Internet Technology Practical File

Name – SONU

Examination Roll No. – 20009570049

Roll No. – 2002058

Semester – Vth

Year – 2022-23

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.

```
C:\Windows\system32\cmd.exe

C:\Users\smali>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 1:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 2:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::6cfc:4183:6c3:a903%7
   IPv4 Address. . . . . . . . . . . : 192.168.1.14
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1

C:\Users\smali>
```

As the IPv4 address starts with **192**, that means it is a Class C address. Therefore, the network address would be **192.168.1.0**. The maximum number of possible systems to use on the network would be **254**. The range of Ip addresses would be **192.168.1.1 - 192.168.1.254**.

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.

```
C:\Windows\system32\cmd.exe                                                    —  □  ✕

C:\Users\smali>ping google.com

Pinging google.com [142.250.206.110] with 32 bytes of data:
Reply from 142.250.206.110: bytes=32 time=4ms TTL=119
Reply from 142.250.206.110: bytes=32 time=5ms TTL=119
Reply from 142.250.206.110: bytes=32 time=4ms TTL=119
Reply from 142.250.206.110: bytes=32 time=5ms TTL=119

Ping statistics for 142.250.206.110:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 5ms, Average = 4ms

C:\Users\smali>tracert google.com

Tracing route to google.com [142.250.206.110]
over a maximum of 30 hops:

  1     4 ms     2 ms     2 ms  192.168.1.1 [192.168.1.1]
  2     3 ms     4 ms     3 ms  10.199.199.199 [10.199.199.199]
  3     4 ms     4 ms     5 ms  103.196.220.109
  4     4 ms     4 ms     5 ms  10.100.8.2 [10.100.8.2]
  5     4 ms     4 ms     4 ms  72.14.205.145
  6     4 ms     4 ms     5 ms  74.125.244.193
  7     5 ms     5 ms     4 ms  142.251.76.193
  8     5 ms     4 ms     5 ms  del11s20-in-f14.1e100.net [142.250.206.110]

Trace complete.

C:\Users\smali>
```

```
C:\Windows\system32\cmd.exe                                                    —  ⬜  ✕

C:\Users\smali>netstat -ano -p tcp

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       560
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING       5400
  TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING       2640
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING       768
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING       624
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING       1520
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING       1292
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING       3164
  TCP    0.0.0.0:49670          0.0.0.0:0              LISTENING       760
  TCP    127.0.0.1:8884         0.0.0.0:0              LISTENING       4
  TCP    192.168.1.14:139       0.0.0.0:0              LISTENING       4
  TCP    192.168.1.14:50068     142.250.192.170:443   ESTABLISHED     8620
  TCP    192.168.1.14:50069     142.250.192.170:443   ESTABLISHED     8620
  TCP    192.168.1.14:50071     20.198.118.190:443    ESTABLISHED     7276
  TCP    192.168.1.14:50081     20.198.118.190:443    ESTABLISHED     3720
  TCP    192.168.1.14:50496     52.114.44.75:443      ESTABLISHED     12376
  TCP    192.168.1.14:50516     52.114.15.123:443     ESTABLISHED     12864
  TCP    192.168.1.14:50763     52.108.44.14:443      ESTABLISHED     12628
  TCP    192.168.1.14:60094     52.111.252.7:443      ESTABLISHED     12628
  TCP    192.168.1.14:60130     13.76.153.29:443      ESTABLISHED     15208
  TCP    192.168.1.14:60276     52.111.252.7:443      ESTABLISHED     12628
  TCP    192.168.1.14:60277     13.107.42.12:443      ESTABLISHED     12628
  TCP    192.168.1.14:60281     52.108.44.14:443      ESTABLISHED     12628
  TCP    192.168.1.14:60298     142.251.10.188:5228   ESTABLISHED     12388
  TCP    192.168.1.14:60307     40.100.141.162:443    ESTABLISHED     11528
  TCP    192.168.1.14:60495     20.185.212.106:443    ESTABLISHED     252
  TCP    192.168.1.14:60651     13.107.42.12:443      ESTABLISHED     12628
  TCP    192.168.1.14:60656     13.107.42.12:443      ESTABLISHED     12628
  TCP    192.168.1.14:60882     20.43.44.165:8883     ESTABLISHED     11764
  TCP    192.168.1.14:61067     142.250.193.234:443   CLOSE_WAIT      8620
  TCP    192.168.1.14:64434     23.63.109.106:443     CLOSE_WAIT      11340
  TCP    192.168.1.14:64596     172.217.166.202:443   CLOSE_WAIT      8620
  TCP    192.168.1.14:64597     142.250.193.234:443   CLOSE_WAIT      8620
  TCP    192.168.1.14:64610     13.89.178.27:443      TIME_WAIT       0
  TCP    192.168.1.14:64634     52.109.56.78:443      TIME_WAIT       0
  TCP    192.168.1.14:64635     40.99.9.162:443       TIME_WAIT       0
  TCP    192.168.1.14:64636     40.99.9.162:443       TIME_WAIT       0
  TCP    192.168.1.14:64643     104.16.53.111:443     TIME_WAIT       0
  TCP    192.168.1.14:64644     216.58.196.110:443    TIME_WAIT       0
```

```
C:\Windows\system32\cmd.exe                                                                                    —   □   ×
  TCP    192.168.1.14:64645     104.18.70.113:443      TIME_WAIT      0
  TCP    192.168.1.14:64647     104.18.70.113:443      TIME_WAIT      0
  TCP    192.168.1.14:64648     156.146.56.162:443     TIME_WAIT      0
  TCP    192.168.1.14:64649     104.16.51.111:443      TIME_WAIT      0
  TCP    192.168.1.14:64650     104.16.53.111:443      TIME_WAIT      0
  TCP    192.168.1.14:64651     104.18.70.113:443      TIME_WAIT      0
  TCP    192.168.1.14:64652     104.16.51.111:443      TIME_WAIT      0
  TCP    192.168.1.14:64653     52.114.16.177:443      TIME_WAIT      0
  TCP    192.168.1.14:64654     162.159.129.81:443     TIME_WAIT      0
  TCP    192.168.1.14:64655     162.159.136.84:443     TIME_WAIT      0
  TCP    192.168.1.14:64656     162.159.136.84:443     TIME_WAIT      0
  TCP    192.168.1.14:64657     172.64.152.80:443      TIME_WAIT      0
  TCP    192.168.1.14:64658     162.159.136.84:443     TIME_WAIT      0
  TCP    192.168.1.14:64659     104.18.232.170:443     TIME_WAIT      0
  TCP    192.168.1.14:64660     104.18.231.170:443     TIME_WAIT      0
  TCP    192.168.1.14:64666     162.247.241.2:443      TIME_WAIT      0
  TCP    192.168.1.14:64668     54.254.66.154:443      TIME_WAIT      0
  TCP    192.168.1.14:64669     108.158.245.65:443     TIME_WAIT      0
  TCP    192.168.1.14:64671     34.107.204.85:443      TIME_WAIT      0
  TCP    192.168.1.14:64673     20.189.173.11:443      ESTABLISHED    12628
  TCP    192.168.1.14:64674     142.250.194.208:443    TIME_WAIT      0
  TCP    192.168.1.14:64675     162.247.241.2:443      TIME_WAIT      0
  TCP    192.168.1.14:64676     172.64.152.80:443      TIME_WAIT      0
  TCP    192.168.1.14:64677     52.109.56.83:443       TIME_WAIT      0
  TCP    192.168.1.14:64678     52.109.56.83:443       TIME_WAIT      0
  TCP    192.168.1.14:64679     52.109.56.83:443       TIME_WAIT      0
  TCP    192.168.1.14:64680     52.109.56.83:443       TIME_WAIT      0
  TCP    192.168.1.14:64681     20.185.212.106:443     TIME_WAIT      0
  TCP    192.168.1.14:64682     162.247.241.2:443      TIME_WAIT      0
  TCP    192.168.1.14:64683     20.44.229.112:443      ESTABLISHED    2620
  TCP    192.168.1.14:64684     162.247.241.2:443      TIME_WAIT      0
  TCP    192.168.1.14:64685     162.247.241.2:443      TIME_WAIT      0
  TCP    192.168.1.14:64693     192.0.73.2:443         TIME_WAIT      0
  TCP    192.168.1.14:64697     142.250.77.200:443     TIME_WAIT      0
  TCP    192.168.1.14:64699     142.250.194.142:443    TIME_WAIT      0
  TCP    192.168.1.14:64700     172.217.167.193:443    TIME_WAIT      0
  TCP    192.168.1.14:64701     142.250.194.166:443    TIME_WAIT      0
  TCP    192.168.1.14:64706     13.107.42.12:443       ESTABLISHED    7276
  TCP    192.168.1.14:64707     162.159.129.81:443     TIME_WAIT      0
  TCP    192.168.1.14:64708     162.159.136.84:443     TIME_WAIT      0
  TCP    192.168.1.14:64709     162.159.136.84:443     TIME_WAIT      0
  TCP    192.168.1.14:64710     172.64.152.80:443      TIME_WAIT      0
  TCP    192.168.1.14:64711     13.89.178.27:443       ESTABLISHED    7276

C:\Users\smali>
```

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.

```html
<html>
  <head>
    <title>Multiplication Table</title>
  </head>
  <body>
  <p id="2" style = "font-size:20px; color:black;"></p>
  <p id="3" style = "font-size:25px; color:violet;"></p>
  <p id="4" style = "font-size:30px; color:indigo;"></p>
  <p id="5" style = "font-size:35px; color:blue;"></p>
  <p id="6" style = "font-size:40px; color:green;"></p>
  <p id="7" style = "font-size:45px; color:yellow;"></p>
  <p id="8" style = "font-size:50px; color:orange;"></p>
  <p id="9" style = "font-size:55px; color:red;"></p>
```

```html
<p id="10" style = "font-size:60px; color:black;"></p>
<script>
    let table = 2;
    var intv = setInterval(begin,2000);
    function begin(){
        var result  ='';
        var tmp = 1;
        for(let i = 1; i<=10; i++){
            tmp = table*i;
            result = result.concat(tmp.toString());
            result = result.concat(' ');
        }
        if(table>9){
            clearInterval(intv);
            document.getElementById(table.toString()).innerHTML = result;
        }else{
            document.getElementById(table.toString()).innerHTML = result;
            table++;
        }
    }
</script>
</body>
</html>
```

2 4 6 8 10 12 14 16 18 20

3 6 9 12 15 18 21 24 27 30

4 8 12 16 20 24 28 32 36 40

5 10 15 20 25 30 35 40 45 50

6 12 18 24 30 36 42 48 54 60

7 14 21 28 35 42 49 56 63 70

8 16 24 32 40 48 56 64 72 80

9 18 27 36 45 54 63 72 81 90

10 20 30 40 50 60 70 80 90 100

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.

```
<html>
  <style>
    #grad1 {
  background-color: red;
  background-image: linear-gradient(to right, red, orange, yellow, green, blue, indigo, violet);
}
    </style>
<head>
  <title>About me</title>
</head>
<div id="grad1">
<body>
```

```html
    <div style=" text-align:center; margin:auto; font-size: 25px; font-
weight:bold">

    <h1 style="color: darkblue;">My Origin</h1>

    <p style="color:blueviolet;">I am from Kolkata, West Bengal</p>

    <h1 style="color:darkblue;">My Course</h1>

    <p style="color:blueviolet;">B.Sc(H) Computer Science</p>

    <h1 style="color:darkblue;">My Hobbies</h1>

    <p style="color:blueviolet;">Gaming and reading</p>

    <h1 style="color:darkblue;">My Address</h1>

    <p style="color:blueviolet;">Saket, New Delhi</p>

    <h1 style="color:darkblue;">My Plans</h1>

    <p style="color:blueviolet;">Masters in computer science</p>
</div>
</body>
</div>
</html>
```
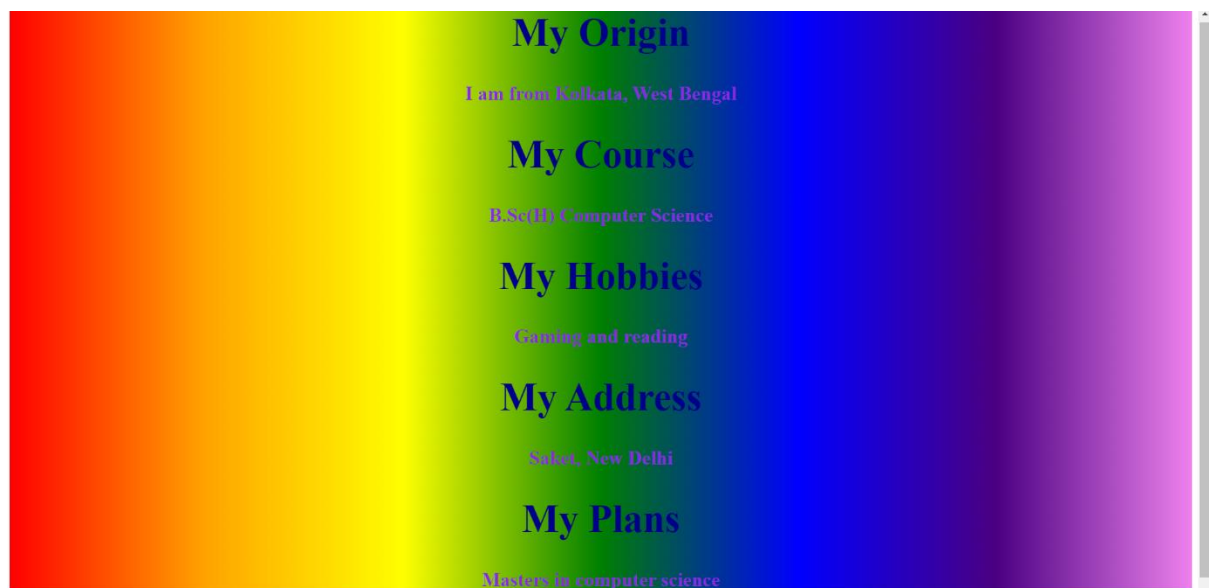
5. Create an HTML page with a paragraph written on it and under which 9 buttons are placed in a 3X3 grid. The first row is for buttons labeled with colors names Red, Green, and Blue, the second row with numbers 10, 20, 30, and the third row with different font names. Click event of each of the buttons should make the appropriate change in the style of paragraph.

```html
<html>
  <head>
    <style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
}
.grid-item {
  border: 1px solid rgba(0, 0, 0, 0.8);
  font-size: 30px;
  text-align: center;
}
</style>
  </head>
  <body>
  <div style='text-align: center; width: 60%;margin-left: 25%;'>
  <p id='demo' style='font-size: 20px; font-family: Arial, Helvetica, sans-serif;'>
      Contrary to popular belief, Lorem Ipsum is not simply random text.
      It has roots in a piece of classical Latin literature from 45 BC,
      making it over 2000 years old.
      Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia,
      looked up one of the more obscure Latin words, consectetur,
```

from a Lorem Ipsum passage, and going through the cites of the word in

classical literature, discovered the undoubtable source.

Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum

et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC.

This book is a treatise on the theory of ethics, very popular during the

Renaissance.

  </p>
  <div class="grid-container">
  <div class="grid-item"><p><button onclick="change(1)" class='button buttnblue'>Blue</button>

  </p></div>
  <div class="grid-item"><p><button onclick="change(2)" class='button buttnviolet'>Violet</button>

  </p></div>
  <div class="grid-item"><p><button onclick="change(3)" class='button buttnorange'>Orange</button>

  </p></div>
  <div class="grid-item"> <p><button onclick="change(4)" class='button buttnsz2'>15</button>

  </p></div>
  <div class="grid-item"><p><button onclick="change(5)" class='button buttnasz1'>25</button>

  </p></div>
  <div class="grid-item"><p><button onclick="change(6)" class='button buttnsz3'>35</button>

  </p></div>
  <div class="grid-item"> <p><button onclick="change(7)" class='button buttnft1'>Garamond</button>

```
    </p></div>
  <div class="grid-item"><p><button onclick="change(8)" class='button
buttnft2'>Courier New</button>

    </p></div>
  <div class="grid-item"> <p><button onclick="change(9)" class='button
buttnft3'>Georgia</button>

    </p>

    </div>

</div>
  <script>
      function change(buttonId){
        if(buttonId==1){
          document.getElementById('demo').style.color = 'blue';
        }
        if(buttonId==2){
          document.getElementById('demo').style.color = 'violet';
        }
        if(buttonId==3){
          document.getElementById('demo').style.color = 'orange';
        }
        if(buttonId==4){
          document.getElementById('demo').style.fontSize = '15px';
        }
        if(buttonId==5){
          document.getElementById('demo').style.fontSize = '25px';
        }
        if(buttonId==6){
```

```javascript
        document.getElementById('demo').style.fontSize = '35px';
    }
    if(buttonId==7){
        document.getElementById('demo').style.fontFamily = 'Garamond';
    }
    if(buttonId==8){
        document.getElementById('demo').style.fontFamily = 'Courier New';
    }
    if(buttonId==9){
        document.getElementById('demo').style.fontFamily = 'Georgia';
    }
}
    </script>
    </body>
</html>
```

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance.

Blue	Violet	Orange
15	25	35
Garamond	Courier New	Georgia

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance.

Blue	Violet	Orange
15	25	35
Garamond	Courier New	Georgia

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance.

Blue	Violet	Orange
15	25	35
Garamond	Courier New	Georgia

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance.

Blue	Violet	Orange
15	25	35
Garamond	Courier New	Georgia

6. Create a form that takes data about a pet. The form must be well designed and should accept the pet's name, age, weight, type, 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.

```html
<!DOCTYPE html>
<html>
  <head>
    <!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>Document</title>
    </head>
    <body>

    </body>
    </html>
      <style>
          html {
      background-color: #BFF4BF;
    }
    input[type=text],
    select,
    textarea,
    input[type=number] {
     width: 70%;
     padding: 12px;
     border: 1px solid whitesmoke;
```

```css
  border-radius: 4px;

  box-sizing: border-box;

  margin-top: 6px;

  margin-left: 10%;

  margin-bottom: 16px;

  resize: vertical

}

button {

  background-color: green;

  color: white;

  padding: auto;

  border-radius: 4px;

  cursor: pointer;

  margin-left: 30%;

}

label {

  font-size: xx-large;

  margin-left: 20%;

  padding: auto;

}


.container {

  width: 30%;

  margin-left: 25%;

  padding: 20px;

}
```

```html
    </style>
  </head>
  <body>
    <div class="container">
      <form id="pet">
        <label for="petName">Name : </label><br />
  <input type="text" id="name" /><br />
  <label for="age">Age : </label><br />
  <input type="number" id="Age" /><br />
  <label for="weight">Weight : </label><br />
  <input type="number" id="Weight" /><br />
  <label for="type">Type : </label><br />
  <input type="text" id="Type" /><br />
  <label for="thing">Favourite Thing : </label><br />
  <input type="text" id="Thing" /><br /><br />
  <button type="button" onclick="submit()">Submit</button>
      </form>
    </div>
    <script>
      function submit(){
        const pt = {
          name: document.getElementById("Name").value,
          age: document.getElementById("Age").value,
          weight: document.getElementById("Weightt").value,
          type: document.getElementById("Type").value,
          favourite: document.getElementById("Thing").value,
```

```
                };

                console.log(pt);

                const jSONPet = JSON.stringify(pt);

                console.log(jSONPet);

                document.getElementById("pet").reset;

            }

        </script>

        </body>

    </html>
```

Name :

Age :

Weight :

Type :

Favourite Thing :

Submit

Name :

Zophia

Age :

8

Weight :

12

Type :

Bird

Favourite Thing :

Spice

Submit

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.

JSON File:

```json
[
    {
        "name": "Zophia",
        "age": "5",
        "weight": "11",
        "type": "Bird",
        "thing": "Spice"
    },
    {
        "name": "Kiira",
        "age": "5",
        "weight": "22",
        "type": "Dog",
        "thing": "Chicken"
    }
]
```

HTML File:

```html
<html>
    <head>
        <!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>Pet</title>
            <style>
                html {
            background-color: orangered;
        }

        table {
            text-align: center;
            width: 100%;
            border: 1px solid;
        }

        th {
```

```css
            font-size: xx-large;
            text-decoration: underline;
        }

    tbody tr:hover {
        background-color: tomato;
        font-size: x-large;
    }

    tr {
        font-size: large;
    }

    .container {
        width: 50%;
        margin-left: 25%;
        margin-top: 5%;
        border-radius: 5px;
        background-color: yellow;
        padding: 20px;
    }
        </style>
</head>
<body>
    <p style="text-align: center; font-size: 100px;">PET</p>
    <div class="container">
        <table><thead>
            <th>Name</th>
            <th>Age</th>
            <th>Weight</th>
            <th>Type</th>
            <th>Favourite Thing</th>
        </thead>
            <tbody id="a">
            </tbody>
        </table>
    </div>
    <script>
        setTimeout(loadDoc(), 0);
        function loadDoc() {
            fetch('http://localhost:8000/data.json')
            .then((response) => response.json())
            .then((json) => {
                var temp = "";
                for (var i = 0; i < json.length; i++) {
                    temp = temp.concat("<tr><td>" + json[i].name +
"</td><td>" +
```

```
                              json[i].age + "</td><td>" + json[i].weight +
"</td><td>" +
                              json[i].type + "</td><td>" + json[i].thing +
                              "</td></tr>");
                    document.getElementById("a").innerHTML = temp;
              }
          });
          }
      </script>
   </body>
</html>
```


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.

CSS File:

.htmlClass {

    background: linear-gradient(to right, #d95d39, #946569);

}

.divClass {

```css
    text-align: center;

}


.tableClass {

    margin-left: auto;

    margin-right: auto;

}


.th {

    font-size: 30px;

    color: #474350;

    font-style: Georgia;


}


.td {

    font-size: 30px;

}


.heading {

    font-size: 70px;

    font-family: Cambria, serif;

    text-decoration:blanchedalmond;

}


.fee {
```

```css
    font-size: 60px;

    font-family: Times,serif;

}


.future{

    font-size: 40px;

    text-decoration:underline dotted;

}


.futureClass{

    font-size: 40px;

    text-decoration:underline dotted;

}


.element.criteria {

    font-family: 'Courier New', Courier, monospace;

    font-weight: 1000;

    font-style: oblique;

    font-size: x-large;

    color: firebrick;

}


.element.eligibility {

    font-family: 'Courier New', Courier, monospace;

    font-weight: 1000;

    font-style: oblique;
```

```
        font-size: large;
}
```

HTML File:

```html
<!DOCTYPE html>
<head>
    <!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>Computer Science</title>
        <link rel="stylesheet" href="design.css">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
    <script src="query.js"></script>
    </head>
    <body>
        <div id="main">
            <h1 id="heading">B.Sc. Hons CS</h1>
            <p id="fee">
                Fee : ₹ 80,000
            </p>
            <p> <element id="criteria">Eligibility Criteria :</element><element id="eligibility"> A total of no less than 60% marks in English, Mathematics and best of the two
```

subjects

from Physics, Chemistry, Computer Science/Informatics Practices

OR

A total of no less than 60% marks in English, Mathematics and two subjects (other than Physics, Chemistry,

Computer Science/Informatics Practices) from List B with a deduction of 1% per subject in the aggregate.

```
        </element>
      </p>
      <p>
      <div id="tableContainer">
        <table>
          <thead>
            <th>S. No.</th>
            <th>Name</th>
            <th>Credit</th>
          </thead>
          <tbody>
            <tr>
              <td>1.</td>
              <td>
                Computer System Architecture
              </td>
              <td>
                4
              </td>
            </tr>
```

```html
<tr>
    <td>2.</td>
    <td>
        C++ Programming
    </td>
    <td>
        4
    </td>
</tr>
<tr>
    <td>3.</td>
    <td>
        Programming in Java
    </td>
    <td>
        4
    </td>
</tr>
<tr>
    <td>4.</td>
    <td>
        Discrete Structure
    </td>
    <td>
        4
    </td>
</tr>
```

```html
</tr>
<tr>
  <td>5.</td>
  <td>
    Data Structures
  </td>
  <td>
    5
  </td>
</tr>
<tr>
  <td>6.</td>
  <td>
    Operating System
  </td>
  <td>
    5
  </td>
</tr>
<tr>
  <td>7.</td>
  <td>
    Computer Networks
  </td>
  <td>
    5
```

```html
      </td>
  </tr>
  <tr>
    <td>8.</td>
    <td>
      Design And Analysis of Algorithm
    </td>
    <td>
      5
    </td>
  </tr>
  <tr>
    <td>9.</td>
    <td>
      Software Engineering
    </td>
    <td>
      5
    </td>
  </tr>
  <tr>
    <td>10.</td>
    <td>
      Database Management Systems
    </td>
    <td>
```

```
      5
    </td>
  </tr>
  <tr>
    <td>11.</td>
    <td>
      Internet Technologies
    </td>
    <td>
      4
    </td>
  </tr>
  <tr>
    <td>12.</td>
    <td>
      Theory of Computation
    </td>
    <td>
      5
    </td>
  </tr>
  <tr>
    <td>13.</td>
    <td>
      Artificial Intelligence
    </td>
```

```html
            <td>
                4
            </td>
        </tr>
        <tr>
            <td>14.</td>
            <td>
                Computer Graphics
            </td>
            <td>
                5
            </td>
        </tr>
        </tbody>
    </table>
</div>
</p>
<p id="future">
    <element id="futureElement">Future Possibilities :
    </element>
    You can become someone in field of Computer.
</p>
<p><button onclick="letsChange()">Re-Design</button></p>
</div>
</body>
</html>
```

B.Sc. Hons CS

Fee : ₹ 80,000

Eligibility Criteria : A total of no less than 60% marks in English, Mathematics and best of the two subjects from Physics, Chemistry, Computer Science/Informatics Practices OR A total of no less than 60% marks in English, Mathematics and two subjects (other than Physics, Chemistry, Computer Science/Informatics Practices) from List B with a deduction of 1% per subject in the aggregate.

S. No.	Name	Credit
1.	Computer System Architecture	4
2.	C++ Programming	4
3.	Programming in Java	4
4.	Discrete Structure	4
5.	Data Structures	5
6.	Operating System	5
7.	Computer Networks	5
8.	Design And Analysis of Algorithm	5
9.	Software Engineering	5
10.	Database Management Systems	5
11.	Internet Technologies	4
12.	Theory of Computation	5
13.	Artificial Intelligence	4
14.	Computer Graphics	5

Future Possibilities : You can become someone in field of Computer.

Re-Design

B.Sc. Hons CS

Fee : ₹ 80,000

Eligibility Criteria : A total of no less than 60% marks in English, Mathematics and best of the two subjects from Physics, Chemistry, Computer Science/Informatics Practices OR A total of no less than 60% marks in English, Mathematics and two subjects (other than Physics, Chemistry, Computer Science/Informatics Practices) from List B with a deduction of 1% per subject in the aggregate.

S. No.	Name	Credit
1.	Computer System Architecture	4
2.	C++ Programming	4
3.	Programming in Java	4
4.	Discrete Structure	4
5.	Data Structures	5
6.	Operating System	5
7.	Computer Networks	5
8.	Design And Analysis of Algorithm	5
9.	Software Engineering	5
10.	Database Management Systems	5
11.	Internet Technologies	4
12.	Theory of Computation	5
13.	Artificial Intelligence	4
14.	Computer Graphics	5

Future Possibilities : You can become someone in field of Computer.

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.

```
<!DOCTYPE html>

<html lang="en">


<head>
```

```html
<meta charset="UTF-8">

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

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

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/css/bootstrap.min.css" rel="stylesheet">

<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/js/bootstrap.bundle.min.js"></script>

<title>Image Gallery</title>


<script>
    $(function () {
        $("img").addClass("img-fluid imgClass");
    })
</script>
<style>
    body {
        background-image: url("catart_html.jpg");
        background-repeat: no-repeat;
        background-attachment: fixed;
        background-size: cover;
    }

    .imgClass {
```

```
            border: 10px solid;

            margin: 1em;

            width: 910px;

            height: 480px;

        }


        p {

            font-size: xx-large;

            font-style: oblique;

            font-weight: 500;

            text-align: center;

            margin: 1%;

        }

    </style>


</head>


<body>


    <p>Image Gallery</p>

    <img

        src= persiancat.jpg>

    <img src= siamesecat.jpg>

    <img

        src=munchkincat.jpg>

        <img src=scottishcat.jpg>
```

```
    <img src=sphinxcat.jpg>

  </body>


</html>
```


Image Gallery

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.

```
var http = require('http');


http.createServer(function (req, res) {

    res.write('Hi , This is a Node.js server');

    res.end();

  }).listen(10000);
```

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.

CSS File:

```css
html {
  background-color: darkslateblue;
}

input[type=text],
select,
textarea,
input[type=number],
input[type=email],input[type=password]  {
  width: 70%;
  padding: 12px;
  border: 1px solid #ccc;
  border-radius: 4px;
  box-sizing: border-box;
  margin-top: 6px;
  margin-left: 10%;
  margin-bottom: 16px;
  resize: vertical
}

button {
  background-color: blueviolet;
  color: white;
  padding: 12px 20px;
  border: none;
  border-radius: 4px;
```

```css
    cursor: pointer;

    margin-left: 10%;

}


.buttonGroup{

    margin-left: auto;

    margin-right: auto;

}


button:hover {

    background-color: darkblue;

}


label {

    font-size: xx-large;

    margin-left: 10%;

}


.container {

    width: 50%;

    margin-left: 25%;

    margin-top: 10%;

    border-radius: 5px;

    background-color: lightblue;

    padding: 20px;

}
```

JS File:

```
var signInForm = true;

var signUp = '<label for="firstName">First Name : </label><br />   <input
type="text" id="firstName" /><br />   <label for="lastName">Last Name :
</label><br />   <input type="text" id="lastName" /><br />   <label
for="emailID">Email ID : </label><br /><input type="email" id="emailID"
autocomplete="username"/><br /><label for="password">Password :
</label><br /><input type="password" id="password" autocomplete="new-
password"/><br /><label for="confirmPassword">Confirm Password :
</label><br /><input type="password" id="confirmPassword"
autocomplete="new-password"/><br /><button type="button"
onClick="createAccount()">Create Account</button><button type="button"
onclick="changeForm()">Login</button>';

var signIn = '<label for="emailID">Email ID : </label><br /><input type="email"
id="emailID" autocomplete="username"/><br /><label
for="password">Password : </label><br /><input type="password"
id="password" autocomplete="current-password"/><br /><div
class="buttonGroup"><button type="button"
onClick="login()">Login</button><button type="button"
onclick="changeForm()">Sign Up</button></div>'


$(function () {
    document.getElementById('changeState').innerHTML = signIn;
});


const HTTP = new XMLHttpRequest();

const url = 'http://localhost:10000/';

HTTP.open("get", url + 'welcome');

HTTP.send();
```

```javascript
function changeForm() {

    if (signInForm) {

        document.getElementById('changeState').innerHTML = signUp;

        signInForm = false;

    } else {

        document.getElementById('changeState').innerHTML = signIn;

        signInForm = true;

    }

}


function login() {

    var password = document.getElementById("password").value;

    var emailId = document.getElementById("emailID").value;

    console.log(password, emailId);

    HTTP.open("GET", url + 'authenticate/' + emailId + '/' + password, false);

    HTTP.send();

    console.log(HTTP.responseText);

    if (HTTP.responseText === 'true') {

        welcomeUser();

    }

    else {

        confirm("Credential Invalid");

    }

}


function createAccount() {
```

```javascript
    var password = document.getElementById("password").value;

    var emailId = document.getElementById("emailID").value;

    var confirmPassword =
document.getElementById("confirmPassword").value;

    var name = document.getElementById("firstName").value + " " +
document.getElementById("lastName").value;

    if (confirmPassword === password) {

        HTTP.open("POST" , url+'new/' + name +'/' +emailId +'/' + password , false
);

        console.log( url+'new/' + name +'/' +emailId +'/' + password);

        HTTP.send();

        console.log(HTTP.responseText);

        if(HTTP.responseText === 'true'){

            confirm("Account Created Successfully");

            location.href = "index.html";

        }else{

            confirm("Email ID already in use.Use different Email ID.")

        }

    }

    else {

        confirm("Password didn't match");

    }

}


function welcomeUser() {

    var welcome = '<p style = "font-size: xx-large">Welcome User !!</p>';

    document.getElementById('changeState').innerHTML = welcome;
```

```
        }

NodeJs File:

var mysql = require('mysql');

var express = require('express');

var app = express();


app.listen(10000, (err) => {

    if (err) throw err;

    console.log("Serever listening at http://127.0.0.1:10000");

})


app.use(function (_req, res, next) {

    res.setHeader('Access-Control-Allow-Origin', '*');

    res.setHeader('Access-Control-Allow-Methods', 'GET, POST, PUT, DELETE');

    res.setHeader('Access-Control-Allow-Headers', 'Origin, X-Requested-With,
Content-Type, Accept');

    res.setHeader('Access-Control-Allow-Credentials', true);

    next();

});

app.get('/welcome', (_req, res) => {

    res.write("API Working");

    res.end();

})

app.get('/authenticate/:userID/:password', function (req, res) {

    var email = req.params.userID.toString();

    var password = req.params.password.toString();
```

```javascript
        var customQuery = "select password from userdata where email = '" + email
+ "'";
        con.query(customQuery, function (_err, result) {
            if (result.length > 0) {
                if (result[0].password === password) {
                    res.send(true);
                    res.end();
                } else {
                    res.send(false);
                    res.end();
                }
            } else {
                res.send(false);
                res.end();
            }
        })
})
app.post('/new/:name/:userID/:password', function (req, res) {
    var name = req.params.name.toString();
    var email = req.params.userID.toString();
    var password = req.params.password.toString();
    var customQuery = "insert into userdata (name , email , password) values ('"
+ name + "','" + email + "','" + password + "')";
    con.query(customQuery, function (err, result) {
        if (err) {
            res.send(false);
        } else {
```

```javascript
                res.send(true);

                console.log("User Created Successfully");

            }
            res.end();

        })
    })


    var con = mysql.createConnection({

        host: "hosting",

        user: "serverroot",

        password: "123456"

    });


    con.connect(function (err) {

        if (err) throw err;

        con.query("use prog11", function (_err, result) {

            console.log(result.message)

        })
        console.log("Connected!");

    });
```
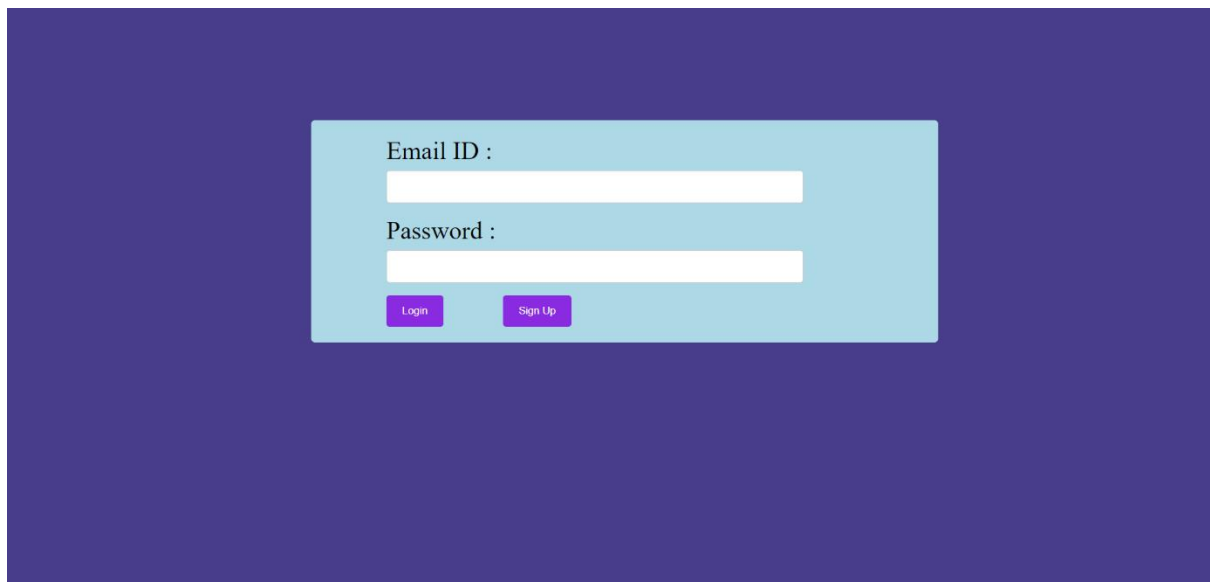
HTML File:

```html
<!DOCTYPE html>

<html lang="en">


<head>
```

```html
    <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>Index

    </title>

    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

    <script src="prog11.js"></script>

    <link rel="stylesheet" href="prog11.css">
</head>


<body>
    <form class="container" id="changeState"></form>
</html>
```

Email ID :

abcd@xyz.com

Password :

••••••••

[Login] [Sign Up]

First Name :

Last Name :

Email ID :

Password :

Confirm Password :

[Create Account] [Login]