

INTERNET TECHNOLOGIES PRACTICAL FILE



Q.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.

OUTPUT

```
Administrator: Command Prompt

C:\WINDOWS\system32>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

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

Ethernet adapter vEthernet (Default Switch):

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::c5d2:2e2f:2e86:dd5%39
    IPv4 Address. . . . . : 172.27.16.1
    Subnet Mask . . . . . : 255.255.240.0
    Default Gateway . . . . . :

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::b47f:ee4a:c7f6:f992%5
    IPv4 Address. . . . . : 10.210.103.86
    Subnet Mask . . . . . : 255.254.0.0
    Default Gateway . . . . . : 10.210.0.1

Ethernet adapter Bluetooth Network Connection:

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

Q.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>ping canva.com

Pinging canva.com [104.17.114.17] with 32 bytes of data:
Reply from 104.17.114.17: bytes=32 time=3ms TTL=62
Reply from 104.17.114.17: bytes=32 time=4ms TTL=62
Reply from 104.17.114.17: bytes=32 time=4ms TTL=62
Reply from 104.17.114.17: bytes=32 time=3ms TTL=62

Ping statistics for 104.17.114.17:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 3ms, Maximum = 4ms, Average = 3ms

C:\WINDOWS\system32>tracert google.com

Tracing route to google.com [142.250.206.142]
over a maximum of 30 hops:

  0  6 ms    3 ms    3 ms  10.210.0.1
  1  27 ms    6 ms    4 ms  74.125.52.6
  2   5 ms    4 ms    4 ms  142.251.77.25
  3   4 ms    3 ms    7 ms  142.251.76.197
  4   6 ms    6 ms    4 ms  del11s21-in-f14.1e100.net [142.250.206.142]

Trace complete.

```

```

Administrator: Command Prompt

C:\WINDOWS\system32>netstat -a

Active Connections

 Proto Local Address           Foreign Address         State
----
TCP    0.0.0.0:135             DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:445             DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:2179            DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:5040            DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:5357            DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:49664           DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:49665           DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:49666           DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:49667           DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:49668           DESKTOP-4RFANP6:0      LISTENING
TCP    0.0.0.0:49669           DESKTOP-4RFANP6:0      LISTENING
TCP    10.210.103.86:139       DESKTOP-4RFANP6:0      LISTENING
TCP    10.210.103.86:53321     del12s07-in-f14:https  TIME_WAIT
TCP    10.210.103.86:53322     del11s14-in-f3:https  TIME_WAIT
TCP    10.210.103.86:53324     del11s10-in-f10:https TIME_WAIT
TCP    10.210.103.86:53428     a104-120-86-116:http  TIME_WAIT
TCP    10.210.103.86:53430     20.198.119.84:https    ESTABLISHED
TCP    10.210.103.86:53432     sd-in-f188:5228        ESTABLISHED
TCP    10.210.103.86:53447     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53450     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53451     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53452     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53453     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53454     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53455     13.107.21.200:https    TIME_WAIT
TCP    10.210.103.86:53458     13.107.21.200:https    ESTABLISHED
TCP    10.210.103.86:53463     104.43.52.220:https    TIME_WAIT
TCP    10.210.103.86:53465     13.107.246.254:https    ESTABLISHED
TCP    10.210.103.86:53466     52.113.196.254:https    ESTABLISHED
TCP    10.210.103.86:53467     204.79.197.239:https    ESTABLISHED
TCP    10.210.103.86:53468     52.123.129.254:https    ESTABLISHED
TCP    10.210.103.86:53469     204.79.197.222:https    ESTABLISHED
TCP    10.210.103.86:53488     8.241.139.254:http     ESTABLISHED
TCP    10.210.103.86:53489     a23-49-78-32:http      ESTABLISHED
TCP    10.210.103.86:53490     8.241.139.254:http     ESTABLISHED
TCP    127.0.0.1:1001           DESKTOP-4RFANP6:0      LISTENING
TCP    127.0.0.1:10000          DESKTOP-4RFANP6:0      LISTENING
TCP    127.0.0.1:58573         DESKTOP-4RFANP6:0      LISTENING

```

C:\ Administrator: Command Prompt

TCP	127.0.0.1:10000	DESKTOP-4RFANP6:0	LISTENING
TCP	127.0.0.1:58573	DESKTOP-4RFANP6:0	LISTENING
TCP	172.27.16.1:139	DESKTOP-4RFANP6:0	LISTENING
TCP	:::135	DESKTOP-4RFANP6:0	LISTENING
TCP	:::445	DESKTOP-4RFANP6:0	LISTENING
TCP	:::2179	DESKTOP-4RFANP6:0	LISTENING
TCP	:::5357	DESKTOP-4RFANP6:0	LISTENING
TCP	:::49664	DESKTOP-4RFANP6:0	LISTENING
TCP	:::49665	DESKTOP-4RFANP6:0	LISTENING
TCP	:::49666	DESKTOP-4RFANP6:0	LISTENING
TCP	:::49667	DESKTOP-4RFANP6:0	LISTENING
TCP	:::49668	DESKTOP-4RFANP6:0	LISTENING
TCP	:::49669	DESKTOP-4RFANP6:0	LISTENING
UDP	0.0.0.0:53	*:*	
UDP	0.0.0.0:3702	*:*	
UDP	0.0.0.0:3702	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5353	*:*	
UDP	0.0.0.0:5355	*:*	
UDP	0.0.0.0:50171	*:*	
UDP	0.0.0.0:51076	*:*	
UDP	0.0.0.0:52702	*:*	
UDP	0.0.0.0:53866	*:*	
UDP	0.0.0.0:54824	*:*	
UDP	0.0.0.0:55002	*:*	
UDP	0.0.0.0:55337	*:*	
UDP	0.0.0.0:55877	*:*	
UDP	0.0.0.0:55952	*:*	
UDP	0.0.0.0:56385	*:*	
UDP	0.0.0.0:56608	*:*	
UDP	0.0.0.0:56738	*:*	
UDP	0.0.0.0:56995	*:*	
UDP	0.0.0.0:58213	*:*	
UDP	0.0.0.0:58235	*:*	
UDP	0.0.0.0:58604	*:*	
UDP	0.0.0.0:58605	*:*	
UDP	0.0.0.0:58981	*:*	

C:\ Administrator: Command Prompt

```
UDP    0.0.0.0:58604      *: *
UDP    0.0.0.0:58605      *: *
UDP    0.0.0.0:58981      *: *
UDP    0.0.0.0:60785      *: *
UDP    0.0.0.0:62631      *: *
UDP    0.0.0.0:62921      *: *
UDP    0.0.0.0:62924      *: *
UDP    0.0.0.0:64071      *: *
UDP    0.0.0.0:64661      *: *
UDP    0.0.0.0:64687      *: *
UDP    0.0.0.0:65007      *: *
UDP    0.0.0.0:65514      *: *
UDP    10.210.103.86:137   *: *
UDP    10.210.103.86:138   *: *
UDP    10.210.103.86:1900  *: *
UDP    10.210.103.86:50266 *: *
UDP    127.0.0.1:1900      *: *
UDP    127.0.0.1:49664      *: *
UDP    127.0.0.1:50267      *: *
UDP    172.27.16.1:67      *: *
UDP    172.27.16.1:68      *: *
UDP    172.27.16.1:137     *: *
UDP    172.27.16.1:138     *: *
UDP    172.27.16.1:1900    *: *
UDP    172.27.16.1:50265   *: *
UDP    [::]:3702             *: *
UDP    [::]:3702             *: *
UDP    [::]:5353             *: *
UDP    [::]:5353             *: *
UDP    [::]:5353             *: *
UDP    [::]:5353             *: *
UDP    [::]:5353             *: *
UDP    [::]:5355             *: *
UDP    [::]:55953           *: *
UDP    [::]:58606           *: *
UDP    [::1]:1900           *: *
UDP    [::1]:50264           *: *
UDP    [fe80::b47f:ee4a:c7f6:f992%5]:1900 *: *
UDP    [fe80::b47f:ee4a:c7f6:f992%5]:50263 *: *
UDP    [fe80::c5d2:2e2f:2e86:dd5%39]:1900 *: *
UDP    [fe80::c5d2:2e2f:2e86:dd5%39]:50262 *: *
```

Q.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 it looks nice.

Code :

```
<!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>About me</title>
  <link rel="stylesheet" href="./Q3_index.css" />
</head>

<body>
  <header class="tophead">
    <p class="tophead-intro">Hello People, this is</p>
    <h1 class="tophead-heading GradientText">Nitish
Kumar</h1>
  </header>
  <section class="about-section">
    <h1>About me</h1>
    <p>
      I'm a final year student of Bsc (Hons.) Computer
Science at Ramanujan College, University of Delhi.
    </p>
  </section>

  <section class="hobbies-section">
    <h1>My Hobbies</h1>
    <ul>
      <li>Reading Books </li>
      <li>Learn about Psychology & Neurological behaviour
</li>
      <li>Watching Movies & Anime </li>
      <li>Gardening </li>
    </ul>
  </section>

  <section class="course-section">
    <h1>My Course</h1>
    <p>
      <br>Things that I have learnt so far:
    <div class="course-list">
      <ul>
        <li>C++    &emsp;&ensp; &emsp;
          <progress value="50" max="100">
        </progress></li>
```

[illegible]

CSS:

```
.GradientText{
    background: linear-gradient(to left, indigo, rgb(223, 20,
20), rgb(20, 240, 42));
    -webkit-background-clip: text;
```

```
-webkit-border-image: auto;
-webkit-text-fill-color: transparent;
}

.GradientText:hover{
  background: linear-gradient(to bottom right, #22f13ad9,
#712cbbd9, #f46e07);
  -webkit-background-clip: text;
  -webkit-border-image: auto;
  -webkit-text-fill-color: transparent;
}

body {
  color: #902222;
  font-size: 1em;
  font-family: "Open Sans", "Helvetica Neue", sans-serif;
}

.tophead-intro {
  margin-bottom: 0.1em;
  font-family: "Gentium Book Basic", Georgia, serif;
  font-size: 2em;
}

.tophead-heading {
  color: #f1c863;
  margin-top: -0.2em;
  font-family: "Open Sans", "Helvetica Neue", sans-serif;
  font-weight: bold;
  font-size: 6em;
  letter-spacing: -0.02em;
  text-transform: uppercase;
}

.about-section,
.hobbies-section,
.course-section,
.plan-section {
  max-width: 38em;
  margin-left: auto;
  margin-right: auto;
  margin-top: 2em;
}

.about-section > p,
.hobbies-section > p,
.course-section,
.plan-section > p,
.content-footer > p {
  font-weight: 300;
  letter-spacing: 0.05em;
}
```



```
section > h1 {
  margin-top: 2em;
}

.content-footer,
.tophead-heading,
.tophead-intro {
  text-align: center;
  margin-top: 2em;
}

.tophead {
  display: -moz-box;
  background-size: cover;
  background-repeat: no-repeat;
  border-radius: 10;
  border: 12pt #1c9a3c;
  border-style: groove;
}

.tophead-intro {
  margin-bottom: 0.1em;
  font-family: "Gentium Book Basic", Georgia, serif;
  font-size: 2em;
}

.tophead-heading {
  margin-top: -0.2em;
  font-family: "Open Sans", "Helvetica Neue", sans-serif;
  font-weight: bold;
  font-size: 6em;
  letter-spacing: -0.02em;
  text-transform: uppercase;
}

.about-section > p,
.hobbies > p,
.content-footer > p {
  font-weight: 300;
  letter-spacing: 0.05em;
}

h2 {
  font-family: "Gentium Book Basic", Georgia, serif;
  font-size: 1.2em;
  font-weight: bold;
}

.about-section,
.hobbies-section,
.course-section,
.plan-section {
  max-width: 38em;
}
```

```

margin-left: auto;
margin-right: auto;
margin-top: 2em;
}
.content-footer {
margin: 50px 0;
padding: 20px 0;
text-align: center;
background-image: linear-gradient(to left top, rgb(114, 17,
183, 0.7), rgb(173, 15, 28),rgb(24, 40, 191));
border: 1px dashed black;
border-radius: 15px;
}
.social > li {
display: inline-block;
margin: 0 5px;
}
.content-footer > p {
color: #f7edel;
}
a {
font-weight: bold;
text-decoration: none;
color: rgb(248, 244, 250);
}

a:hover{
color: chartreuse;
}

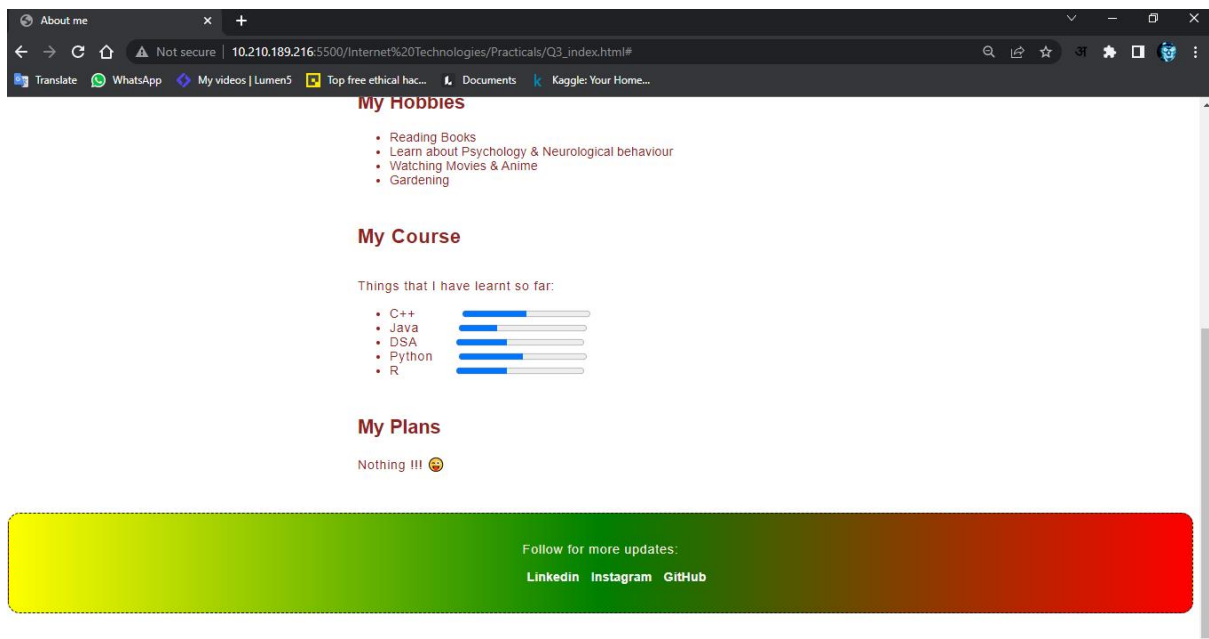
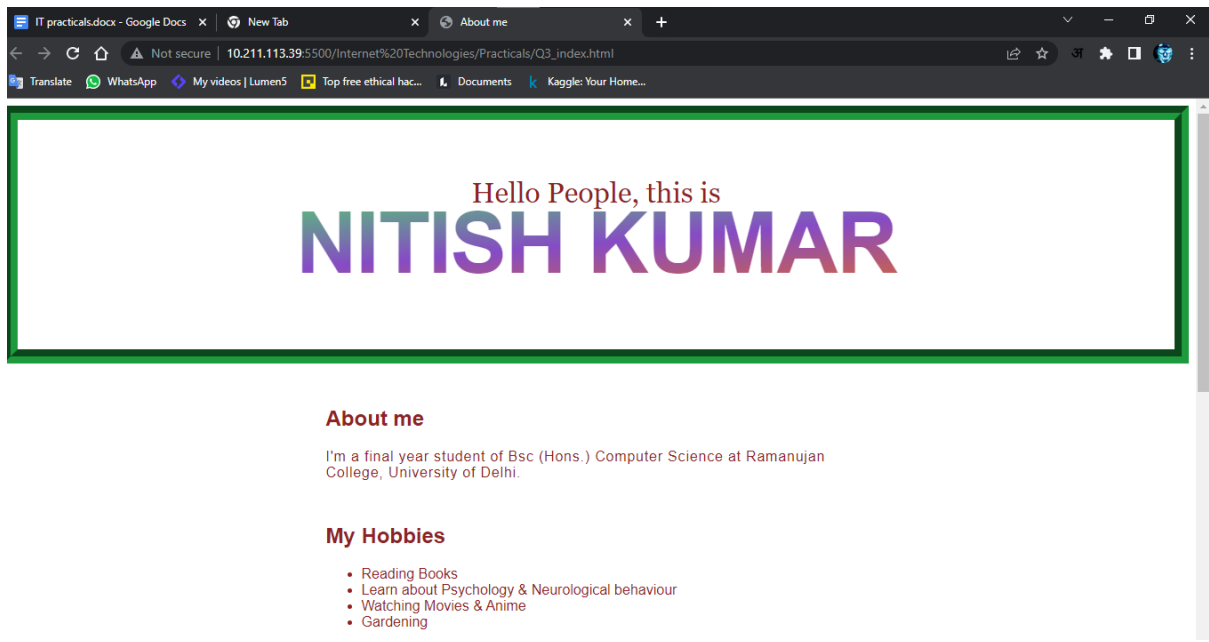
@media only screen and (max-width: 500px) {
.tophead {
padding: 3em 0;
}

.tophead-intro{
font-size: 2em;
}
.tophead-heading {
font-size: 3em;
}
.content-footer {
padding: 2em 2.5em;
}
}

.content-footer:hover{
background-image: linear-gradient(to left,red,green, yellow);
}

```

OUTPUT:



Q.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 :

```
<!DOCTYPE html>
<html>
<head>
  <title>Practical 4</title>
  <meta charset="utf-8">
  <style type="text/css">
    table{
      border: 1px solid black;
      border-collapse: collapse;
      width: 80%;
      margin: auto;
    }
    td,th{
      border: 1px solid black;
      padding: 5px;
      /* border-collapse: collapse; */
    }
    .center{
      text-align: center;
    }
  </style>
</head>
<body>
  <h1>Printing Table from 2 to 10</h1>
  <table class="center" id="content">
  </table>
  <script type="text/javascript">
    function getRandomColor() {
      var letters = '0123456789ABCDEF';
      var color = '#';
      for (var i = 0; i < 6; i++) {
        color += letters[Math.floor(Math.random() * 16)];
      }
      return color;
    }
  </script>
</body>
</html>
```

```

var tbl = document.getElementById('content');
var number = 2;
var abc = setInterval(printrow,5000);

function printrow(){
  if (number == 10)
  {
    clearInterval(abc);
  }
  var result = "";
  for(var i = 1; i<= 10; i++){
    result = result + "<td>" + number + "*" + i + "=" + number * i+"</td>";
  }
  number++;
  var row = document.createElement('tr');
  row.style.color= getRandomColor();
  row.style.fontSize = (number+10)+"px";
  row.innerHTML=result;
  tbl.append(row);
}
</script>

</body>
</html>

```

OUTPUT:

Printing Table from 2 to 10

2*1=2	2*2=4	2*3=6	2*4=8	2*5=10	2*6=12	2*7=14	2*8=16	2*9=18	2*10=20
3*1=3	3*2=6	3*3=9	3*4=12	3*5=15	3*6=18	3*7=21	3*8=24	3*9=27	3*10=30
4*1=4	4*2=8	4*3=12	4*4=16	4*5=20	4*6=24	4*7=28	4*8=32	4*9=36	4*10=40
5*1=5	5*2=10	5*3=15	5*4=20	5*5=25	5*6=30	5*7=35	5*8=40	5*9=45	5*10=50
6*1=6	6*2=12	6*3=18	6*4=24	6*5=30	6*6=36	6*7=42	6*8=48	6*9=54	6*10=60
7*1=7	7*2=14	7*3=21	7*4=28	7*5=35	7*6=42	7*7=49	7*8=56	7*9=63	7*10=70
8*1=8	8*2=16	8*3=24	8*4=32	8*5=40	8*6=48	8*7=56	8*8=64	8*9=72	8*10=80
9*1=9	9*2=18	9*3=27	9*4=36	9*5=45	9*6=54	9*7=63	9*8=72	9*9=81	9*10=90
10*1=10	10*2=20	10*3=30	10*4=40	10*5=50	10*6=60	10*7=70	10*8=80	10*9=90	10*10=100

Q.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 the paragraph.

Code

Html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grid</title>
  <link rel="stylesheet" href="practical4b.css">
</head>

<body>
  <p> Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga
  optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis
  ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit
  amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea
  distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam
  perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit.
  Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum
  vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a?
  Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime
  cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id
  nulla, vel vero quisquam magni necessitatibus magnam. </p>

  <table>
    <tr>
      <td><button class="btn1">Red</button></td>
      <td class="vertical">
        <button class="btn2">Green</button>
      </td>
      <td><button class="btn3">Blue</button></td>
    </tr>
    <tr>
```

```

<td class="horizontal">
    <button class="btn4">10</button>
</td>
<td class="vertical horizontal">
    <button class="btn5">20</button>
</td>
<td class="horizontal">
    <button class="btn6">30</button>
</td>
</tr>
<tr>
    <td><button class="btn7">Courier New</button></td>
    <td class="vertical">
        <button class="btn8">Franklin Gothic Medium</button>
    </td>
    <td><button class="btn9">Lucida Sans</button></td>
</tr>
</table>
<script src="practical4b.js"></script>
</body>

</html>

```

CSS code

```

td, button {
    width: 100px;
    height: 100px;
    margin: 0 auto;
}

.vertical {
    border-left: 1px solid black;
    border-right: 1px solid black;
}

.horizontal {
    border-top: 1px solid black;
    border-bottom: 1px solid black;
}

table {
    margin: auto;
}

```

```
}
```

Javascript code

```
const button1 = document.querySelector('.btn1');
const button2 = document.querySelector('.btn2');
const button3 = document.querySelector('.btn3');
const button4 = document.querySelector('.btn4');
const button5 = document.querySelector('.btn5');
const button6 = document.querySelector('.btn6');
const button7 = document.querySelector('.btn7');
const button8 = document.querySelector('.btn8');
const button9 = document.querySelector('.btn9');
button1.addEventListener('click', function () {
    document.querySelector('p').style.color = 'red';
})
button2.addEventListener('click', function () {
    document.querySelector('p').style.color = 'green';
})
button3.addEventListener('click', function () {
    document.querySelector('p').style.color = 'blue';
})
button4.addEventListener('click', function () {
    document.querySelector('p').style.fontSize = '10px';
})
button5.addEventListener('click', function () {
    document.querySelector('p').style.fontSize = '20px';
})
button6.addEventListener('click', function () {
    document.querySelector('p').style.fontSize = '30px';
})
button7.addEventListener('click', function () {
    document.querySelector('p').style.fontFamily = 'Courier New', Courier,
monospace;
})
button8.addEventListener('click', function () {
    document.querySelector('p').style.fontFamily = 'Franklin Gothic Medium', 'Arial
Narrow', Arial, sans - serif;
})
button9.addEventListener('click', function () {
    document.querySelector('p').style.fontFamily = 'Lucida Sans', 'Lucida Sans Regular',
'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans - serif;
})
```


Output

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id nulla, vel vero quisquam magni necessitatibus magnam.

Red	Green	Blue
10	20	30
Courier New	Franklin Gothic Medium	Lucida Sans

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id nulla, vel vero quisquam magni necessitatibus magnam.

Red	Green	Blue
10	20	30
Courier New	Franklin Gothic Medium	Lucida Sans

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id nulla, vel vero quisquam magni necessitatibus magnam.

Red	Green	Blue
10	20	30
Courier New	Franklin Gothic Medium	Lucida Sans

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id nulla, vel vero quisquam magni necessitatibus magnam.

Red	Green	Blue
10	20	30
Courier New	Franklin Gothic Medium	Lucida Sans

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id nulla, vel vero quisquam magni necessitatibus magnam.

Red	Green	Blue
10	20	30
Courier New	Franklin Gothic Medium	Lucida Sans

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus impedit fuga optio. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minima facilis adipisci quis laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspiciatis vel exercitationem pariatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliquid. Ex ullam natus autem vitae, dignissimos iure id nulla, vel vero quisquam magni necessitatibus magnam.

Red	Green	Blue
10	20	30
Courier New	Franklin Gothic Medium	Lucida Sans

Q.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.

CODE:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Practical 6</title>
  <style type="text/css">
    .container{
      width: 60%;
      margin: auto;
      border: 1px solid black;
      border-radius: 8px;
      padding: 50px;

    }

    .btn-submit{
      border-radius: 5px;
      color: white;
      background: greenyellow;
      font-weight: bold;
      font-size: 1rem;
      margin: 20px;
    }
    @media(width<=575){
      .container{
        width: 84%;
      }
    }
  </style>
</head>
<body>
  <div class="container">
    <h1>Pet's Information</h1>
    <hr>
    <label for="name">Pet's Name: </label>
    <input type="text" name="name"><br><br>
    <label for="age">Age: </label>
```

```

<input type="number" name="age">
<label for="weight">Weight: </label>
<input type="number" name="weight" class=""><br><br>
<label for="type">Pet type: </label>
<input type="text" name="type"><br><br>
<label for="likes">Likes: </label>
<input type="text" name="likes"><br>
<button class="btn-submit" onclick="display()">Submit</button>
</div>

<script type="text/javascript">

function display(){
    var pet = {};
    var input_fields = document.getElementsByTagName('input');
    for (var i = 0; i < input_fields.length; i++) {
        pet[input_fields[i].name] = input_fields[i].value;
    }
    console.log(pet);
}

</script>

</body>
</html>

```

OUTPUT:

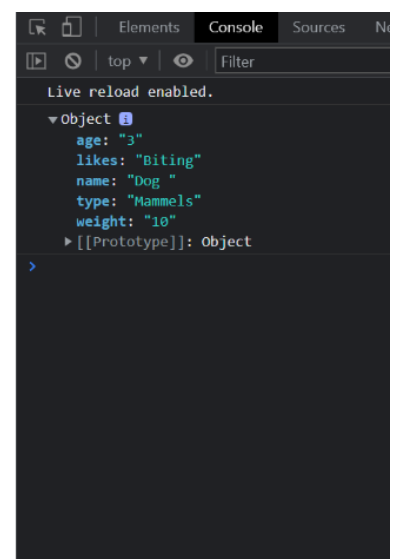
Pet's Information

Pet's Name:

Age: Weight:

Pet type:

Likes:



Q.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" />
  <title>Practical 7</title>
  <style type="text/css">
    #pet-data{
      border: 1px solid black;
      border-radius: 10px;
      border-collapse: collapse;
    }
    td{
      border: 1px solid black;
      border-collapse: collapse;
    }
    #btn-fetch{
      margin-top: 20px;
      font-size: 24px;
      font-weight: bold;
      background-color: black;
      color: white;
      border-radius: 8px;
    }
  </style>
</head>
<body>
  <div id="content">

  </div>

  <button id="btn-fetch">Fetch Data</button>
```

```

    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
</script>
    <script type="text/javascript">
        var btnFetch = document.getElementById('btn-fetch');
        var content = document.getElementById('content');

        btnFetch.addEventListener('click', ()=>{
            const xhr = new XMLHttpRequest();
            xhr.open("GET", "./pet.json", true);
            xhr.onload = ()=>{
                console.log(xhr.responseText);
                renderHtml (JSON.parse(xhr.responseText));
            }
            xhr.send();
        });

        function renderHtml(data){
            content.innerHTML = "";
            for (var i = 0; i <= data.length; i++) {
                let p = document.createElement('p');
                let htmlpart = "";
                htmlpart += data[i].name+" is a "+data[i].type+" with
age "+data[i].age+" years and weight "+data[i].weight+"kg and likes
"+data[i].likes;

                p.innerHTML = htmlpart;
                content.append(p);
                // console.log(tr);
                htmlpart="";
            }
        }
    </script>
</body>
</html>

```

JSON

```

[
  {
    "name": "Pluto",
    "age": 3,
    "weight": 12,
    "type": "Pavellion",
    "likes": "eating, playing with ball"
  }
]

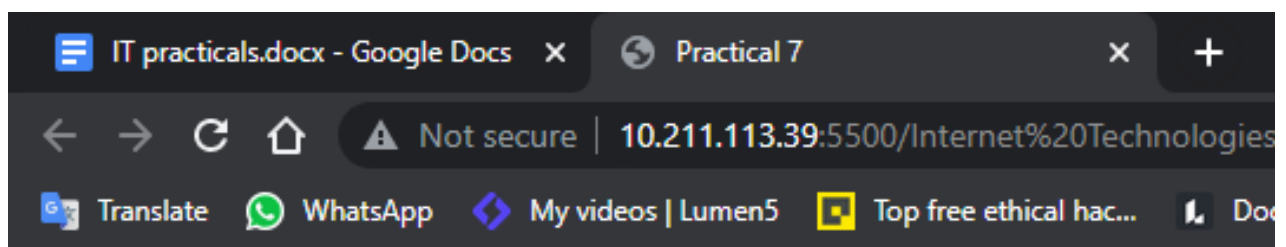
```

```

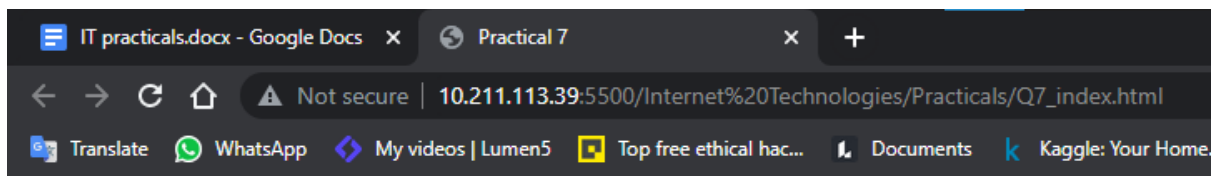
},
{
  "name": "Hulk",
  "age": 4,
  "weight": 22,
  "type": "German Sefford",
  "likes": "Biting, eating flesh"
},
{
  "name": "Jerry",
  "age": 2.5,
  "weight": 8,
  "type": "cat",
  "likes": "sleeping"
},
{
  "name": "Tom",
  "age": 0.5,
  "weight": 0.7,
  "type": "Mouse",
  "likes": "running, eating cheese"
},
{
  "name": "Chiku",
  "age": 1,
  "weight": 1.2,
  "type": "Rabbit",
  "likes": "running, eating carrot"
}
]

```

Output:



Fetch Data



Pluto is a Pavellion with age 3 years and weight 12kg and likes eating, playing with ball

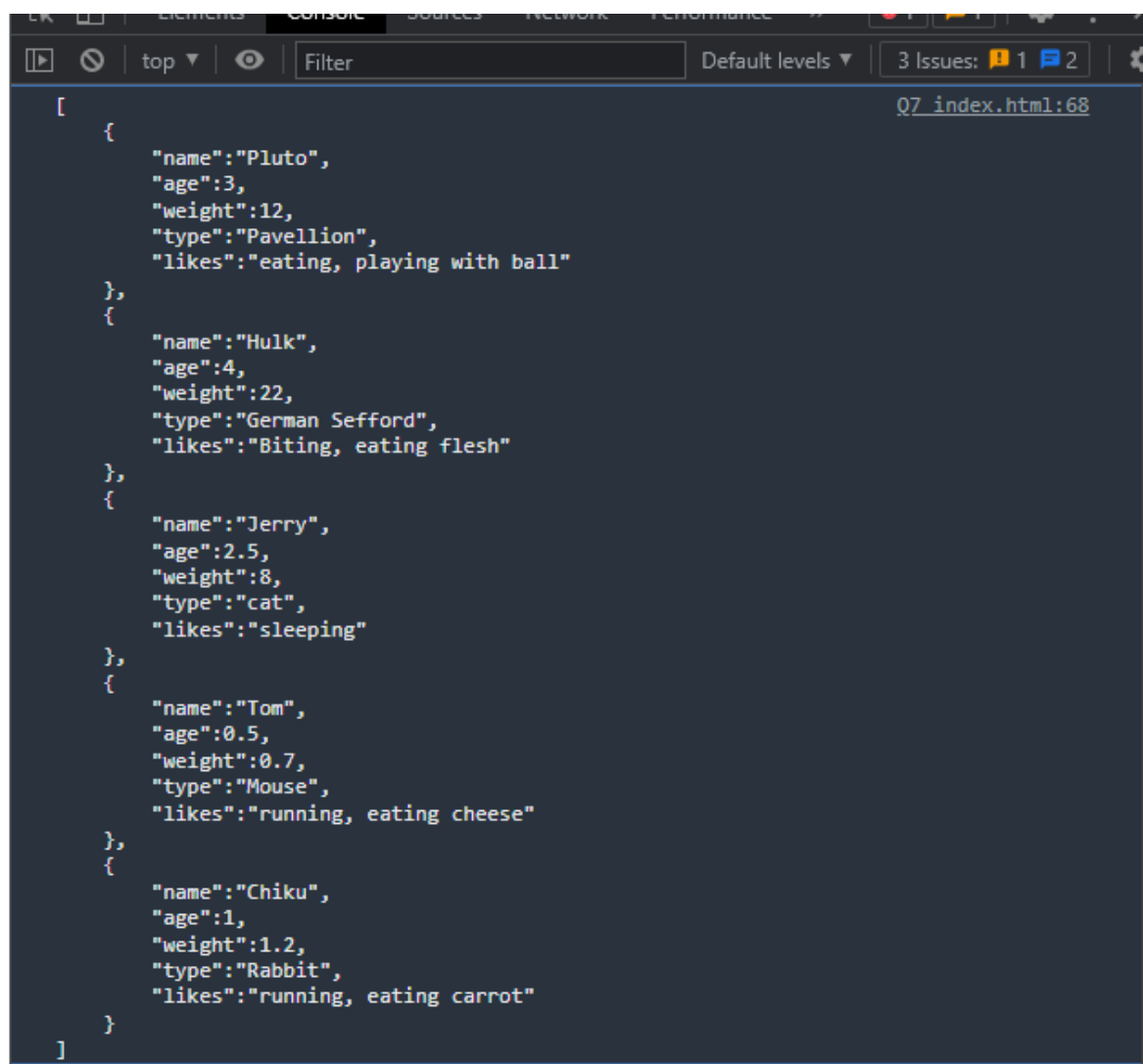
Hulk is a German Sefford with age 4 years and weight 22kg and likes Biting, eating flesh

Jerry is a cat with age 2.5 years and weight 8kg and likes sleeping

Tom is a Mouse with age 0.5 years and weight 0.7kg and likes running, eating cheese

Chiku is a Rabbit with age 1 years and weight 1.2kg and likes running, eating carrot

Fetch Data



Q.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 the 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>

<head>
  <meta charset="utf-8">
  <title>Practical 8</title>
  <style type="text/css">
    .container {
      width: 70%;
      margin: auto;
      align-items: center;
      background-color: #D9CAB3;
      padding-bottom: 10px;
    }

    .info-table {
      width: 80%;
      margin: auto;
      border: 3px solid black;
      border-collapse: collapse;
      margin-top: 2%;
      margin-bottom: 2%;
    }

    .table-row {
      width: 100%;
      margin: auto;
    }
  </style>
</head>

<body>
  <div class="container">
    <table border="1" class="info-table">
      <tr>
        <th>Sl. No.</th>
        <th>Name</th>
        <th>Credits</th>
        <th>Fee</th>
      </tr>
      <tr>
        <td>1</td>
        <td>John Doe</td>
        <td>3</td>
        <td>1000</td>
      </tr>
      <tr>
        <td>2</td>
        <td>Jane Smith</td>
        <td>4</td>
        <td>1200</td>
      </tr>
      <tr>
        <td>3</td>
        <td>Mike Johnson</td>
        <td>2</td>
        <td>800</td>
      </tr>
    </table>
    <div class="table-row">
      <div>Future possibilities</div>
      <div>After the course, students can pursue a Master's degree in Computer Science or a job in the IT industry.</div>
    </div>
  </div>
  <div class="container">
    <button class="styling-button">Click here to style the page</button>
  </div>
</body>
</html>
```



```

        <td>4</td>
    </tr>
</table>
</td>
</tr>
<tr>
    <td>Future Opportunities</td>
    <td>infinite Opportunities but its totally depend on you and your skill</td>
</tr>
</table>
</div>

<button id="btn-style">
    Style Page
</button>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
</script>

<script type="text/javascript">
    $(document).ready(function () {
        $('#btn-style').click(function () {
            $("div").addClass('container');
            $("table").addClass('info-table');
            $("tr").addClass('table-row');
            $("td").addClass('table-data');
            $(".heading").css({
                "textAlign": 'center'
            });
        });
    });

</script>

</body>

</html>

```

Output:

Bsc Hons Computer Science

Fee 33000

Eligibility Criteria 12th Pass

	Subject	Credit score
Subjects and credit scores	IT	6
	Toc	6
	DAV/System Programming	4
	DIP/Micro	4

Future Opportunities infinite Opportunities but its totally depend on you and your skill

[Style Page](#)

Bsc Hons Computer Science

Fee	33000										
Eligibility Criteria	12th Pass										
Subjects and credit scores	<table><tr><th>Subject</th><th>Credit score</th></tr><tr><td>IT</td><td>6</td></tr><tr><td>Toc</td><td>6</td></tr><tr><td>DAV/System Programming</td><td>4</td></tr><tr><td>DIP/Micro</td><td>4</td></tr></table>	Subject	Credit score	IT	6	Toc	6	DAV/System Programming	4	DIP/Micro	4
Subject	Credit score										
IT	6										
Toc	6										
DAV/System Programming	4										
DIP/Micro	4										
Future Opportunities	infinite Opportunities but its totally depend on you and your skill										

Q.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>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link
    rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <title>Practical 9</title>
  <style type="text/css">
    /*img{
      margin: 20px;
    }*/
  </style>
</head>
<body>

  <div class="jumbotron text-center">
    <h1>IMAGE GALLERY</h1>
    <p>Responsive Image gallery using bootstrap.</p>
  </div>
  <div class="container">
    </img>
    </img>
    </img>
    </img>
    </img>
    </img>
  </div>

  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">
</script>
  <script
    src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js">
  </script>

</body>
</html>
```

Output:

IMAGE GALLERY

Responsive Image gallery using bootstrap.



Q.10. Create an HTTP server using Node.js which handles requests on port 10000

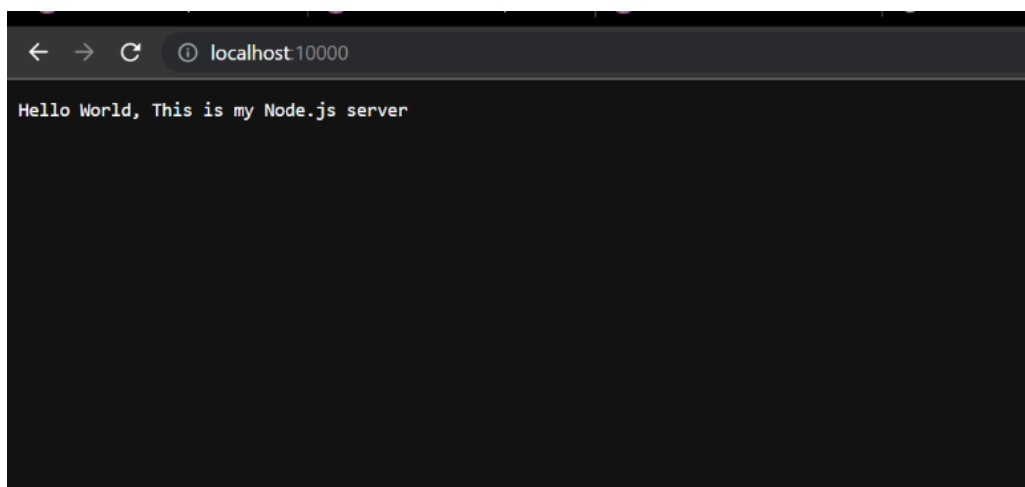
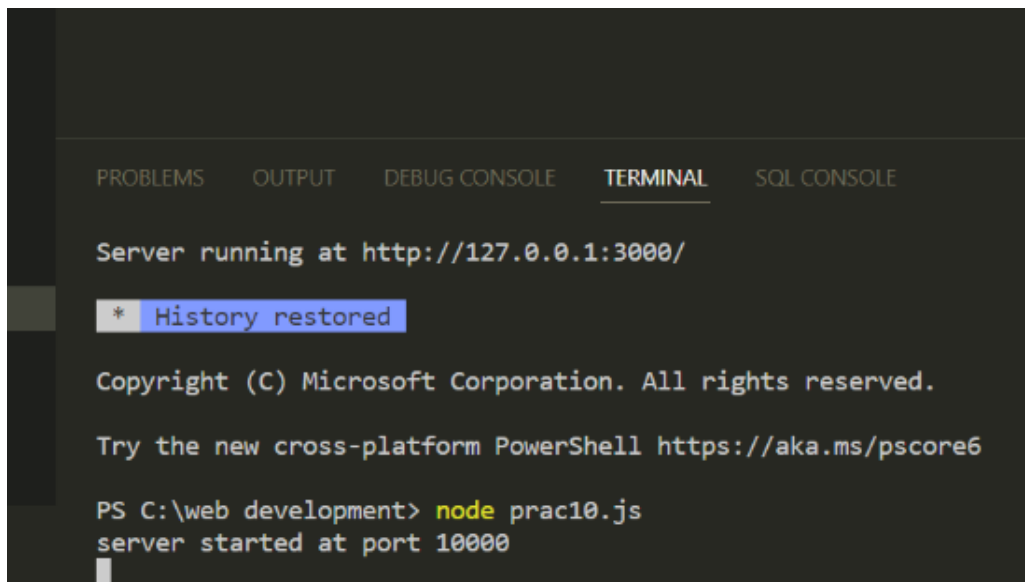
or a free port beyond 10000. Modify the server in such a way that opening localhost:10000 will display “Hello world, This is my Node.js server” on the browser.

Code

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

http.createServer((request,response)=>{
  response.write("Hello World, This is my Node.js server");
  response.end();
})
.listen(10000,()=>console.log('server started at port 10000'));
```

Output



Q.11. Create index.html file containing two forms for SignIn and SignUp.

Submitting SignIn form should search for credentials in mysql database using a server created in previous practical. On successful signin, a welcome page should be displayed. Submitting the SignUp form should insert a new entry for credentials in the mysql database using the server created in previous practical. On successful sign up, the user should be returned back to index.html.

Code:

```
<!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>SignIn-SignUp Form</title>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <section class="container">
      <h2>SignIn Form</h2>
      <form class="form">
        <div>
          <label for="username">Username:</label>
          <input type="text" id="username" placeholder="Enter username" />
        </div>
        <div>
          <label for="password">Password:</label>
          <input type="password" id="password" placeholder="Enter password" />
        </div>
        <button class="submit-btn">Submit</button>
      </form>
    </section>

    <section class="container">
      <h2>SignUp Form</h2>
      <form class="form">
```



```

<div>
  <label for="name">Name:</label>
  <input type="text" id="name" placeholder="Enter name" />
</div>
<div>
  <label for="email">E-mail:</label>
  <input type="email" id="email" placeholder="Enter e-mail" />
</div>
<div>
  <label for="new-username">Username:</label>
  <input
    type="text"
    class="new-username"
    placeholder="Create username"
  />
</div>
<div>
  <label for="new-password">Password:</label>
  <input
    type="password"
    id="new-password"
    placeholder="Enter password"
  />
</div>
<div>
  <label for="re-password">Password (confirm):</label>
  <input
    type="password"
    id="re-password"
    placeholder="Enter password again"
  />
</div>
<button class="submit-btn">Submit</button>
</form>
</section>
</body>
</html>

```

CSS:

```

/*
SPACING SYSTEM (px)
2 / 4 / 8 / 12 / 16 / 24 / 32 / 48 / 64 / 80 / 96 / 128

```

```
FONT SIZE SYSTEM (px)
10 / 12 / 14 / 16 / 18 / 20 / 24 / 30 / 36 / 44 / 52 / 62 / 74 / 86 / 98
*/

/*
MAIN COLOR: #099268
GREY COLOR: #087f5b
ACCENT: #20c997, #e6fcf5
*/
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-family: sans-serif;
  width: 850px;
  margin: 40px auto;
  display: grid;
  grid-template-columns: repeat(2, 1fr);
  column-gap: 40px;
  align-items: center;
}

.container {
  background-color: #e6fcf5;
  padding: 40px;
  border-radius: 30px;
  box-shadow: 0 20px 30px 0 rgb(0, 0, 0, 0.07);
}

.form {
  display: grid;
  grid-template-columns: 1fr;
  justify-items: start;
  align-items: center;
  row-gap: 30px;
}

h2 {
  background-color: #099268;
  display: inline-block;
```

```
padding: 20px;
border-radius: 20px;
color: #e6fcf5;
margin-bottom: 52px;
word-spacing: 2px;
letter-spacing: 0.5px;
}

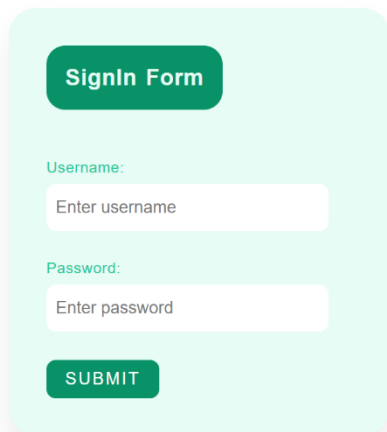
.submit-btn {
font-size: 18px;
text-transform: uppercase;
background-color: #099268;
border: none;
color: #fff;
padding: 10px 20px;
border-radius: 10px;
align-self: end;
letter-spacing: 2px;
}

.submit-btn:hover {
background-color: #087f5b;
cursor: pointer;
}

label {
display: inline-block;
margin-bottom: 8px;
color: #20c997;
letter-spacing: 0.5px;
font-size: 16px;
}

.form input {
padding: 10px;
border-radius: 10px;
width: 300px;
height: 50px;
border: none;
font-size: 18px;
}
```

Output:



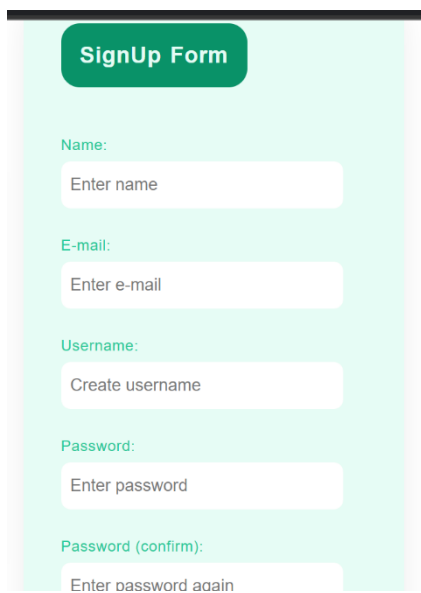
A light green rounded rectangle with a dark green header bar containing the text "SignIn Form". Below the header, there are two input fields: "Username:" followed by a white box with "Enter username", and "Password:" followed by a white box with "Enter password". At the bottom is a dark green button with the text "SUBMIT".

SignIn Form

Username:
Enter username

Password:
Enter password

SUBMIT



A light green rounded rectangle with a dark green header bar containing the text "SignUp Form". Below the header, there are five input fields: "Name:" followed by a white box with "Enter name", "E-mail:" followed by a white box with "Enter e-mail", "Username:" followed by a white box with "Create username", "Password:" followed by a white box with "Enter password", and "Password (confirm):" followed by a white box with "Enter password again".

SignUp Form

Name:
Enter name

E-mail:
Enter e-mail

Username:
Create username

Password:
Enter password

Password (confirm):
Enter password again

Password:

Enter password

Password (confirm):

Enter password again

SUBMIT

The End