



RAMANUJAN COLLEGE
(UNIVERSITY OF DELHI)

SUBJECT-INTERNET TECHNOLOGIES
(PRACTICAL FILE)

NAME=MOHD SHOHEL

COLLEGE ROLL NO. = 20201417

EXAMINATION ROLL NO. = 20020570022

SUBMITTED TO = DR.KAMLESH KUMAR RAGHUVANSHI

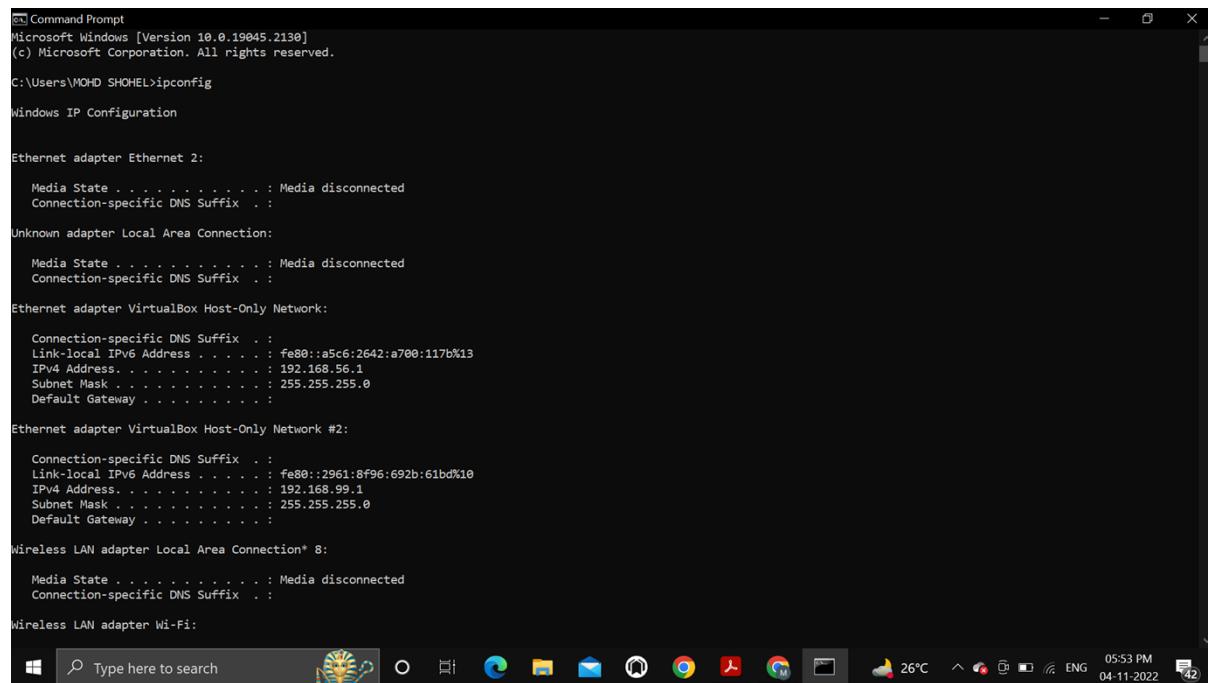
SEMESTER = 5th

Practicals:

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.

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.

OUTPUTS:



```
Microsoft Windows [Version 10.0.19045.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\MOHD SHOHEL>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet 2:

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

Unknown adapter Local Area Connection:

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

Ethernet adapter VirtualBox Host-Only Network:

    Connection-specific DNS Suffix . :
    Link-local IPv6 Address . . . . . : fe80::a5c6:2642:a700:117b%13
    IPv4 Address . . . . . : 192.168.56.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Ethernet adapter VirtualBox Host-Only Network #2:

    Connection-specific DNS Suffix . :
    Link-local IPv6 Address . . . . . : fe80::2961:8f96:692b:61bd%10
    IPv4 Address . . . . . : 192.168.99.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Wireless LAN adapter Local Area Connection#8:

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

Wireless LAN adapter Wi-Fi:
```

```
C:\ Command Prompt
IPv4 Address . . . . . : 192.168.99.1
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :

Wireless LAN adapter Local Area Connection* 8:

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

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . . . . . :
Link-local IPv6 Address . . . . . : fe80::78ca:562a:908e:bcbe%16
IPv4 Address . . . . . : 192.168.29.90
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.29.1

Ethernet adapter Bluetooth Network Connection:

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

C:\Users\MOHD SHOHEL>
```

```
Windows Type here to search 0 Type here to search
C:\ Command Prompt
```

```
C:\Users\MOHD SHOHEL>ping google.com

Pinging google.com [142.250.192.78] with 32 bytes of data:
Reply from 142.250.192.78: bytes=32 time=35ms TTL=52
Reply from 142.250.192.78: bytes=32 time=37ms TTL=52
Reply from 142.250.192.78: bytes=32 time=32ms TTL=52
Reply from 142.250.192.78: bytes=32 time=561ms TTL=52

Ping statistics for 142.250.192.78:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 32ms, Maximum = 561ms, Average = 166ms

C:\Users\MOHD SHOHEL>
```

```
Windows Type here to search 0 Type here to search
C:\ Command Prompt
```

C:\ Command Prompt
Microsoft Windows [Version 10.0.19045.2130]
(c) Microsoft Corporation. All rights reserved.
C:\Users\MOHD SHOHEL>tracert canva.com
Tracing route to canva.com [104.17.114.17]
over a maximum of 30 hops:

 1 4 ms 2 ms 4 ms reliance.reliance [192.168.29.1]
 2 38 ms 5 ms 24 ms 10.0.72.1
 3 91 ms 8 ms 9 ms 172.16.18.29
 4 49 ms 13 ms 7 ms 192.168.96.234
 5 5 ms 7 ms 8 ms 172.26.111.117
 6 12 ms 9 ms 10 ms 172.26.111.131
 7 10 ms 8 ms 6 ms 192.168.44.44
 8 * * ms * ms Request timed out.
 9 * * ms * ms Request timed out.
10 * * ms * ms Request timed out.
11 23 ms 7 ms 24 ms 49.44.220.129
12 11 ms 12 ms 7 ms 104.17.114.17

Trace complete.
C:\Users\MOHD SHOHEL>netstat -r
=====
Interface List
22...00 ff e8 ef 75 8bExpressVPN TAP Adapter
14...00 ff 8a a3 b2 adTAP-Windows Adapter V9
13...0a 00 27 00 00 0dVirtualBox Host-Only Ethernet Adapter
10...0a 00 27 00 00 0aVirtualBox Host-Only Ethernet Adapter #2
26...70 cf 49 ee c6 4bMicrosoft Wi-Fi Direct Virtual Adapter
7...72 cf 49 ee c6 4aMicrosoft Wi-Fi Direct Virtual Adapter #2
16...70 cf 49 ee c6 4aIntel(R) Wi-Fi 6 AX201 160MHz
19...70 cf 49 ee c6 4eBluetooth Device (Personal Area Network)
1.....00 00 00 00 00 00Software Loopback Interface 1
=====
IPv4 Route Table
=====
Active Routes:
Network Destination Netmask Gateway Interface Metric
C:\ Command Prompt
C:\Users\MOHD SHOHEL>netstat -a
Active Connections

Proto Local Address Foreign Address State
TCP 0.0.0.0:135 ShoheL:0 LISTENING
TCP 0.0.0.0:445 ShoheL:0 LISTENING
TCP 0.0.0.0:3306 ShoheL:0 LISTENING
TCP 0.0.0.0:5840 ShoheL:0 LISTENING
TCP 0.0.0.0:5357 ShoheL:0 LISTENING
TCP 0.0.0.0:6646 ShoheL:0 LISTENING
TCP 0.0.0.0:33060 ShoheL:0 LISTENING
TCP 0.0.0.0:49664 ShoheL:0 LISTENING
TCP 0.0.0.0:49665 ShoheL:0 LISTENING
TCP 0.0.0.0:49666 ShoheL:0 LISTENING
TCP 0.0.0.0:49667 ShoheL:0 LISTENING
TCP 0.0.0.0:49668 ShoheL:0 LISTENING
TCP 0.0.0.0:49674 ShoheL:0 LISTENING
TCP 127.0.0.1:2815 ShoheL:0 LISTENING
TCP 127.0.0.1:15292 ShoheL:0 LISTENING
TCP 127.0.0.1:24642 ShoheL:0 LISTENING
TCP 127.0.0.1:27017 ShoheL:0 LISTENING
TCP 127.0.0.1:45623 ShoheL:0 LISTENING
TCP 127.0.0.1:49670 kubernetes:49671 ESTABLISHED
TCP 127.0.0.1:49671 kubernetes:49670 ESTABLISHED
TCP 127.0.0.1:49672 kubernetes:49673 ESTABLISHED
TCP 127.0.0.1:49673 kubernetes:49672 ESTABLISHED
TCP 127.0.0.1:49716 kubernetes:49717 ESTABLISHED
TCP 127.0.0.1:49717 kubernetes:49716 ESTABLISHED
TCP 127.0.0.1:49721 kubernetes:49722 ESTABLISHED
TCP 127.0.0.1:49722 kubernetes:49721 ESTABLISHED
TCP 127.0.0.1:49934 kubernetes:49935 ESTABLISHED
TCP 127.0.0.1:49935 kubernetes:49934 ESTABLISHED
TCP 127.0.0.1:53783 ShoheL:0 LISTENING
TCP 127.0.0.1:53783 kubernetes:53785 ESTABLISHED
TCP 127.0.0.1:53785 kubernetes:53783 ESTABLISHED
TCP 127.0.0.1:53892 ShoheL:0 LISTENING
TCP 127.0.0.1:53893 ShoheL:0 LISTENING
TCP 127.0.0.1:55624 ShoheL:0 LISTENING
TCP 192.168.29.90:139 ShoheL:0 LISTENING
C:\Users\MOHD SHOHEL>


```
C:\ Command Prompt
UDP 192.168.56.1:137    *;*
UDP 192.168.56.1:138    *;*
UDP 192.168.56.1:1900   *;*
UDP 192.168.56.1:60242  *;*
UDP 192.168.73.134:62662 *;*
UDP 192.168.99.1:137    *;*
UDP 192.168.99.1:138    *;*
UDP 192.168.99.1:1900   *;*
UDP 192.168.99.1:49665  *;*
UDP [::]:500    *;*
UDP [::]:3762   *;*
UDP [::]:3762   *;*
UDP [::]:4500   *;*
UDP [::]:5353   *;*
UDP [::]:5355   *;*
UDP [::]:56272  *;*
UDP [::]:1900   *;*
UDP [::]:64957  *;*
UDP [fe80::2961:8f96.692b:61bd%10]:1900 *;*
UDP [fe80::2961:8f96.692b:61bd%10]:64955 *;*
UDP [fe80::78ca:562a:908e:bcb6%16]:1900 *;*
UDP [fe80::78ca:562a:908e:bcb6%16]:64956 *;*
UDP [fe80::a5c6:2642:a700:117b%13]:1900 *;*
UDP [fe80::a5c6:2642:a700:117b%13]:64954 *;*

C:\Users\MOHD SHOHEL>ping 25.56.227.210

Pinging 25.56.227.210 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 25.56.227.210:
 Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Windows Search icon  Type here to search  O  Microsoft Edge  File  Mail  Hub  Google Chrome  Microsoft Edge  26°C  ENG  06:14 PM  04-11-2022 42

C:\ Command Prompt
UDP [::]:5353    *;*
UDP [::]:5355    *;*
UDP [::]:56272   *;*
UDP [::]:1900   *;*
UDP [::]:64957  *;*
UDP [fe80::2961:8f96.692b:61bd%10]:1900 *;*
UDP [fe80::2961:8f96.692b:61bd%10]:64955 *;*
UDP [fe80::78ca:562a:908e:bcb6%16]:1900 *;*
UDP [fe80::78ca:562a:908e:bcb6%16]:64956 *;*
UDP [fe80::a5c6:2642:a700:117b%13]:1900 *;*
UDP [fe80::a5c6:2642:a700:117b%13]:64954 *;*

C:\Users\MOHD SHOHEL>ping 25.56.227.210

Pinging 25.56.227.210 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 25.56.227.210:
 Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Windows Search icon  Type here to search  O  Microsoft Edge  File  Mail  Hub  Google Chrome  Microsoft Edge  26°C  ENG  06:07 PM  04-11-2022 42

C:\Users\MOHD SHOHEL>ping 192.168.43.19

Pinging 192.168.43.19 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.43.19:
 Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

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.

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>About me</title>
    <link rel="stylesheet" href="./main.css" />
  </head>

  <body>
    <header class="tophead">
      <p class="tophead-intro">Hello Everyone this is</p>
      <h1 class="tophead-heading">Mohd Shohel</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>
      <p>I love play Badminton ,listening to music, watching anime, web series and reading books.</p>
    </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++</li>
            <li>Java </li>
            <li>DSA</li>
            <li>Python</li>
            <li>Computer Networks</li>
          </ul>
        </div>
      </p>
    </section>

    <section class="plan-section">
      <h1>My Plans</h1>
```

```
<p>FULL STACK DEVELOPER</p>
</section>

<footer class="content-footer">
  <p>Follow for more updates:</p>
  <ul class="social">
    <li>
      <a class="css-is-deranged" href="#">LinkedIn</a>
    </li>
    <li>
      <a class="css-is-deranged" href="#">Instagram</a>
    </li>
    <li>
      <a class="css-is-deranged" href="#">Twitter</a>
    </li>
  </ul>
</footer>
</body>
</html>
```

CSS:

main.css

```
body {
  color: #222222;
  font-size: 1em;
  font-family: "Open Sans", "Helvetica Neue", sans-serif;
}

.content-footer,
.tophead-heading,
.tophead-intro {
  text-align: center;
}

.tophead {
  padding: 6em 0;
  border-top: solid 1em #ffcf33;
}
```

```
.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;  
}  
  
.tophead {  
    padding: 6em 0;  
    background-size: cover;
```

```
background-repeat: no-repeat;
border-top: solid 1em #ffcf33;
}

.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-color: #73a8c2;
text-decoration: none;
}
.social > li {
```

```
    display: inline-block;
    margin: 0 5px;
}
.content-footer > p {
    color: #f7ede1;
}
a {
    font-weight: bold;
    text-decoration: none;
    color: #f1c863;
}

@media only screen and (max-width: 500px) {
    .tophead {
        padding: 3em 0;
    }
    .tophead-heading {
        font-size: 3em;
    }
    .content-footer {
        padding: 2em 2.5em;
    }
}
```

OUTPUT:

Hello Everyone this is
MOHD SHOHEL

About me

I'm a final year student of Bsc (Hons.) Computer Science at Ramanujan College, University of Delhi.

My Hobbies

I love play Badminton ,listening to music, watching anime, web series and reading books.

My Course

Things that I have learnt so far:

- C++
- Java
- DSA
- Python
- Computer Networks

My Plans

FULL STACK DEVELOPER

Follow for more updates:

[Linkedin](#) [Instagram](#) [Twitter](#)

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 :

index.html

```
<!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>

```

```

</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;
    }

    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

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

index.html

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

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

practical4b.css

```

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;
}
```

practical4b.js

```
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

Q5: 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: HTL

10. *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 quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minimus facilis adipisci qui laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo, perspicatis vel exercitationem paratur est repellendus amet, cupiditate illius aspernatur autem reuteri iure quis, error itaque dolore molestiae qui quo dicitur? Lorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum aliqui 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

Q5: 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 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:

Quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ad distinctio omnis, quod quasi minima facilis adipisci laboriosam ab quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ilo, perspicillati vel exercitationem paratur est repellendus amet, cupiditate aspernatur etiam iure qui, error itaque dolore molestiae qui quo dicit? Aorem ipsum dolor sit amet consectetur adipisicing elit. Aut delectus maxime cupiditate consectetur rerum et ullam natus autem vitae, dignissimos iure id nulla, vel vero omnisquam maiores necessitatibus magnam.

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

Q5: 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:

... ipsum accusantium modi enim eos ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipisicing elit. Officia eum esse suscipit cumque natus repellat ea distinctio omnis, quod quasi minimus facilis adipisci quis laboriosam at quisquam perferendis error! Libero? Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ilo, perspicillat vel exercitationem paratur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quo id? Lorent 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

Q5: 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:

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

Q5: 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:

Consectetur adipiscing elit. Eveniet eligendi saepe ad inventore, magnam repellendus maiores nobis ipsum accusantium modi enim eos quasi ex fugit. Laboriosam! Lorem ipsum dolor sit amet consectetur adipiscing 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 adipiscing elit. Illo, perspiciatibus vel exercitationem parvatur est repellendus amet, cupiditate illum vitae aspernatur reprehenderit iure quis, error itaque dolore molestiae qui quod a? Lorem ipsum dolor sit amet consectetur adipiscing 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

Q5: 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:

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, perspiciatibus 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

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:

index.html

```
<!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;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Welcome to Practical 6</h1>
        <p>This practical will help you practice basic CSS skills.</p>
        <input type="button" value="Submit" class="btn-submit" />
    </div>
</body>
</html>
```

```
@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(){
            // event.preventDefault();
            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>
```

The screenshot shows a web browser window. On the left, there is a form titled "Pet's Information" with fields for Pet Name (Dog), Age (3), Weight (10), Pet type (Mammels), Likes (Biting), and a Submit button. On the right, the browser's developer tools are open, specifically the "Console" tab, showing the JSON object that was submitted:

```

Object {
  age: "3",
  likes: "Biting",
  name: "Dog",
  type: "Mammels",
  weight: "10"
}

```

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 :

index.html

```

<!DOCTYPE html>
<html>

<head>
    <title>Read data from External JSON file using JavaScript</title>

    <style>
        body {
            background-color: lavender;
        }

        h1 {
            text-align: center;
            color: brown;
        }

        div {
            font: 17px 'Calibri';
        }

        table,
        th,
        td {
            border: solid 2px black;
            background-color: beige;
            border-collapse: collapse;
            padding: 2px 3px;
        }
    </style>
</head>
<body>
    <h1>Read data from External JSON file using JavaScript</h1>
    <div>
        <table border="1">
            <thead>
                <tr>
                    <th>Name</th>
                    <th>Age</th>
                    <th>Type</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td>Dog</td>
                    <td>3</td>
                    <td>Mammels</td>
                </tr>
                <tr>
                    <td>Cat</td>
                    <td>5</td>
                    <td>Mammels</td>
                </tr>
                <tr>
                    <td>Bird</td>
                    <td>2</td>
                    <td>Avian</td>
                </tr>
            </tbody>
        </table>
    </div>
</body>

```

```

        text-align: center;
        font: 200 20px Times New Roman;
        color: navy;
        margin: auto;
    }

    th {
        font-weight: bold;
        color: black;
    }

```

</style>

</head>

<body>

PET DETAILS

</h1>

<div id='showTable'></div>

<script>

```

// Create XMLHttpRequest object.
var oXHR = new XMLHttpRequest();
// Initiate request.
oXHR.onreadystatechange = reportStatus;
oXHR.open("GET", "data.json", true); // get json file.
oXHR.send();
function reportStatus() {
    if (oXHR.readyState == 4) // Check if request is complete.
    {
        // Create an HTML table using a response from the server.
        displayData(this.responseText);
    }
}
// Create an HTML table using the JSON data.
function displayData(jsonData) {
    var arrData = [];
    arrData = JSON.parse(jsonData); // Convert JSON to array.
    var col = [];
    for (var i = 0; i < arrData.length; i++) {
        for (var key in arrData[i]) {
            if (col.indexOf(key) === -1) {
                col.push(key);
            }
        }
    }
    // Create a dynamic table.
    var table = document.createElement("table");
    // Create table header.
    var tr = table.insertRow(-1); // Table row.

```

```

        for (var i = 0; i < col.length; i++) {
            var th = document.createElement("th");           // Table header.
            th.innerHTML = col[i];
            tr.appendChild(th);
        }
        // Add JSON to the table rows.
        for (var i = 0; i < arrData.length; i++) {
            tr = table.insertRow(-1);
            for (var j = 0; j < col.length; j++) {
                var tabCell = tr.insertCell(-1);
                tabCell.innerHTML = arrData[i][col[j]];
            }
        }
        // Finally, add the dynamic table to a container.
        var divContainer = document.getElementById("showTable");
        divContainer.innerHTML = "";
        divContainer.appendChild(table);
    };
</script>

</body>

</html>

```

data.json

```
[
    {
        "Name": "Tommy",
        "Age": "2 months",
        "Weight": "5 kg",
        "Type": "Dog",
        "Likes": "Playing and Eating"
    },
    {
        "Name": "Jenny",
        "Age": "3 months",
        "Weight": "3 kg",
        "Type": "Cat",
        "Likes": "Playing"
    },
    {
        "Name": "Bunny",
        "Age": "5 months",
        "Weight": "4 kg",
        "Type": "Rabbit",
    }
]
```

```

    "Likes": "Carrots"
  },
  {
    "Name": "Goldy",
    "Age": "1 month",
    "Weight": "1 kg",
    "Type": "Fish",
    "Likes": "Swimming"
  }
]

```

Output:

| PET DETAILS | | | | |
|-------------|----------|--------|--------|--------------------|
| Name | Age | Weight | Type | Likes |
| Tommy | 2 months | 5 kg | Dog | Playing and Eating |
| Jenny | 3 months | 3 kg | Cat | Playing |
| Bunny | 5 months | 4 kg | Rabbit | Carrots |
| Goldy | 1 month | 1 kg | Fish | Swimming |

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:

index.html

```

<!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;
}

.table-data {
    width: 50%;
    border: 2px solid white;
    border-collapse: collapse;
}
</style>
</head>

<body>
<div>
    <h1 class="heading">Bsc Hons Computer Science</h1>
    <table>
        <tr>
            <td>Fee</td>
            <td>33000</td>
        </tr>
        <tr>
            <td>Eligibility Criteria</td>
            <td>12th Pass</td>
        </tr>
        <tr>
            <td>Subjects and credit scores</td>
            <td>
                <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>
    </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
	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](#)

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:

index.html

```
<!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>

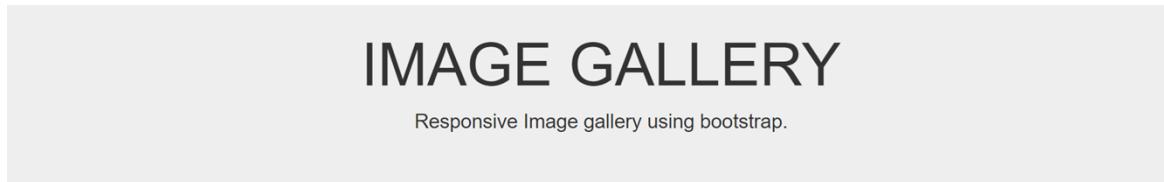
    <div class="jumbotron text-center">
        <h1>IMAGE GALLERY</h1>
        <p>Responsive Image gallery using bootstrap.</p>
    </div>
    <div class="container">
        </img>
        </img>
        </img>
        </img>
        </img>
        </img>
    </div>

    <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:





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

practical10.js

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

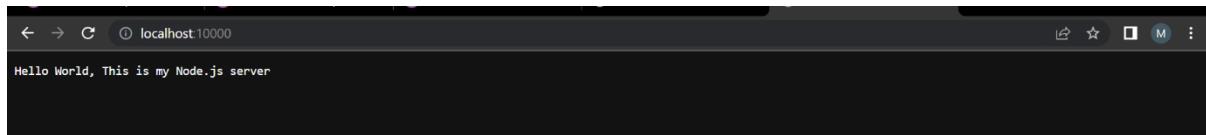
http.createServer((request,response)=>{
    response.write("Hello World, This is my Node.js server");
    response.end();
})
.listen(10000, ()=>console.log('server started at port 10000'));
```

Output

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE
Server running at http://127.0.0.1:3000/
* History restored
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\web development> node prac10.js
server started at port 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.

Code:

Index.html

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

style.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:

SignIn Form

Username:

Enter username

Password:

Enter password

SUBMIT

SignUp Form

Name:

Enter name

E-mail:

Enter e-mail

Username:

Create username

Password:

Enter password

Password:

Enter password

Password (confirm):

Enter password again

SUBMIT

THANK YOU