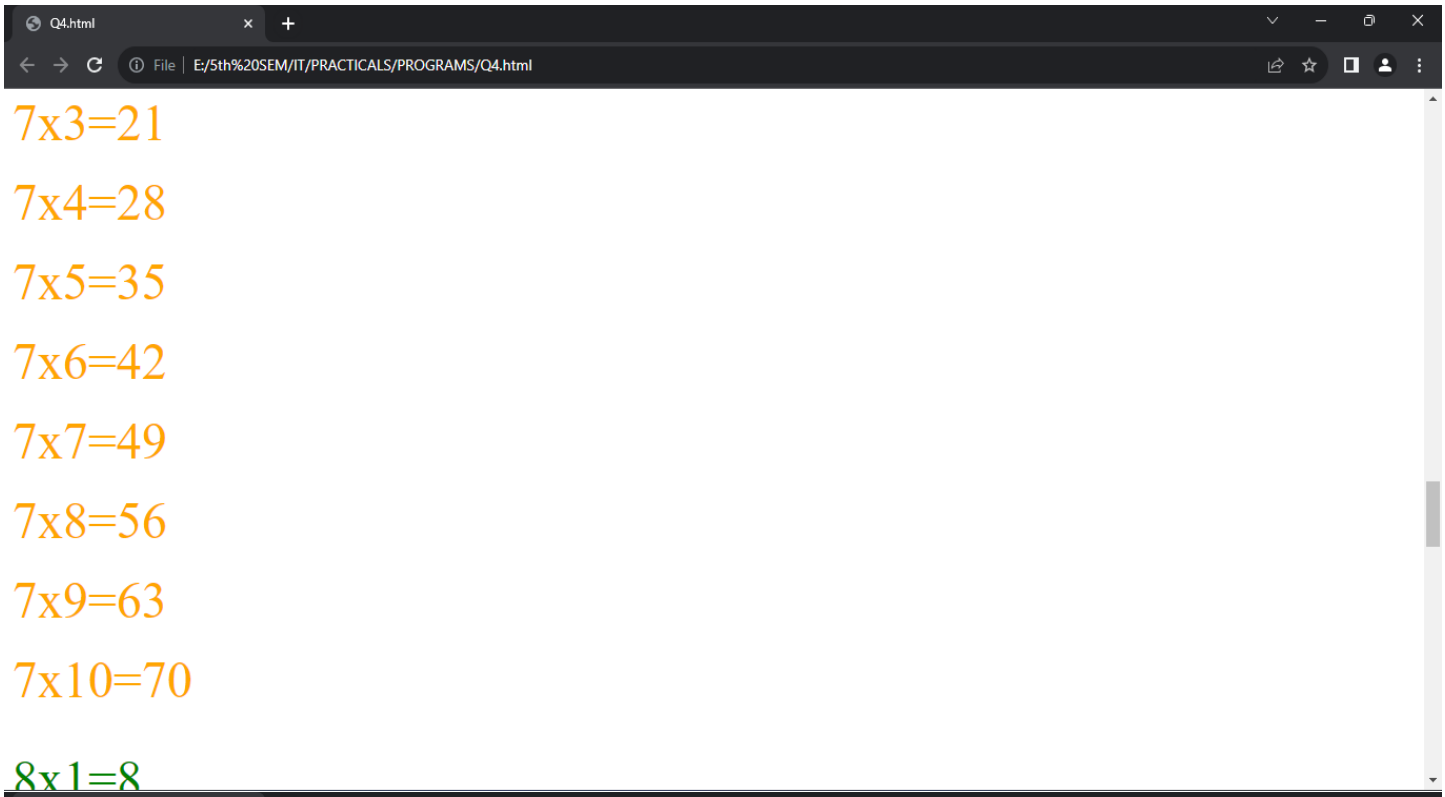
6x8=48

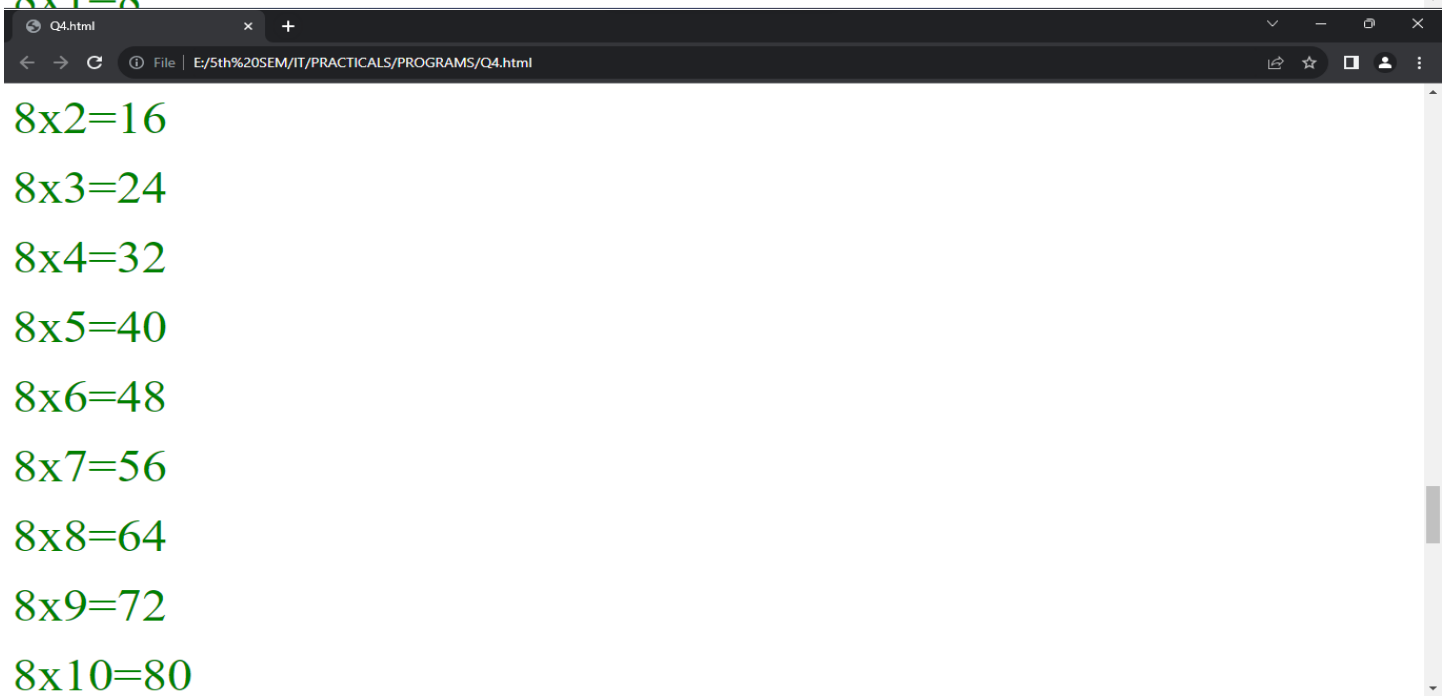
6x9=54

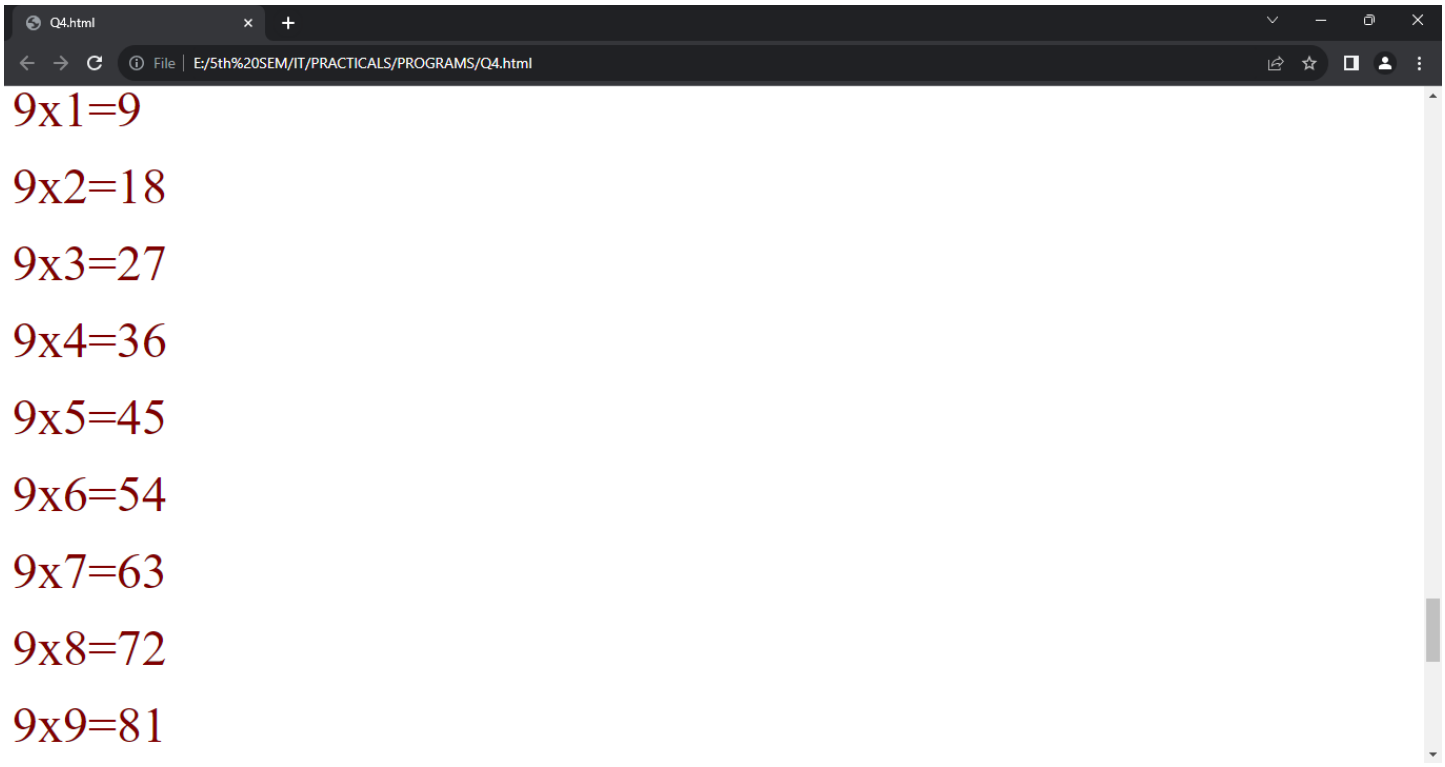
6x10=60

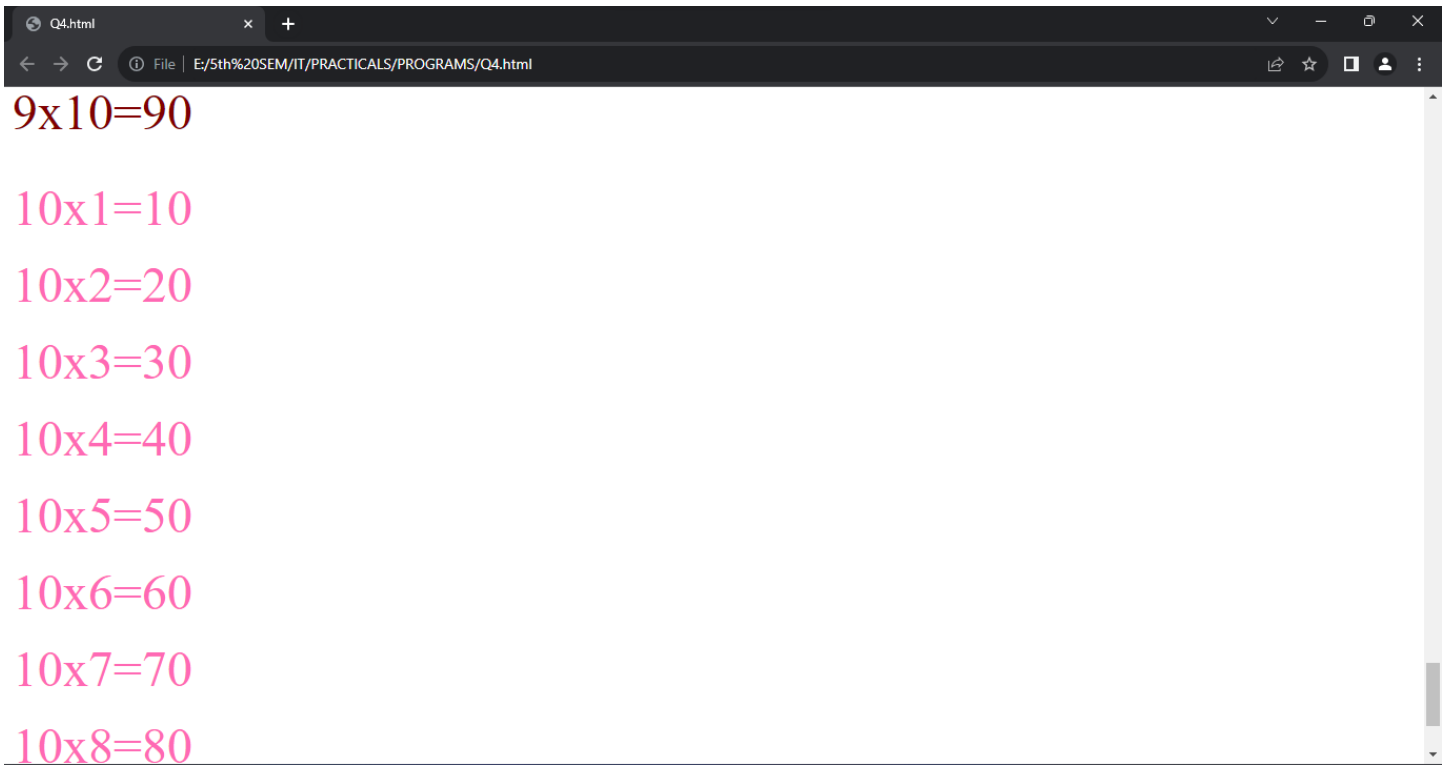
7x1=7

7x2=14


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
7x3=21
7x4=28
7x5=35
7x6=42
7x7=49
7x8=56
7x9=63
7x10=70
8x1=8


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
8x2=16
8x3=24
8x4=32
8x5=40
8x6=48
8x7=56
8x8=64
8x9=72
8x10=80


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
9x1=9
9x2=18
9x3=27
9x4=36
9x5=45
9x6=54
9x7=63
9x8=72
9x9=81


Q4.html
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q4.html
9x10=90
10x1=10
10x2=20
10x3=30
10x4=40
10x5=50
10x6=60
10x7=70
10x8=80

PRACTICAL 5

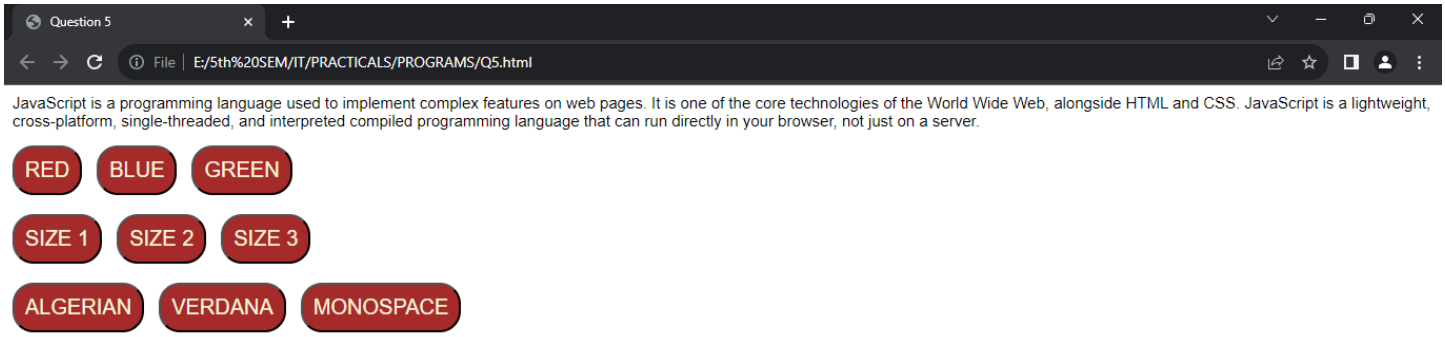
Create an HTML page with a paragraph written on it and under which 9 buttons are placed in a 3X3 grid. The first row is for buttons labeled with colors names Red, Green, and Blue, the second row with numbers 10, 20, 30, and the third row with different font names. Click event of each of the buttons should make the appropriate change in the style of paragraph.

CODE

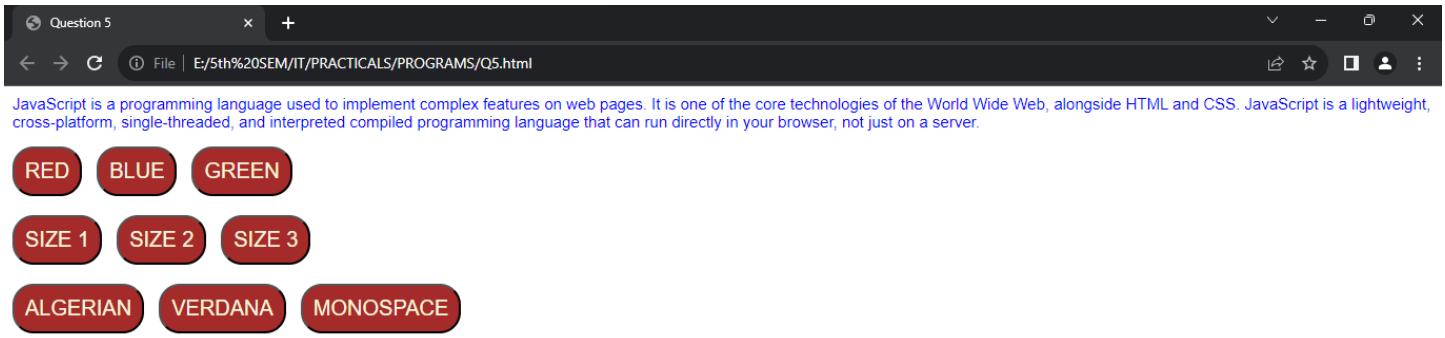
```
<html>
    <head>
        <title>
            Question 5
        </title>
        <SCRIPT>
            function f1(){
                document.getElementById("0").style.color="red";
            }
            function f2(){
                document.getElementById("0").style.color="blue";
            }
            function f3(){
                document.getElementById("0").style.color="green";
            }
            function f4(){
                document.getElementById("0").style.fontSize="20px";
            }
            function f5(){
                document.getElementById("0").style.fontSize="30px";
            }
            function f6(){
                document.getElementById("0").style.fontSize="40px";
            }
            function f7(){
                document.getElementById("0").style.fontFamily="algerian";
            }
            function f8(){
                document.getElementById("0").style.fontFamily="verdana";
            }
            function f9(){
                document.getElementById("0").style.fontFamily="monospace";
            }
        </SCRIPT>
        <style>
            p{
                font-family: Arial, Helvetica, sans-serif;
                font-size: 15px;
            }
            button{
                background-color: brown;
                font-size: 20px;
                color: beige;
                margin: 0px 10px 0px 0px;
```

```
                padding: 10px;
                border-radius: 20px;
            }
        </style>
    </head>
    <body>
        <p id="0">JavaScript is a programming language used to implement complex features on
web pages.
             It is one of the core technologies of the World Wide Web, alongside HTML and
CSS.
             JavaScript is a lightweight, cross-platform, single-threaded, and interpreted
compiled
              programming language that can run directly in your browser, not just on a
server.
        </p>
            <button  id="1" onclick="f1()">RED</button>
            <button id="2" onclick="f2()">BLUE</button>
            <button id="3" onclick="f3()">GREEN</button><br><br>

            <button id="4" onclick="f4()">SIZE 1</button>
            <button id="5" onclick="f5()">SIZE 2</button>
            <button id="6" onclick="f6()">SIZE 3</button><br><br>

            <button id="7" onclick="f7()">ALGERIAN</button>
            <button id="8" onclick="f8()">VERDANA</button>
            <button id="9" onclick="f9()">MONOSPACE</button><br><br>

    </body>
</html>
```

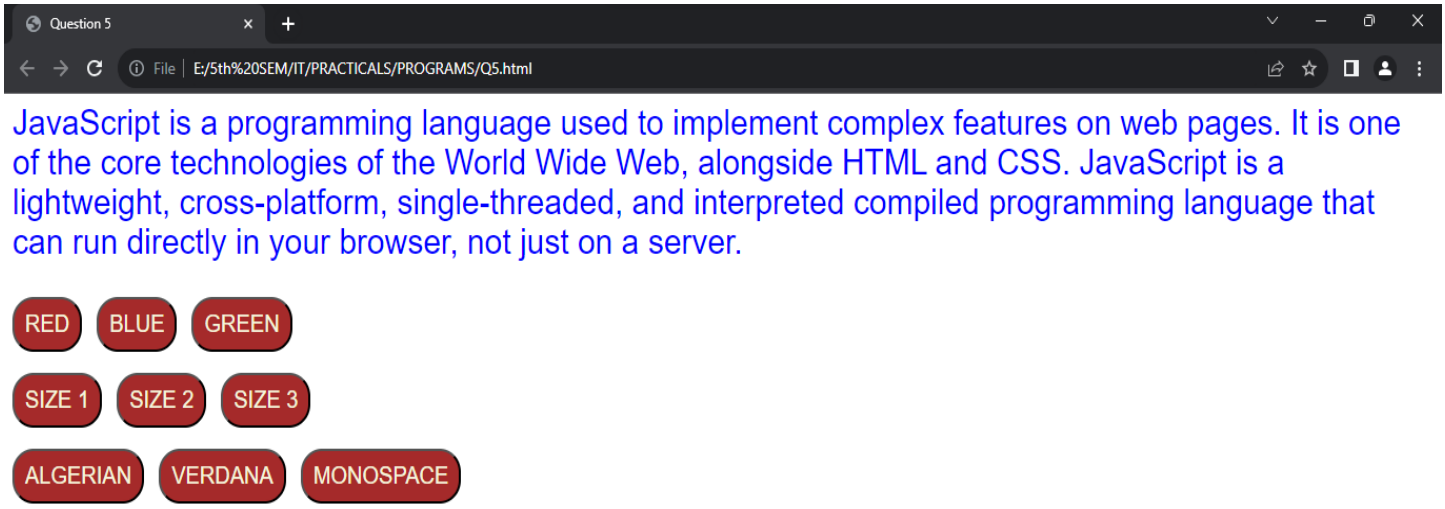
OUTPUT



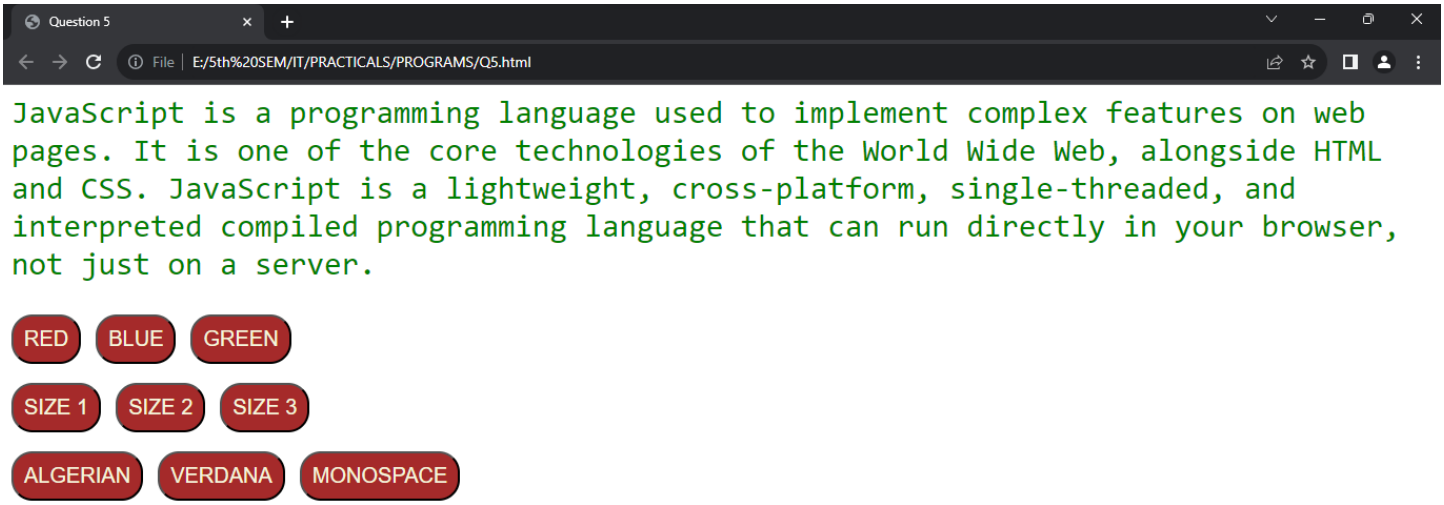
CLICKING “BLUE” BUTTON



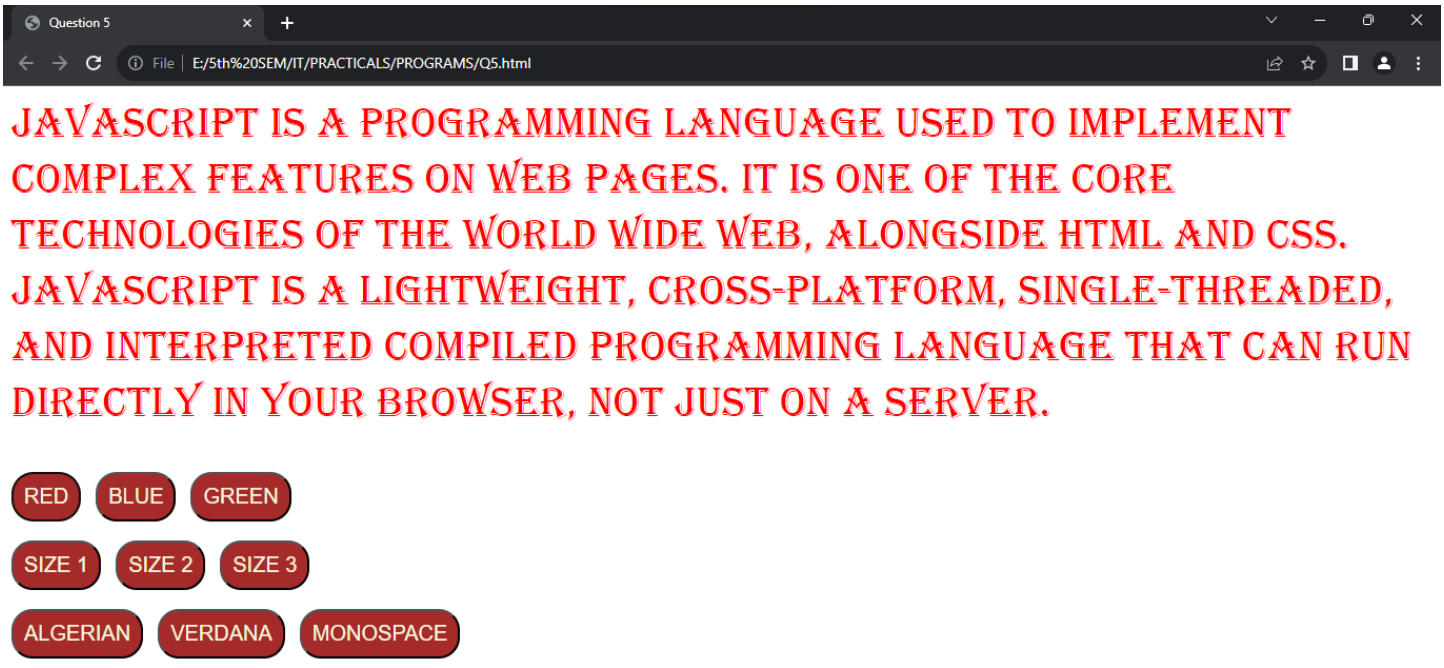
CLICKING “SIZE 2” BUTTON



CLICKING “GREEN” AND “MONOSPACE” BUTTON



CLICKING “RED”, “SIZE3” & “ALGERIAN” BUTTON



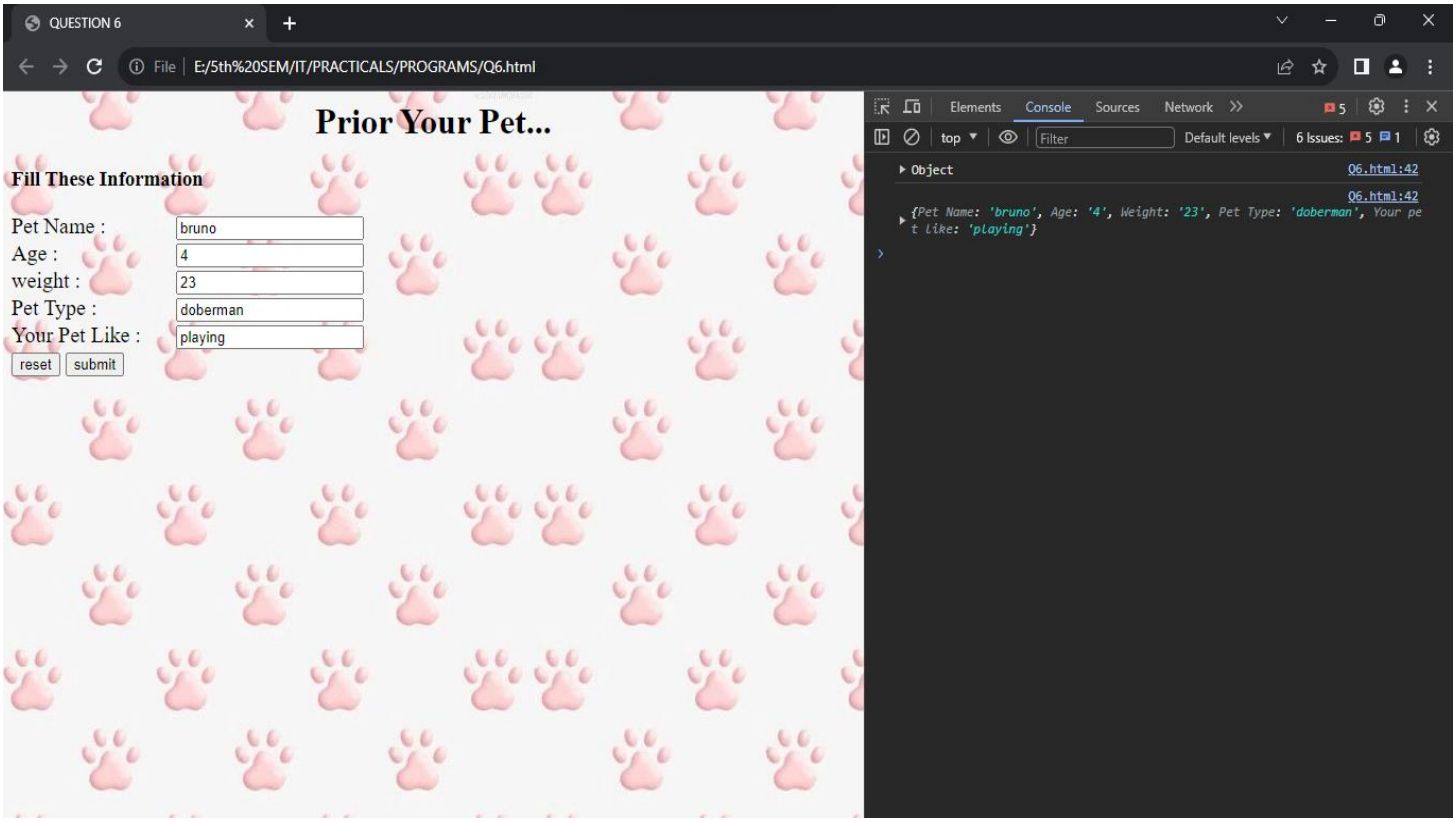
PRACTICAL 6

Create a form that takes data about a pet. The form must be well designed and should accept the, and what it likes most. At the submission of this form create a Pet object in JavaScript filled with these values and log that object and equivalent JSON on the console.

CODE

```
<html>
    <head>
        <title>
            QUESTION 6
        </title>
    </head>
    <style>
        body{background-color:palevioletred;}
        h1{text-align: center;}
        label{width: 150px;
            display: inline-block;
        }
        form{font-size:20px;}
    </style>
    <body background="panje.jpg">

        <h1><b>Prior Your Pet...</b></h1>
        <h3><b>Fill These Information</b></h3>
        <form>
            <label for="pname">Pet Name :</label>
            <input type="text" id="1" name="Pet Name"><br>
            <label for="age">Age :</label>
            <input type="number" id="2" name="Age"><br>
            <label for="weight">weight :</label>
            <input type="number" id="3" name="Weight"><br>
            <label for="ptype">Pet Type :</label>
            <input type="text" id="4" name="Pet Type"><br>
            <label for="like">Your Pet Like :</label>
            <input type="text" id="5" name="Your pet like"><br>
            <input type="reset" value="reset">
            <input type="button" value="submit" onclick="abc()">
        </form>
    <script>
            function abc(){
                var temp=document.getElementsByTagName("input");
                var obj={}
                for(let i=0;i<temp.length-2;i++){
                    let a=temp[i].name;
                    let b=temp[i].value;
                    obj[a]=b;
                }
                console.log(obj);
        }
    </script>
    </body>
</html>
```

OUTPUT


QUESTION 6
File | E:/5th%20SEM/IT/PRACTICALS/PROGRAMS/Q6.html
Prior Your Pet...
Fill These Information
Pet Name : bruno
Age : 4
weight : 23
Pet Type : doberman
Your Pet Like : playing
reset
submit
Elements
Console
Sources
Network
top
Filter
Default levels
6 Issues: 5 1
Object
Q6.html:42
Q6.html:42
{Pet Name: 'bruno', Age: '4', Weight: '23', Pet Type: 'doberman', Your pet Like: 'playing'}

PRACTICAL 7

Store JSON data of few pets that you created in previous practical in a JSON file (copy from console output of previous program to a .json file). Using AJAX, load data from the file and display it in a presentable way using HTML and CSS.

CODE

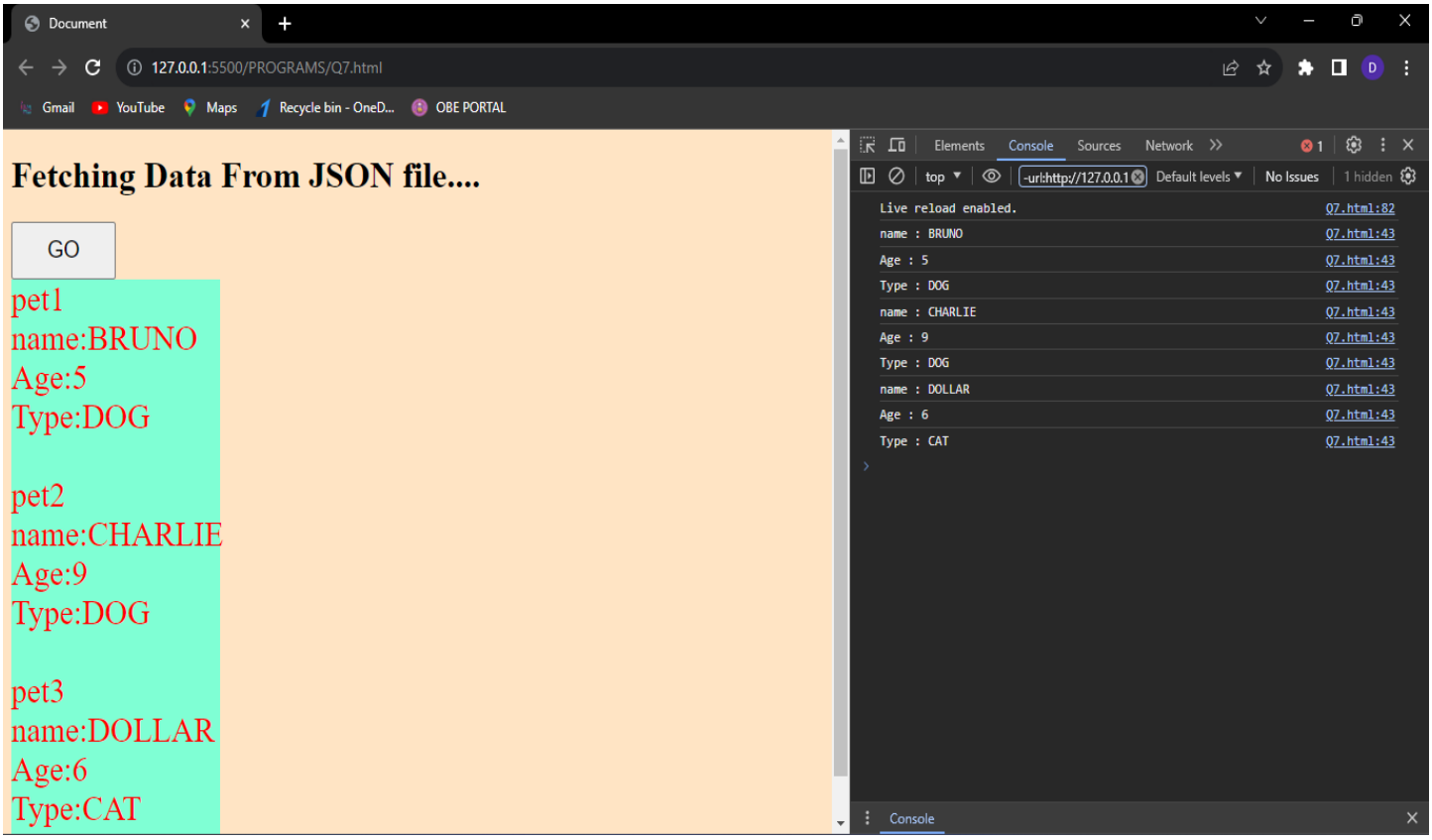
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
    <title>Document</title>
    <style>
        body{
            background-color: bisque;
        }
        #d1{
            height: 100%;
            width: 200px;
            background-color: aquamarine;
            color: red;
            font-size: 30px;
        }
        button{
            width: 100px;
            height: 50px;
            font-size: 20px;
        }
    </style>
</head>
<body>
    <h1>Fetching Data From JSON file....</h1>
    <button type="button" onclick="abc()">GO</button>
    <div id="d1"></div>
    <script>
        function abc(){
            //$(document).ready
            var a=document.getElementById('d1');
            $.ajax({
                dataType:'json',
                url : 'q7_json.json',
                request : 'get',
                success: function(data){
                    $.each(data, function(key,val){
                        var temp=(key+'<br>');
                        a.innerHTML+=temp;
                        $.each(val, function(i,j){
                            console.log(i,':',j);
                            temp=(i+':'+j+'<br>')
                            a.innerHTML+=temp;
```

```
                        $("b").text(key,val);
                    })
                    a.innerHTML+='<br>';
                })
            }
        })
    }
    </script>
</body>
</html>
```

JSON FILE

```
<!-- {
    "pet1" : {
        "name" : "tuffy",
        "Age" : "3",
        "Type" : "cat"
    },

    "pet2" : {
        "name" : "rockey",
        "Age" : "5",
        "Type" : "dog"
    },

    "pet3" : {
        "name" : "doller",
        "Age" : "6",
        "Type" : "goat"
    }
} -->
```

OUTPUT


Document
127.0.0.1:5500/PROGRAMS/Q7.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL
Fetching Data From JSON file....
GO
pet1
name:BRUNO
Age:5
Type:DOG
pet2
name:CHARLIE
Age:9
Type:DOG
pet3
name:DOLLAR
Age:6
Type:CAT
Elements
Console
Sources
Network
top
-url:http://127.0.0.1
Default levels
No Issues
1 hidden
Live reload enabled. Q7.html:82
name : BRUNO Q7.html:43
Age : 5 Q7.html:43
Type : DOG Q7.html:43
name : CHARLIE Q7.html:43
Age : 9 Q7.html:43
Type : DOG Q7.html:43
name : DOLLAR Q7.html:43
Age : 6 Q7.html:43
Type : CAT Q7.html:43
Console

PRACTICAL 8

Create a plain HTML page for B.Sc. Hons CS course, mentioning details like fee, eligibility criteria, papers with names and credits, and future possibilities after the course. A button for styling should be there at bottom of the page. On clicking on this button JavaScript should redesign the complete page using jQuery in a nice presentable way.

CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>QUESTION 8</title>
    <script src="https://code.jquery.com/jquery-3.7.1.slim.js" integrity="sha256-
UgvvN8vBkgO0luPSUl2s8TIlOSYRoGFAX4jlCIm9Adc=" crossorigin="anonymous"></script>
    <script>
</script>
</head>
<style>
    table,th,td{border: 1px solid black;}
    body{background-color:bisque;
    }
    #b1{
        background-color:lavender;
        padding: 20px;
        border-radius: 20px;
    }
    #b2{
        background-color: aquamarine;
        padding: 20px;
        border-radius: 20px;
    }
    #b3{
        background-color: gray;
        padding: 20px;
        border-radius: 20px;
    }
    #b4{
        background-color:indianred;
        padding: 20px;
        border-radius: 20px;
    }
</style>
<body>
    <h1 id="h1">BSc. Honours Computer Science</h1>
    <nav id="n0">
        <div id="d0">
            <button id="b1">Style 1</button>
            <button id="b2">Style 2</button>
            <button id="b3">Style 3</button>
            <button id="b4">Style 4</button>
```

```
        </div>
    </nav>
    <br>
    <nav id="n1">
        <a href="#d2">Fees</a>
        <a href="">Eligibility</a>
        <a href="#h3">Papers</a>
        <a href="#h3">Credit</a>
        <a href="#h4">Future</a>
    </nav>
    <br>
    <br>

    <h3 id="h2">What is computer science</h3>

    <div id="d1">Computing is part of everything we do. Computing drives innovation in
engineering,<br>
        business, entertainment, education, and the sciences—and it provides solutions to
        complex, challenging problems of all kinds.
        Computer science is the study of computers and computational systems.
        It is a broad field which includes everything from the algorithms that make up
software
        to how software interacts with hardware to how well software is developed and
designed.
        Computer scientists use various mathematical algorithms, coding procedures, and their
        expert programming skills to study computer processes and develop new software and
        systems.</div>

        <div id="d2"><span >
            <p>The major highlights of the BSc Computer Science course are tabulated
below:</p>
            <div ><table><thead><tr><th>Course Name</th><th>BSc Computer Science</th>
            </tr></thead><tbody><tr><td>Duration </td><td>3 Years (6
Semesters)</td></tr><tr>
            <td>Eligibility Criteria </td><td>Class 12th in
PCM</td></tr><tr><td>Admission Process 
            </td><td>Merit Basis or Entrance Exam</td></tr><tr><td>BSc Computer Science
Syllabus </td>
            <td>Introduction to Data structures, system programming, introduction to web
technology,
                mobile applications development, python programming, and many
more.</td></tr><tr><td>
                Average Course Fees </td><td>INR 70K </td></tr><tr><td>
                BSc Computer Science Job Profiles Software</td><td>Engineer, Software
Developer, Systems
                Architect, Web developer, Mobile App developer, Website Designing, Network
Engineer,
                Data Analyst, DTP Operator, Tech Support Professional, Network Architect,
Hardware Engineer, Programmer etc.</td></tr><tr><td>Average Starting Salary </td><td>INR
3.60 LPA - INR 22 LPA</td></tr><tr><td>BSc Computer Science Top Recruiters</td><td>HCL,
Google, Microsoft, Deloitte, Facebook, Sapient Publicis, Central Government Organizations,
IBM, Cognizant, etc.</td></tr></tbody></table></div><p><strong></p></span></div>

    <h3 id="h3">Structure & Credit distribution of B.Sc. (H) Computer Science</h3>
    <table id="t1">
```

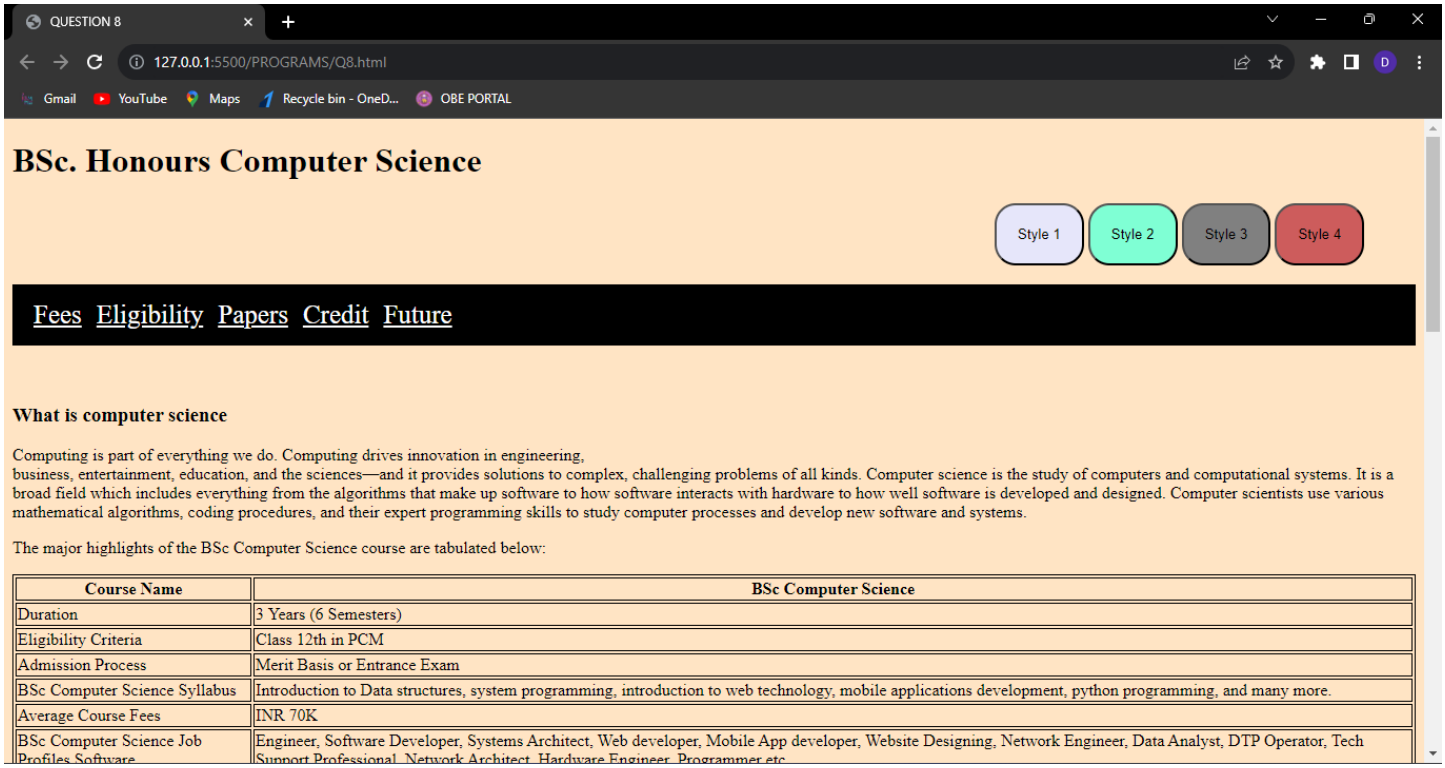
```
        <tr>
            <th>Courses</th>
            <th>Number of Courses</th>
            <th>Credits(Theory+Practical)</th>
            <th>Total Credits</th>
        </tr>
        <tr>
            <td>Core Courses(Core)</td>
            <td>14</td>
            <td>6</td>
            <td>84</td>
        </tr>
        <tr>
            <td>Generic Elective/Interdisciplinary (GE)</td>
            <td>4</td>
            <td>6</td>
            <td>24</td>
        </tr>
        <tr>
            <td>Discipline Specific Elective (DSE)</td>
            <td>4</td>
            <td>6</td>
            <td>24</td>
        </tr>
        <tr>
            <td>Skill Enhancement Courses (SEC)</td>
            <td>2</td>
            <td>4</td>
            <td>8</td>
        </tr>
        <tr>
            <td>Ability Enhancement Compulsory Course (AECC)</td>
            <td>2</td>
            <td>4</td>
            <td>8</td>
        </tr>
    </table>
    <h3 id="h4">What Careers does Computer Science Offer?</h3>
    <div id="d3">Computing jobs are among the highest paid today, and computer science
professionals
        report high job satisfaction. Most computer scientists hold at least a bachelor's
        degree in computer science or a related field.
        Principal areas of study and careers within computer science include artificial
        intelligence, computer systems and networks, security, database systems, human-
computer
        interaction, vision and graphics, numerical analysis, programming languages,
        software engineering, bioinformatics, and theory of computing.</div>
    <div id="d4">Some common job titles for computer scientists include:<br><br>

        -> Computer Programmer<br>
        -> Information Technology Specialist<br>
        -> Data Scientist<br>
        -> Web Optimization Specialist<br>
        -> Database Administrator<br>
        -> Systems Analyst<br>
```

```
        -> Web Developer<br>
        -> Quality Assurance Engineer<br>
        -> Business Intelligence Analyst<br>
        -> Systems Engineer<br>
        -> Product Manager<br>
        -> Software Engineer<br>
        -> Hardware Engineer<br>
        -> Front-End Developer<br>
        -> Back-End Developer<br>
        -> Full-Stack Developer<br>
        -> Mobile Developer<br>
        -> Network Administrator<br>
        -> Chief Information Officer<br>
        -> Security Analyst<br>
        -> Video Game Developer<br>
        -> Health Information Technician</div>


    <h3 id="h5">What Skills do Computer Scientists Need?</h3>
    <div id="d5">Learning how to program and code is only one element of the field.
        Computer scientists design, develop, and analyze the software and hardware used to
        solve problems in all kinds of business, industry, scientific, and social contexts.
        And because computers solve problems to serve and enrich people, there is a
        significant human component to computer science. Due to the range and complexity
        of the projects they take on, computer scientists depend on both technical knowledge
        and essential skills like communication, problem solving, critical thinking,
        and creativity. Other useful skills include:<br><br>

        Analytical and logical thinking<br>
        Technical and mathematical skills<br>
        Attention to detail<br>
        Project management<br>
        Technical writing<br>
        Research<br>
        Art and design<br><br>

        Not all computer science professionals will need every skill listed—because
        of the broad nature of the field, they have the opportunity to focus on the
        skills pertinent to their unique interests and chosen focus area
        (which may change over time).
    </div>
    <h3 id="h6">Undergraduate Degree Programs</h3>
    <div id="d6">Computer Science (BS)—choose a concentration in Computer Science,
Applications, Computer Systems, or Game Development<br>
        Cybersecurity (BS)—choose a concentration in Software Security or System and Network
Security<br>
        Math + Computer Science (BS)<br>
        Software Engineering (BS)</div>
</body>
<script>
     $('#d0').css({
        marginLeft:'70%'
    })

    $('#n1').css({backgroundColor:'black',
```

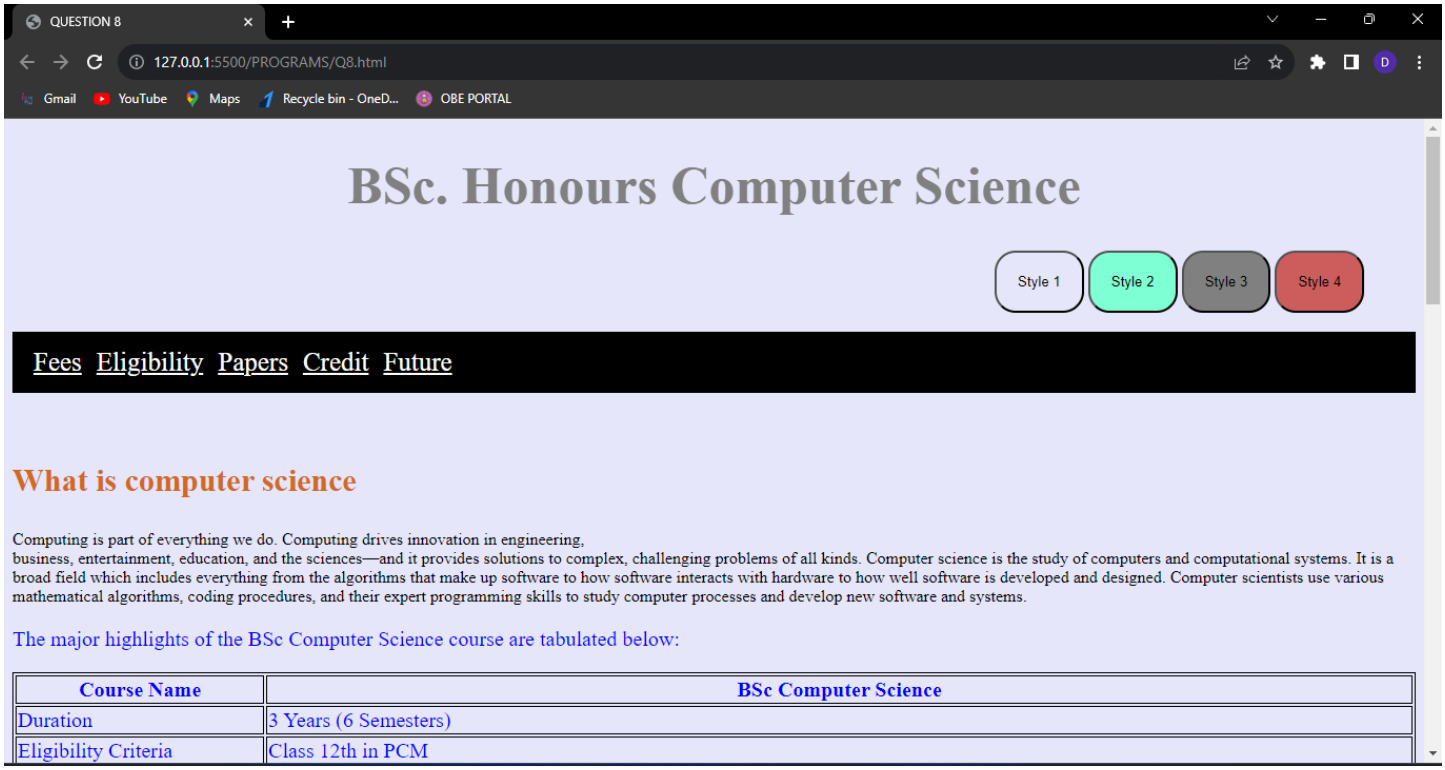
```
padding:'15px',
textAlign:'left'
})

$('a').css({
    color:'white',
    padding:'5px',
    fontSize:'25px'
})

$('#b1').click(function(){
    $('body').css({backgroundColor:'lavender'
        })

    $('h1').css({
        color:'gray',
        fontSize:'50px',
        textAlign:'center'
        });

    $('h3').css({
        color:'chocolate',
        fontSize:'30px'
        });

    $('span').css({
        color:red
    });
})
$('#b2').click(function(){
    $('body').css({backgroundColor:'aquamarine'
    })

    $('h1').css({
    color:'black',
    fontSize:'50px',
    textAlign:'left'
    });
    $('h3').css({
        color:'black'
    })

    $('span').css({color:'blue',
    fontSize:'20px'
});
})
$('#b3').click(function(){
    $('body').css({backgroundColor:'gray'
    })

    $('h1').css({
    color:'black',
    fontSize:'50px',
    textAlign:'left'
    });
```

```
        $('h3').css({
            color:'black'
        })

        $('span').css({color:'red',
        fontSize:'20px'
        });
    })

    $('#b4').click(function(){
        $('body').css({backgroundColor:'indianred'
        })

        $('h1').css({
        color:'black',
        fontSize:'50px',
        textAlign:'left'
        });
        $('h3').css({
            color:'black'
        })

        $('span').css({color:'red',
        fontSize:'20px'
        });
    })
</script>
</html>
```

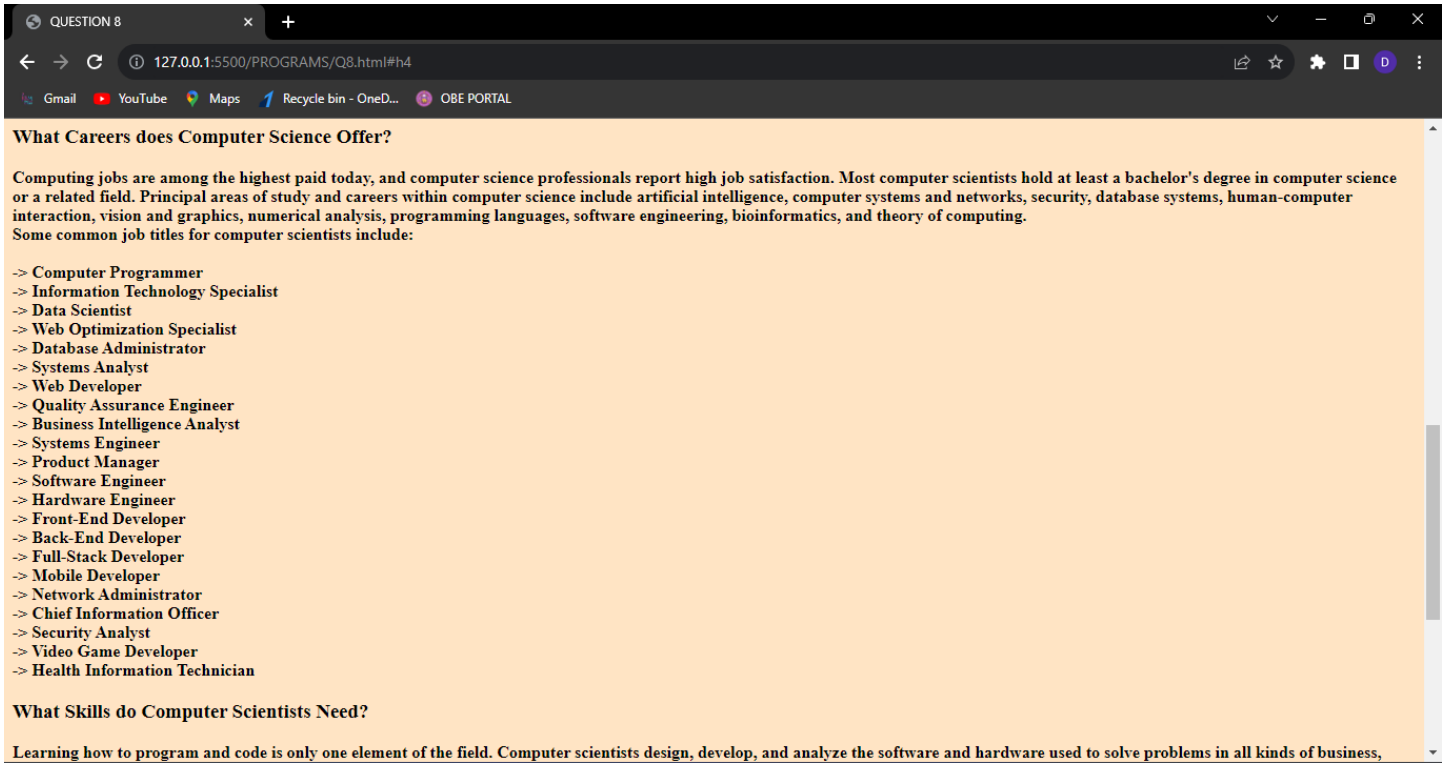
OUTPUT



ON CLICKING “STYLE 1”



ON CLICKING “FUTURE”



PRACTICAL 9

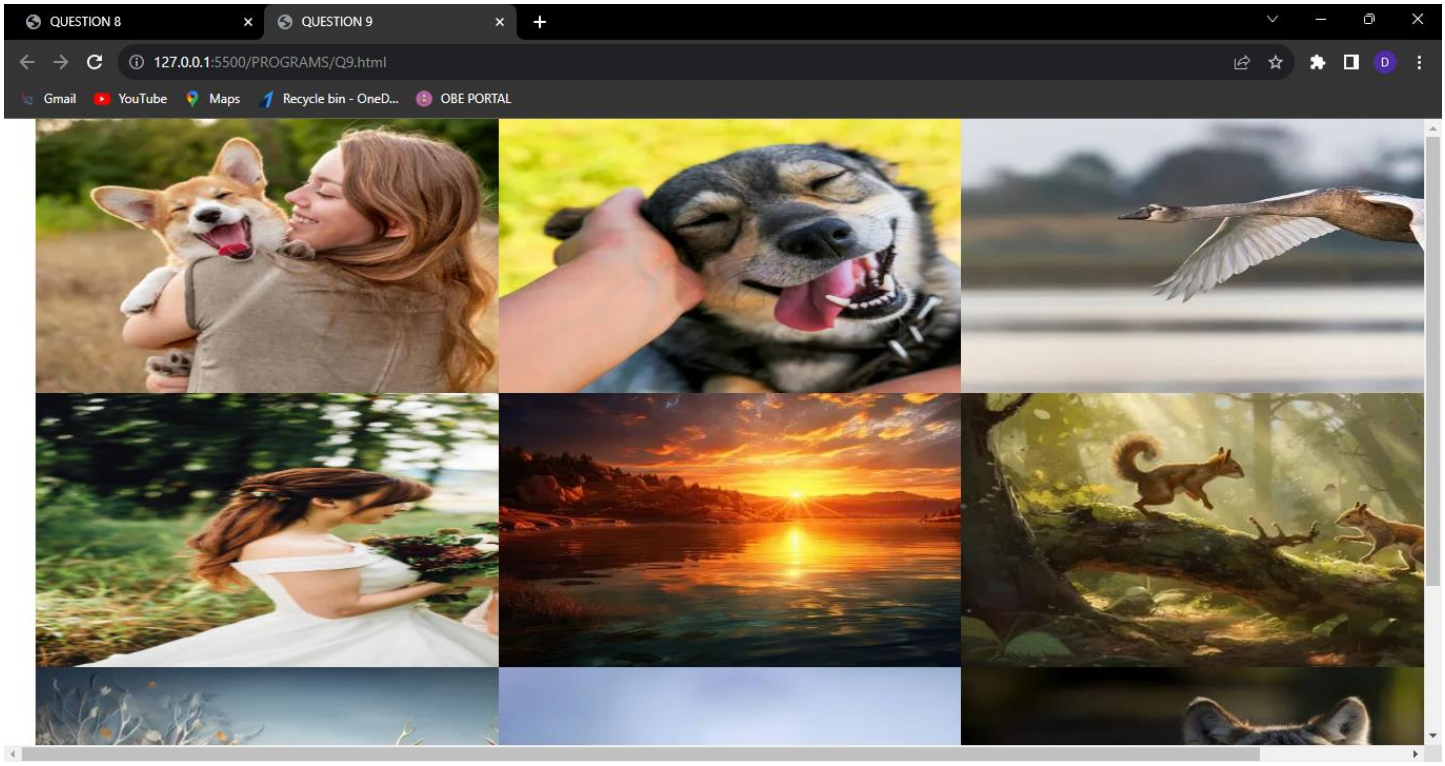
Create an HTML page for an image gallery which shows the use of BOOTSTRAP to rearrange and resize its contents on resizing the browser.

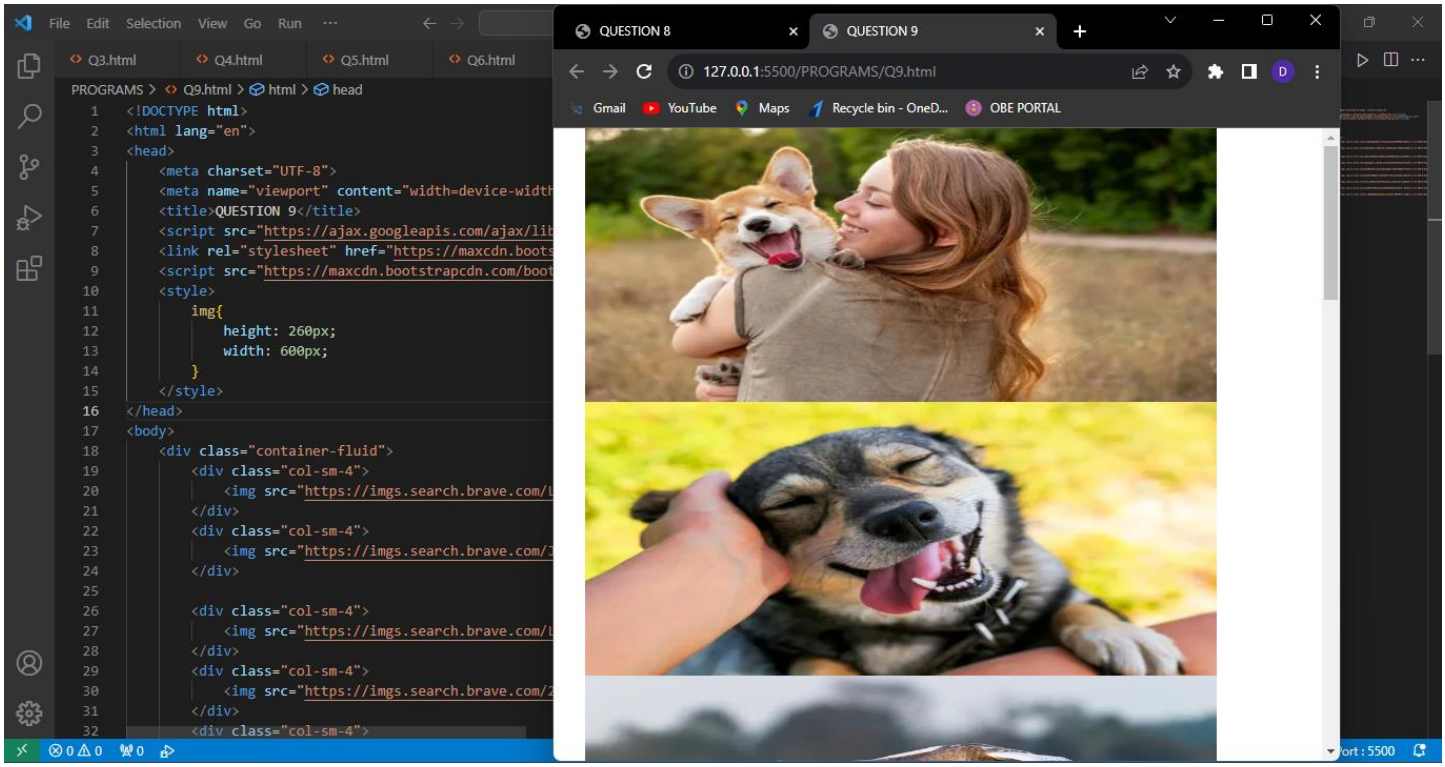
CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>QUESTION 9</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
    <style>
        img{
            height: 260px;
            width: 600px;
        }
    </style>
</head>
<body>
    <div class="container-fluid">
        <div class="col-sm-4">
            <img
src="https://imgs.search.brave.com/L71jgj36gvdZ8yrTn4lpkuyZwv1pJ9EIMMpbCveWHyc/rs:fit:500:0:0
/g:ce/aHR0cHM6Ly9tZWRp/YS5pc3RvY2twaG90/by5jb20vaWQvMTI3/Njc4ODI4My9waG90/by95b3VuZy13b21h/bi
13aXRoLWxhdWdo/aW5nLWNvcmdpLXB1/cHB5LW5hdHVyZS1i/YWNrZ3JvdW5kLmpw/Zz9zPTYxMng2MTIm/dz0wJms9Mj
AmYz1u/T2lCblZBMTNCdXBW/bjB0N281ZkN5dFY1/WlJPZ05nU1drUWFz/M0l1SEl3PQ" alt="1">
        </div>
        <div class="col-sm-4">
            <img
src="https://imgs.search.brave.com/JEdSJXGRi1iYz06znnEs_x63EUjzqVzfLRu0phiVX6g/rs:fit:500:0:0
/g:ce/aHR0cHM6Ly9tZWRp/YS5pc3RvY2twaG90/by5jb20vaWQvMTA0/MTk4NzQ4OC9waG90/by9jdXRlLWRvZy1w/dX
QtaGlzLWZhY2Ut/b24taGlzLWtuZWVz/LXRvLXRoZS1tYW4t/YW5kLXNtaWxpbmct/ZnJvbS10aGUtaGFu/ZHMtc2NyYX
RjaGlu/Zy1oZXItZWFyLmpw/Zz9zPTYxMng2MTIm/dz0wJms9MjAmYz1O/S0dmOG5tWFZka3Nt/TlMwQXk2OTZjVlBO/U
0lmQ0pKMXl1X3k5/akZHQnNNPQ" alt="2">
        </div>

        <div class="col-sm-4">
            <img
src="https://imgs.search.brave.com/L83ZiqYQ5m6avGvp0Sxv8i3m4DNNXlROimvCpGJPmDo/rs:fit:860:0:0
/g:ce/aHR0cHM6Ly9jZG4u/cGl4YWJheS5jb20v/cGhvdG8vMjAyMy8x/MC8xNS8xOC8xMi9z/d2FuLTgzMTc2MDJf/Nj
QwLmpwZw" alt="3">
        </div>
        <div class="col-sm-4">
            <img
src="https://imgs.search.brave.com/2Sqp2gCh1_vjwHlE9_2xylULEnhnI6fnDh3yrElkZV8/rs:fit:500:0:0
```

```
/g:ce/aHR0cHM6Ly9pbWFn/ZXMudW5zcGxhc2gu/Y29tL3Bob3RvLTE0/ODI0ODIwOTc3NTUt/MGI1OTU4OTNiYTYz/P2F1dG89Zm9ybWF0/JmZpdD1jcm9wJnE9/ODAmdz0xMDAwJml4/bGliPXJiLTQuMC4z/Jml4aWQ9TTN3eE1q/QTNmREI4TUh4elpX/RnlZMmg4TVRCOGZI/Qm9iM1J2ZkdWdWZE/QjhmREI4Zkh3dw.jpeg" alt="4">
        </div>
        <div class="col-sm-4">
            <img
src="https://imgs.search.brave.com/4GzMHbJ3HrwvvUc2OiDevZBWitrm4Khf2NgehYlKfeU/rs:fit:860:0:0
/g:ce/aHR0cHM6Ly90My5m/dGNkbi5uZXQvanBn/LzA2LzA1LzU0Lzc2/LzM2MF9GXzYwNTU0/NzY3Ml9uUU95RG1L/VW
5NY3dVcFZEWWh3/TTFLZ1VwdlJ4ZjBz/Ui5qcGc" alt="5">
        </div>
        <div class="col-sm-4">
            <img src="https://imgs.search.brave.com/QsG49CcH9qGUirJYeEjQpVI-FEarZucKjAEVi-
0ByjA/rs:fit:500:0:0/g:ce/aHR0cHM6Ly90My5m/dGNkbi5uZXQvanBn/LzA2LzA3Lzc4LzQy/LzM2MF9GXzYwNzc4
/NDIwMl9UWTJsYkxw/UWNvaEVuTWVSNlJh/NTZoU0lLZndjQ1ky/SS5qcGc" alt="6">
        </div>
        <div class="col-sm-4">
            <img
src="https://imgs.search.brave.com/FnTmLLhQw6soThwJTk5tyXnpLcmcSpzcBnfLJd6g12A/rs:fit:500:0:0
/g:ce/aHR0cHM6Ly90My5m/dGNkbi5uZXQvanBn/LzA2LzMwLzczLzkw/LzM2MF9GXzYzMDcz/OTA1NV9XbTRpQm9Z/QW
5VSnNNbW44YXJj/eWFXTWRRaWdTTzEx/Wi5qcGc" alt="7">
        </div>
        <div class="col-sm-4">
            <img src="https://imgs.search.brave.com/yF1mWPe73G3Ipv0cm6cGE8W9adbN3yuPZVoX--
KTk5k/rs:fit:500:0:0/g:ce/aHR0cHM6Ly9tZWRp/YS5nZXR0eWltYWdl/cy5jb20vaWQvMTIw/MTI1MjE0OC9waG90
/by9idXR0ZXJmbGll/cy5qcGc_cz02MTJ4/NjEyJnc9MCZrPTIw/JmM9cDhnYzJaVERr/SkNSc0k5cjJ5SEdE/MlB5NEx
Tdjd2em5w/SnMwTi1IM3FSST0" alt="8">
        </div>
        <div class="col-sm-4">
            <img src="https://imgs.search.brave.com/de_vGbMeNPdQ2Z8jkDCcXs4LWjyY-bMQYNPzn-
log6k/rs:fit:860:0:0/g:ce/aHR0cHM6Ly9tZWRp/YS5pc3RvY2twaG90/by5jb20vaWQvOTY2/MTE4MzkyL3Bob3Rv
/L2xvdXAtZ3Jpcy1n/cmV5LXdvbGYuanBn/P3M9NjEyeDYxMiZ3/PTAmaz0yMCZjPW5U/c09BRkhnVXB4dUpO/ZjlkRHZ
abDcwLWFP/MUQxMDZJV1JJNEtn/YXItODQ9" alt="9">
        </div>
</div>
</body>
</html>
```

OUTPUT


QUESTION 8
QUESTION 9
127.0.0.1:5500/PROGRAMS/Q9.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL


File Edit Selection View Go Run
Q3.html
Q4.html
Q5.html
Q6.html
PROGRAMS > Q9.html > html > head
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width
<title>QUESTION 9</title>
<script src="https://ajax.googleapis.com/ajax/lib
<link rel="stylesheet" href="https://maxcdn.boots
<script src="https://maxcdn.bootstrapcdn.com/boot
<style>
img{
height: 260px;
width: 600px;
}
</style>
</head>
<body>

<img src="https://imgs.search.brave.com/

<img src="https://imgs.search.brave.com/

<img src="https://imgs.search.brave.com/

<img src="https://imgs.search.brave.com/

QUESTION 8
QUESTION 9
127.0.0.1:5500/PROGRAMS/Q9.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL

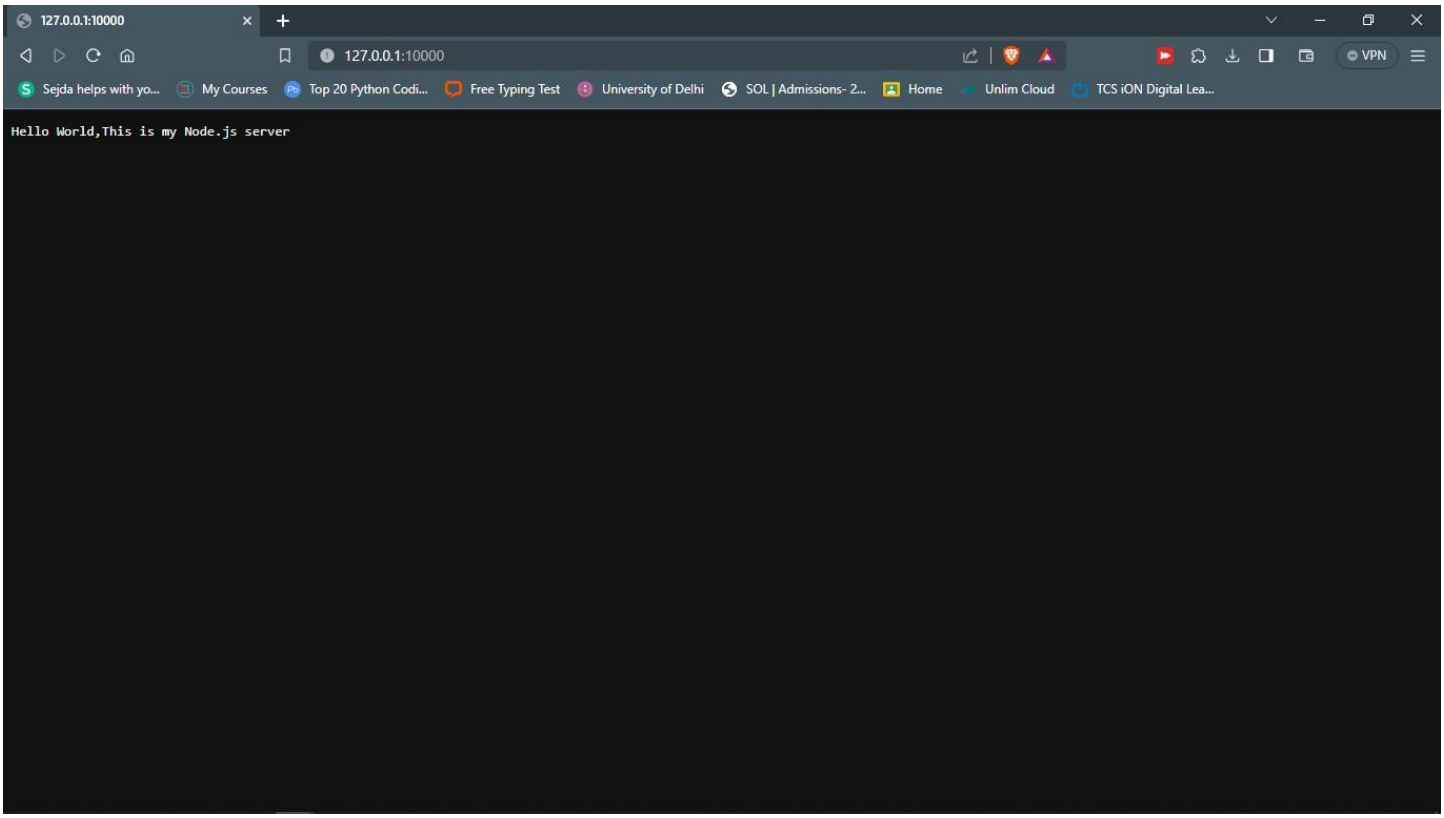
PRACTICAL 10

Create an HTTP server using Node.js which handles requests on port 10000 or a free port beyond 10000. Modify the server in such a way that opening localhost: 10000 will display “Hello world, This is my node.js server” on browser.

CODE

```
const http=require("http")
const hostname='127.0.0.1'
const port=10000
function sayHello(req,res){
    res.statusCode=200
    res.setHeader('Content-Type','text/plain')
    res.write("Hello World,This is my Node.js server")
    res.end()
}
const server =http.createServer(sayHello)
server.listen(port,hostname,()=>{
    console.log(`Server running at http://${hostname}:${port}`)
})
```

OUTPUT



PRACTICAL 11

Create index.html file containing two forms for SignIn and SignUp. Submitting SignIn form should search for credentials in mysql database using server created in previous practical. On successful signin, a welcome page should be displayed. Submitting SignUp form should insert new entry for credentials in mysql database using server created in previous practical. On successful signup, user should be returned back to index.html.

CODE

SERVER.JS

```
const http = require('http');
const fs = require("fs");
const mysql = require("mysql");

const qs = require('querystring');

//creating caonnection to dbms

const mycon = mysql.createConnection({
  host:"localhost",
  user :"root",
  password:"1211",
  database:"entry",
  insecureAuth : true
})


mycon.connect((err,data)=>{
  if(err){
console.log("unable to connect to db ", err.message)
  }else{
    console.log("db connected",)
  }
})


//creating server

const server = http.createServer((req,res)=>{
  const path = req.url

  //storeVisitorsLogs(req);

  if(path == '/welcome'){
    loadWelcome(res)
  }else if(path == '/'){
    loadIndex(res)

  }else if(req.method == 'POST' && path =='/signup' ){
    funSignUp(req,res)

  }else if(req.method == 'POST' && path =='/signin'){
```

```
    funSignIn(req,res)
  }

})


server.listen(3000,()=>{
  console.log("server started")
})


//------------------------function  to do signup ------------------

function funSignUp(req,res){

  var body = '';

  req.on('data', (chunk) => {
    body += chunk;
  });

  req.on('end', () => {

  var d = qs.parse(body);
  var query  = `INSERT INTO users VALUES ( ?, ?, ?)`;

  mycon.query(query,[d.username,d.name,d.pass],(err,data)=>{
    if(err){
      console.log("error in insertion ",err)
      res.statusCode = 409
      res.write("chose other user name")
      res.end()
    }else{
      console.log("inserted")
      res.statusCode = 201
      res.writeHead(302, { 'Location': '/' });
      res.end()
    }

  })
  })
}


//=====================function  to do sign in ======================


function funSignIn(req,res){

  var body = '';


  //working with the body of a POST request
```

```
  req.on('data', (chunk) => {
    body += chunk;
  });


  //---- trigerred when entire entire request has been received.
  //-----when all data has been received for the request.

  req.on('end', () => {
  var d = qs.parse(body);

  var query  = `SELECT * FROM user where username = ? AND password = ? `;

  mycon.query(query,[d.username , d.pass],(err,data)=>{

    if(err){
      console.log("error in server ")
      res.statusCode = 404
      res.write("error in server")
      res.end()

    }else{

      console.log(data)
      if(data.length == 1){

        res.statusCode = 200;
        res.writeHead(302, { 'Location': '/welcome'});
        res.end();

      }else{

        res.statusCode = 409 ;
        res.write("sign in failed")
        res.end();

      }
    }

  })
  })


}



//============================function  to load index.html ====================

function loadIndex(res){

  fs.readFile("index.html",async (err,data)=>{
    if(err){
      console.log("reading file failed")
      res.write("text file nii milla")
```

```
      res.end()

    }else{
      if(res.write(data)){
        res.statusCode = 200
        res.end()
      }
    }
  })
}


//============================function  to LOAD WELCOME PAGE ====================

function loadWelcome(res){

  fs.readFile("welcome.html", async (err,data)=>{
    if(err){
      res.write("error in loading welcome page")
      res.end()
    }else{
      res.write(data)
      res.end();
    }
  })

}

//================================================================================

function storeVisitorsLogs(req){

  var str = req.headers['x-forwarded-for'] || req.connection.remoteAddress;
  var d = new Date();

  const months = [
    'January', 'February', 'March', 'April',
    'May', 'June', 'July', 'August',
    'September', 'October', 'November', 'December'
  ];

  fs.appendFile("serverlogs.txt",`${d.getDay()} ${months[d.getMonth()]}
${d.getHours()}::${d.getMinutes()}::${d.getSeconds()}  ${str} : ${req.method}
${req.url}\n`,(err)=>{

    if(err){
      console.log("unable to log")
    }else{
      console.log("logged the visitor to log txt file")
    }

  })

}
```

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1> Sign in...</h1>
    <form action ="http://127.0.0.1:3000/signin" id="form1" method="post">
        username<input type="text" name="username" id=""><br><br>
        password<input type = "password" name="pass" id=""><br><br>
        <input type="submit" name="submit " id ="" value="sign-in"><br><br>

    </form>
    <h1> Sign up...</h1>
    <form action ="http://127.0.0.1:3000/signup" id="form2" method="post" >
        name<input type="text " name="name"><br><br>
        username<input type="text" name="username" id=""><br><br>
        password<input type = "password" name="pass" id=""><br><br>
        <input type="submit" name="submit " id ="" value="sign-in"><br><br>

    </form>
</body>
</html>
```

OUTPUT

Sign in...

username

password

sign-in

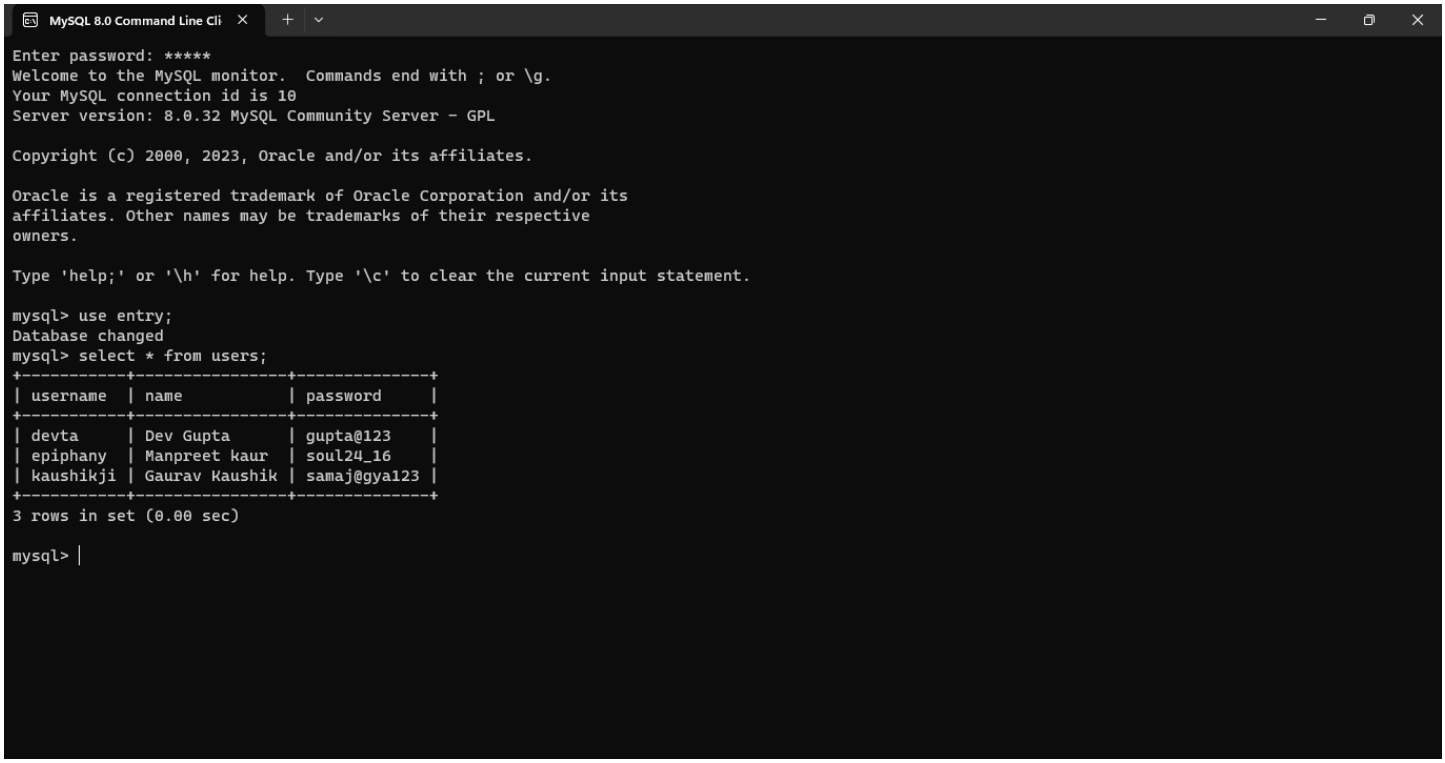
Sign up...

name

username

password

sign-in



EXTRA QUESTIONS

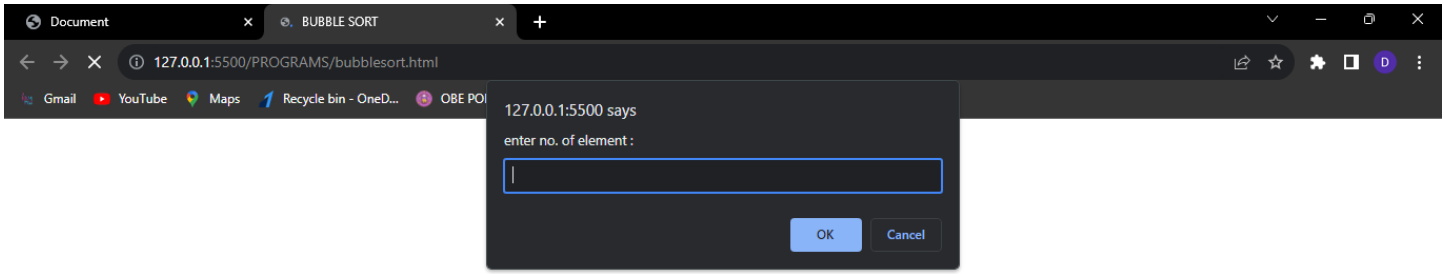
Q1.

Write a program of bubble sort.

CODE

```
<html>
    <head>
        <title>
            BUBBLE SORT
        </title>
    </head>
    <body>
        <script>
            var n=prompt("enter no. of element :")
            const arr=new Array();
            for(var i=1;i<=n;i++){
                var a=prompt("enter element "+i+" : ");
                arr.push(a);
            }
            document.write("array before sorting :"+arr+"<br>");
            for(var i=0;i<n-1;i++){
                for(var j=i;j<n;j++){
                    if(arr[i]>arr[j]){
                        var temp=arr[i];
                        arr[i]=arr[j];
                        arr[j]=temp;
                    }
                }
            }
            document.write("array after sorting :"+arr+"<br>");
        </script>
    </body>
</html>
```

OUTPUT



array before sorting :9,4,7,2,5
array after sorting :2,4,5,7,9

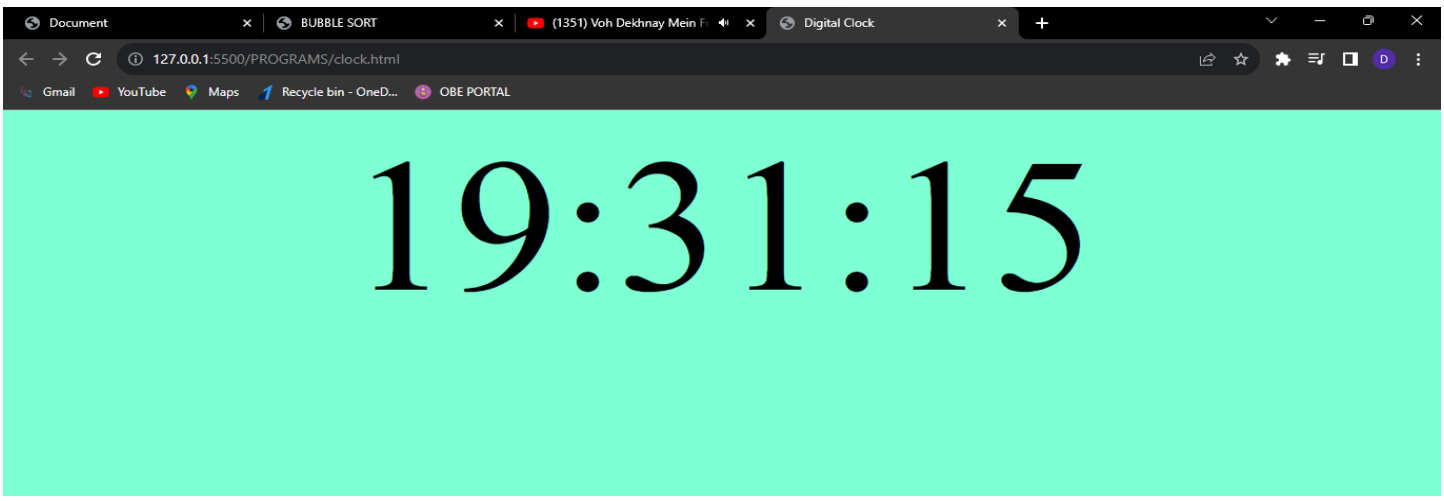
Q2.

Write a Program which shows a digital clock.

CODE

```
<html>
<head>
    <title>Digital Clock</title>
    <style>
        body{
            background-color: aquamarine;
        }
        #clock {
            font-size: 200px;
            text-align: center;
        }
    </style>
</head>
<body>
    <div id="clock"></div>

    <script>
        function updateClock() {
            var now = new Date();
            var hr = now.getHours();
            var min = now.getMinutes();
            var sec = now.getSeconds();
            var timeString = `${hr}:${min}:${sec}`;
            document.getElementById('clock').textContent = timeString;
        }
        function abc(){
            setInterval(updateClock, 1000);
        }
        abc();
    </script>
</body>
</html>
```

OUTPUT



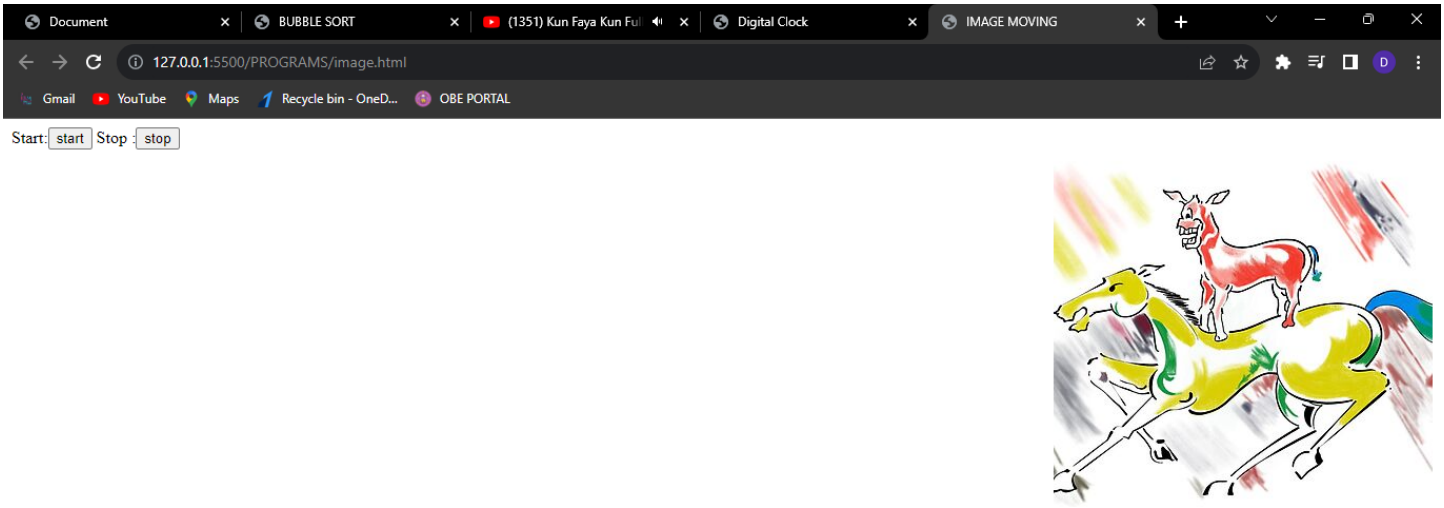
Q3.

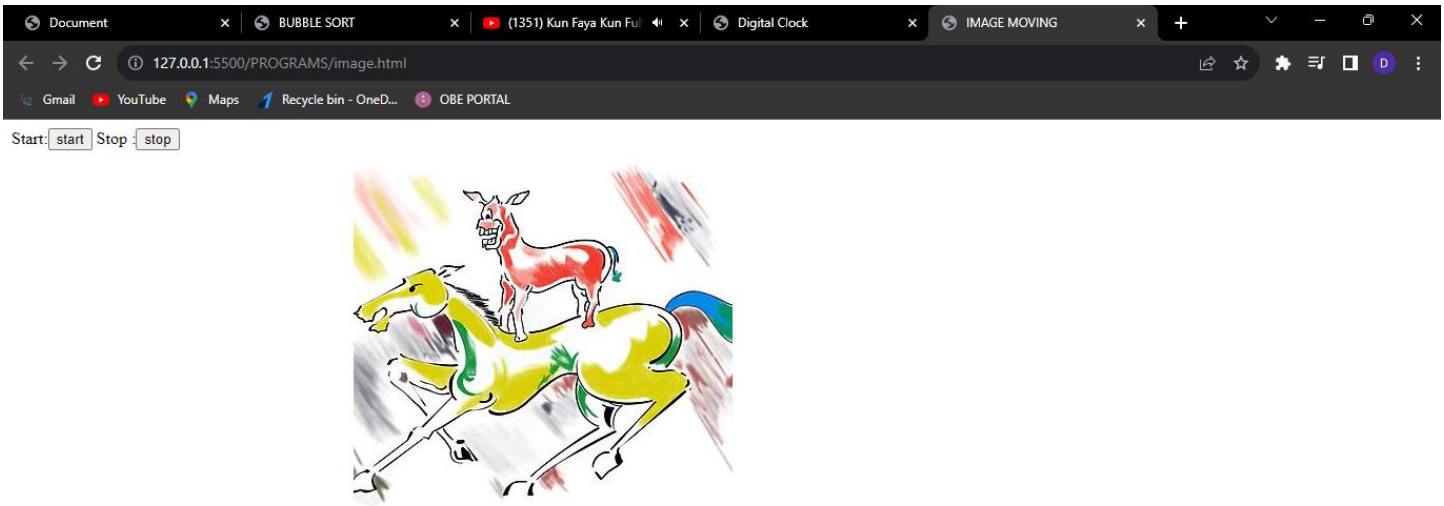
Write a program which shows a image running from right to left when you press start button and stop the image when you press stop button.

CODE

```
<html>
    <head>
        <title>
            IMAGE MOVING
        </title>
    </head>
    <style>
        img{
            float: right;
        }
    </style>
    <body>
        <img src="majnu.jpg" id="1" name="majnu">
        <form>
            Start:<input type="button" value="start" id="101" onclick="abc()"/>
            Stop :<input type="button" value="stop" id="102" onclick="xyz()"/>
        </form>
    </body>

    <script>
        var time;
        var pd=0;
        function b_b_b_saawan_ko_paani(){
            document.getElementById("1").style.paddingRight=pd+"5px";
            pd=pd+1;
        }
        function abc(){
            time=setInterval(b_b_b_saawan_ko_paani,20);
        }
        function xyz(){
            clearInterval(time);
        }
    </script>
</html>
```

OUTPUT


Document
BUBBLE SORT
(1351) Kun Faya Kun Ful
Digital Clock
IMAGE MOVING
127.0.0.1:5500/PROGRAMS/image.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL
Start: start Stop : stop


Document
BUBBLE SORT
(1351) Kun Faya Kun Ful
Digital Clock
IMAGE MOVING
127.0.0.1:5500/PROGRAMS/image.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL
Start: start Stop : stop

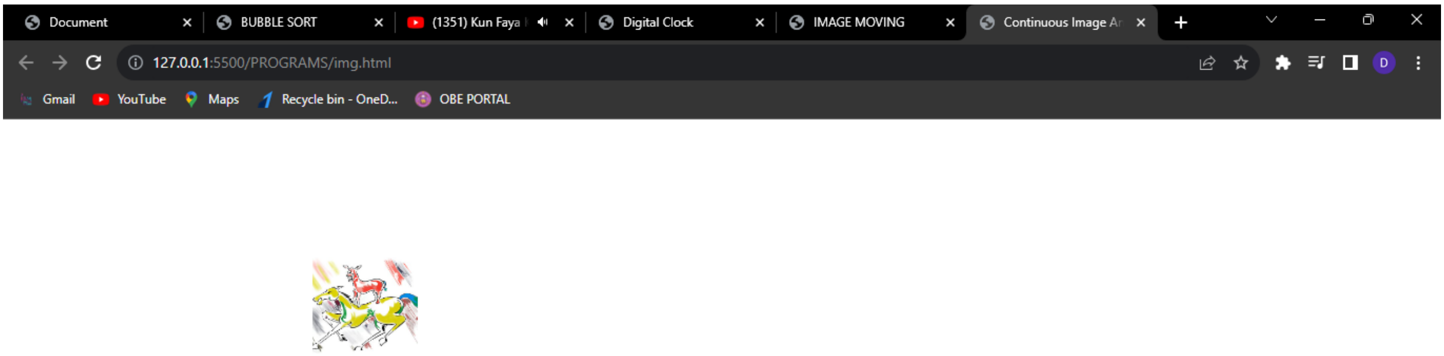
Q4.

Write a program in which a image shows movement in every 2 seconds.

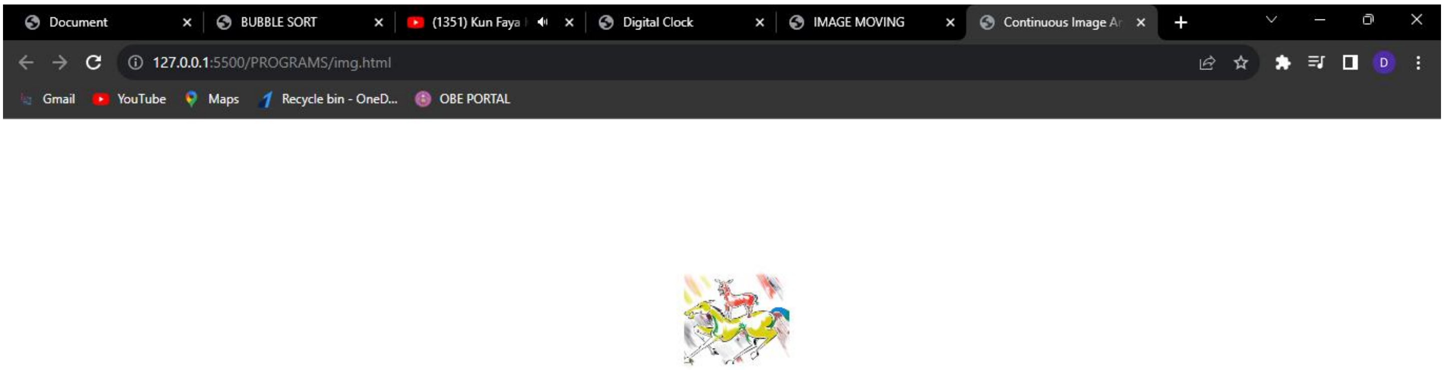
CODE

```
<!DOCTYPE html>
<html>
<head>
    <title>Continuous Image Animation</title>
    <style>
        /* Define the CSS for the image */
        #movingImage {
            position: absolute;
            width: 100px;
            height: 100px;
            transition: 1s; /* Smooth transition for movement */
        }
    </style>
</head>
<body>
    <img id="movingImage" src="majnu.jpg">
    <script>
        function moveImage() {
            var windowWidth = window.innerWidth;
            var windowHeight = window.innerHeight;

            // Calculate random X and Y coordinates within the window
            var randomX = Math.random() * (windowWidth - 100); // 100 is the image width
            var randomY = Math.random() * (windowHeight - 100); // 100 is the image height

            // Set the new coordinates for the image
            document.getElementById("movingImage").style.left = randomX + "px";
            document.getElementById("movingImage").style.top = randomY + "px";

            setTimeout(moveImage, 2000); // Change image location every 2 seconds
        }

        // Start the image movement when the page loads
        window.onload = moveImage;
    </script>
</body>
</html>
```

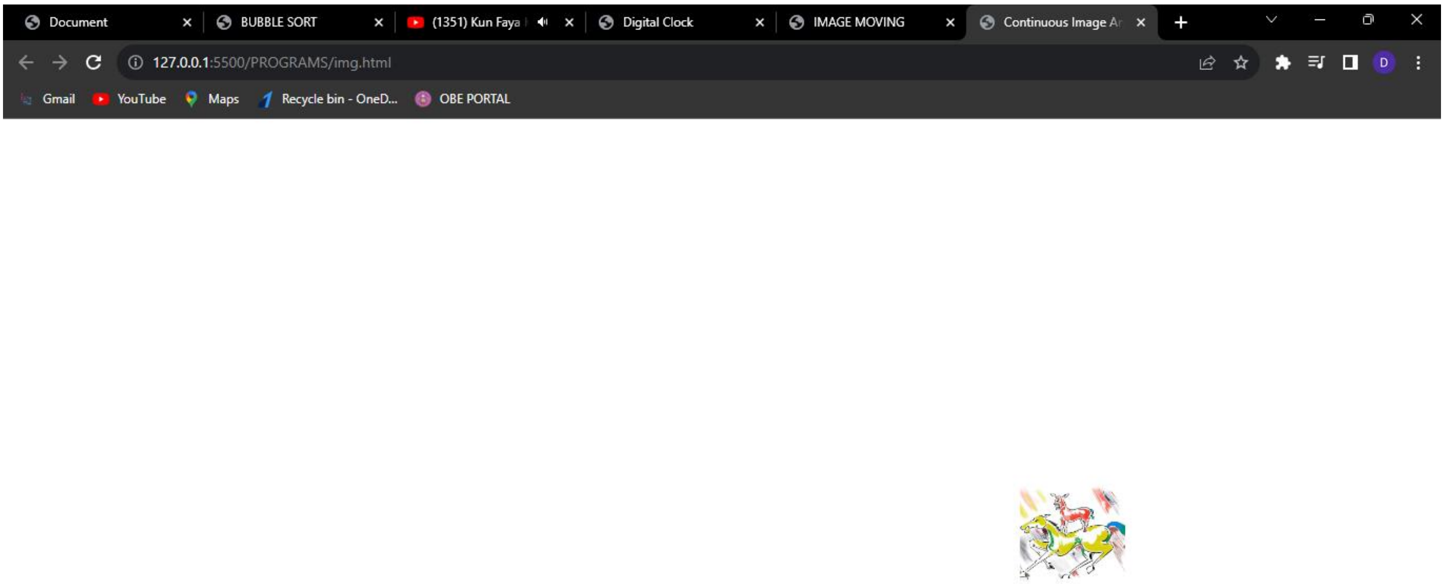
OUTPUT



Document
BUBBLE SORT
(1351) Kun Faya
Digital Clock
IMAGE MOVING
Continuous Image A
127.0.0.1:5500/PROGRAMS/img.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL



Document
BUBBLE SORT
(1351) Kun Faya
Digital Clock
IMAGE MOVING
Continuous Image A
127.0.0.1:5500/PROGRAMS/img.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL



Document
BUBBLE SORT
(1351) Kun Faya
Digital Clock
IMAGE MOVING
Continuous Image A
127.0.0.1:5500/PROGRAMS/img.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL

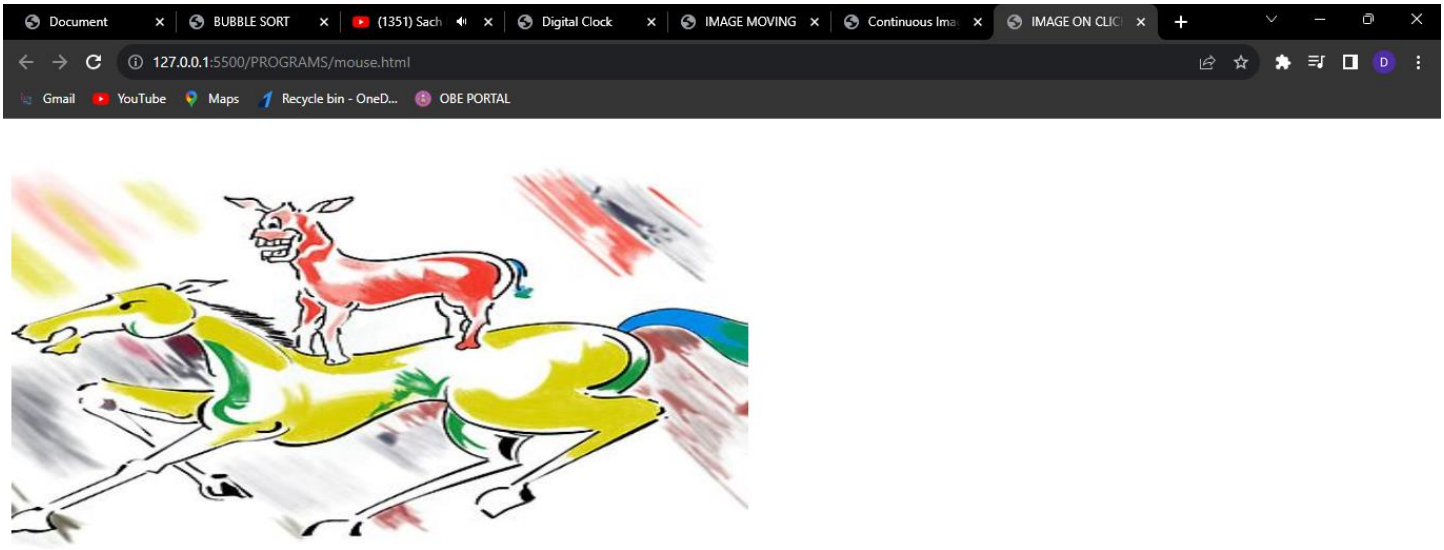
Q5.

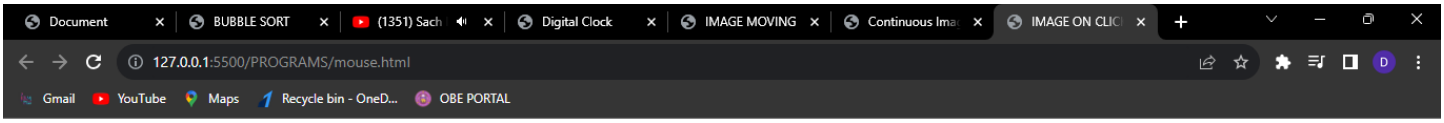
Write a program where a image gets expand when you enter mouse in image and when you exit your mouse, image comes to original size.

CODE

```
<html>
    <head>
        <title>
            IMAGE ON CLICK
        </title>
    </head>
    <body>
        <img src="majnu.jpg" width="400px" height="400px" id="1" name="img"
onmouseenter="abc()" onmouseout="xyz()">
    </body>
    <script>
        function abc(){
            document.getElementById("1").style.width="700px".height="700px";
        }
        function xyz(){
            document.getElementById("1").style.width="400px".height="400px";
        }
    </script>
</html>
```

OUTPUT


Document
BUBBLE SORT
(1351) Sach
Digital Clock
IMAGE MOVING
Continuous Ima
IMAGE ON CLIC
127.0.0.1:5500/PROGRAMS/mouse.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL


Document
BUBBLE SORT
(1351) Sach
Digital Clock
IMAGE MOVING
Continuous Ima
IMAGE ON CLIC
127.0.0.1:5500/PROGRAMS/mouse.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL

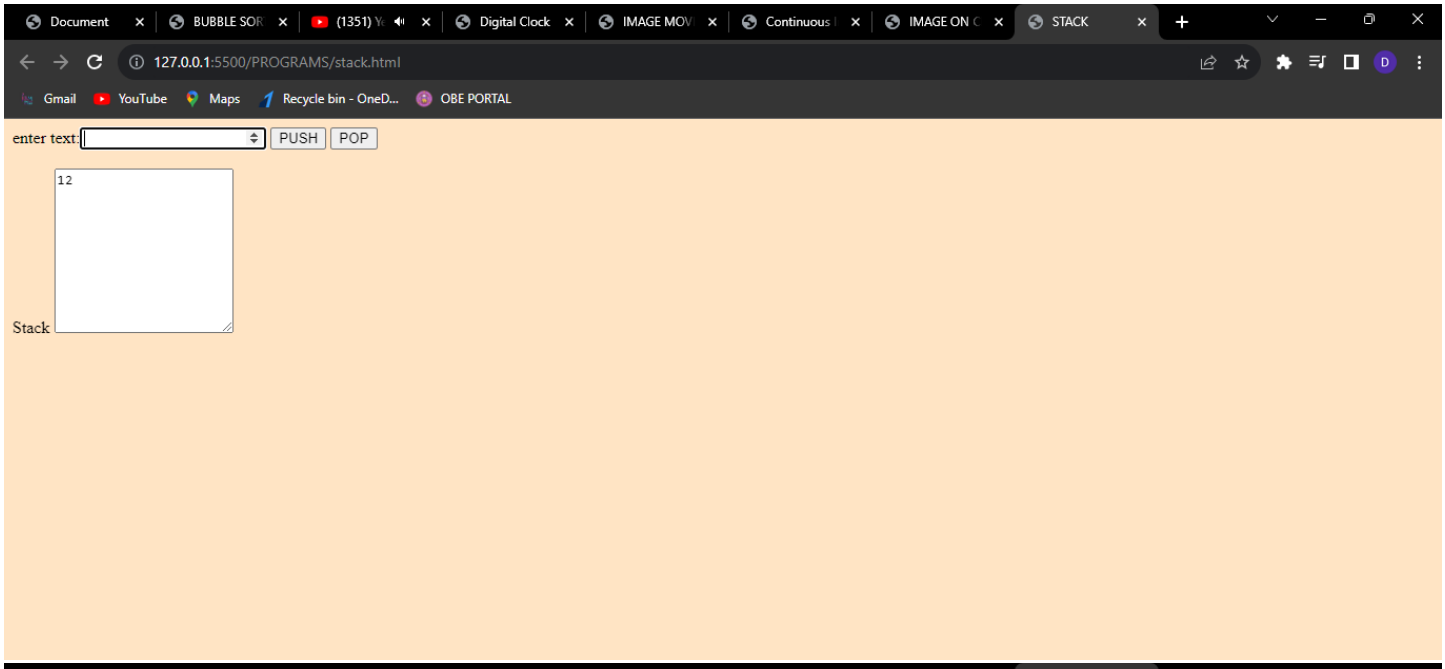
Q6.

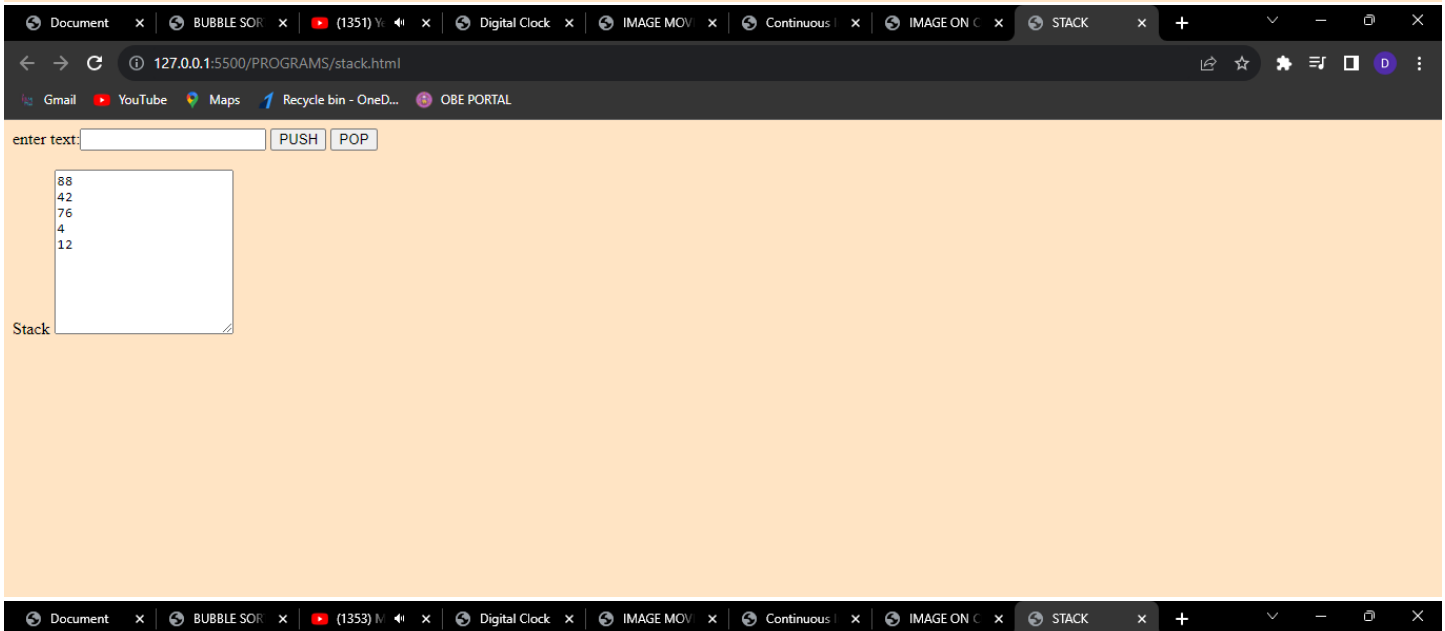
Make a stack program in which push and pop buttons are there when you click on push then push the element in stack and when pop is pressed then remove the upper element from stack.

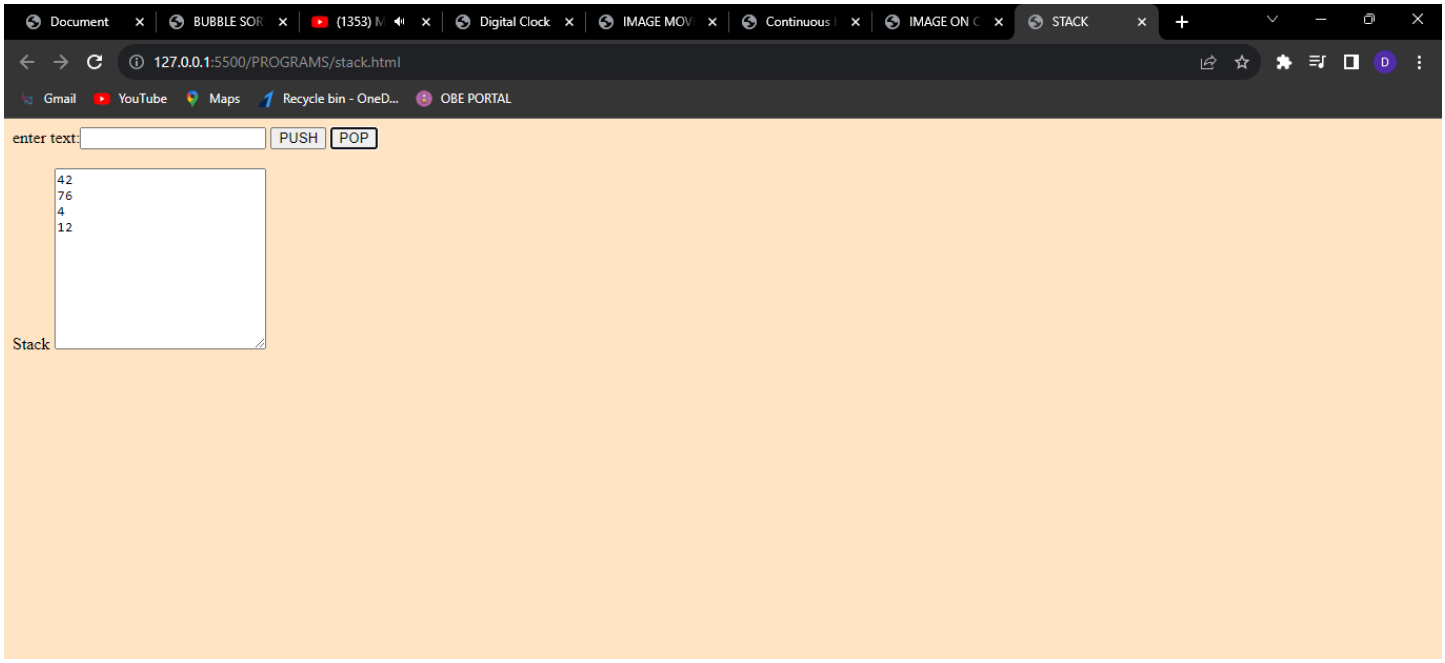
CODE

```
<html>
    <head>
        <title>  STACK   </title>
    </head>
    <style>
        body{background-color:bisque}
    </style>
    <body>
            enter text:<input type="number" id="0"/>
            <button id="b1" onclick="f1()">PUSH</button>
            <button id="b2" onclick="f2()">POP</button><br><br>

            <label id="1">Stack</label>
            <textarea rows="10" id="2"></textarea><br>
        <script>
            const arr=new Array();
            function f1(){
                var v1=document.getElementById("0").value;
                if(v1==""){
                    alert("value is empty");
                }else{
                arr.push(v1);
                document.getElementById("0").value="";
                var s="";
                for(var i=arr.length-1;i>=0;i--){
                    s=s+arr[i]+"\n";
                }
                document.getElementById("2").value=s;}
            }
            function f2(){
                if(arr.length==0){
                    alert("stack is empty");
                }else{
                arr.pop();
                var s="";
                for(var i=arr.length-1;i>=0;i--){
                    s=s+arr[i]+"\n";
                }
                document.getElementById("2").value=s;
                }
            }
         </script>
    </body>
</html>
```

OUTPUT


Document
BUBBLE SOR
Digital Clock
IMAGE MOV
Continuous
IMAGE ON
STACK
127.0.0.1:5500/PROGRAMS/stack.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL
enter text:
PUSH
POP
12
Stack


Document
BUBBLE SOR
Digital Clock
IMAGE MOV
Continuous
IMAGE ON
STACK
127.0.0.1:5500/PROGRAMS/stack.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL
enter text:
PUSH
POP
88
42
76
4
12
Stack


Document
BUBBLE SOR
Digital Clock
IMAGE MOV
Continuous
IMAGE ON
STACK
127.0.0.1:5500/PROGRAMS/stack.html
Gmail
YouTube
Maps
Recycle bin - OneD...
OBE PORTAL
enter text:
PUSH
POP
42
76
4
12
Stack

THANK

YOU!