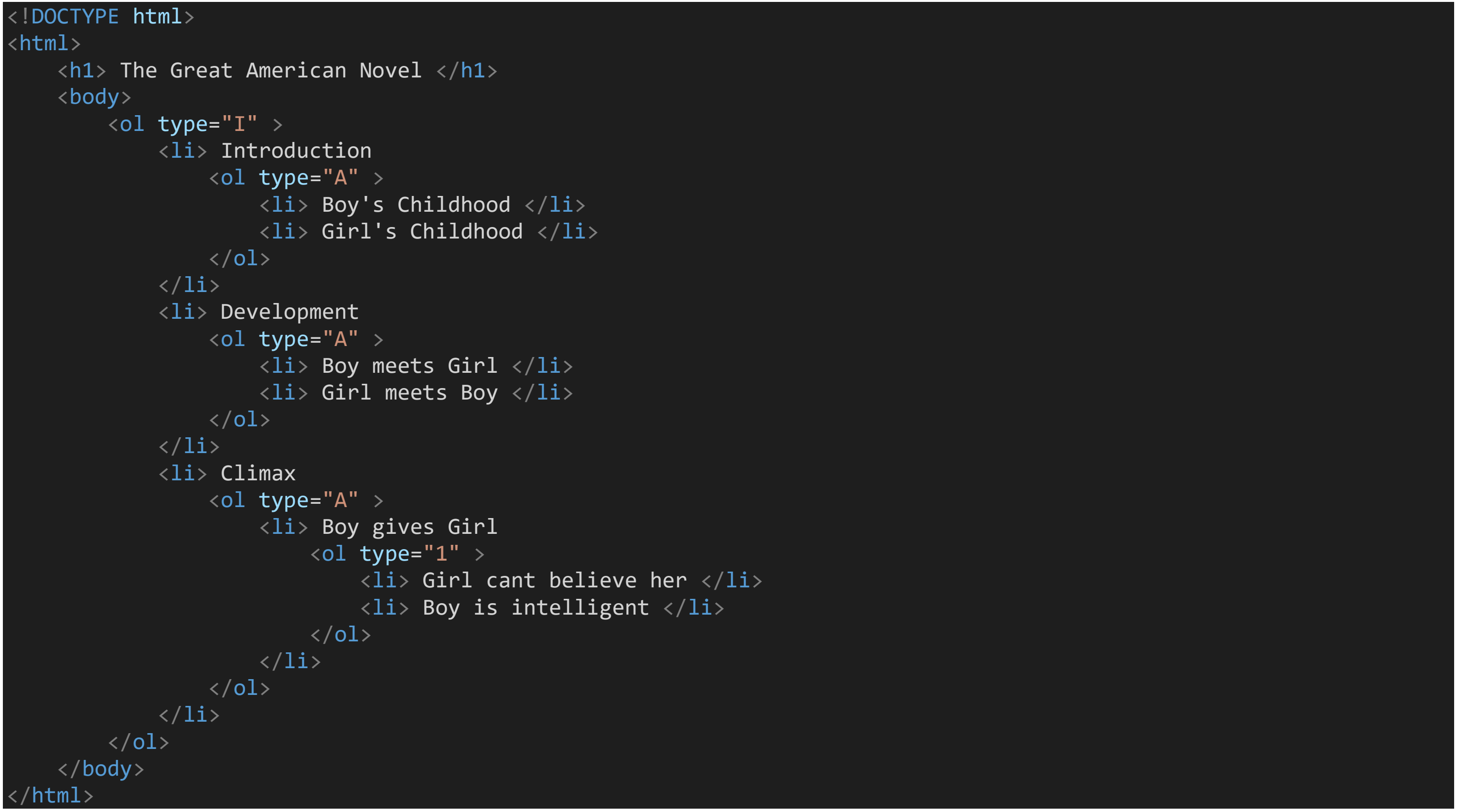
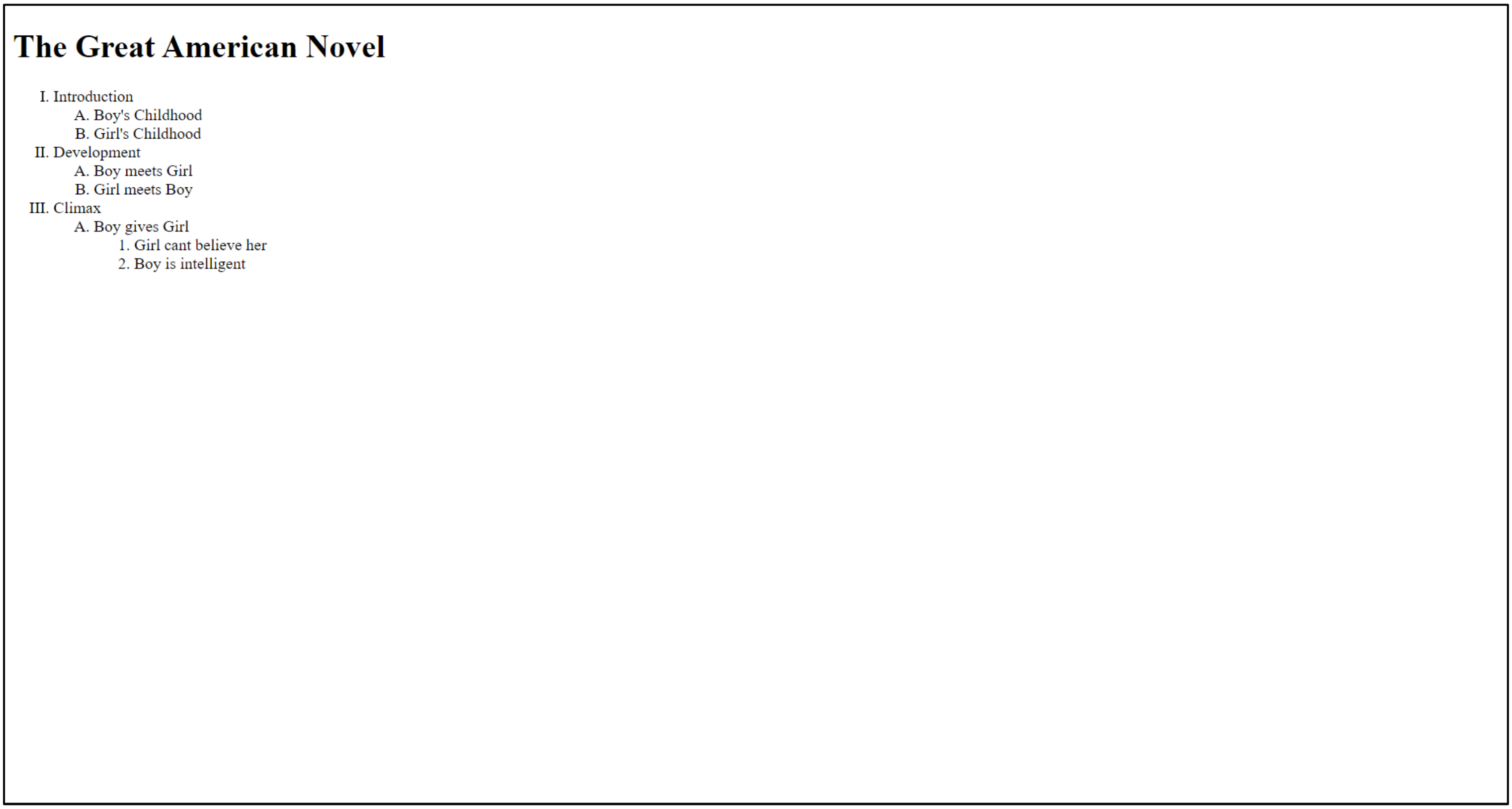
1.1 - HTML Program to work with Lists.



```
<!DOCTYPE html>
<html>
    <h1> The Great American Novel </h1>
    <body>
        <ol type="I" >
            <li> Introduction
                <ol type="A" >
                    <li> Boy's Childhood </li>
                    <li> Girl's Childhood </li>
                </ol>
            </li>
            <li> Development
                <ol type="A" >
                    <li> Boy meets Girl </li>
                    <li> Girl meets Boy </li>
                </ol>
            </li>
            <li> Climax
                <ol type="A" >
                    <li> Boy gives Girl
                        <ol type="1" >
                            <li> Girl cant believe her </li>
                            <li> Boy is intelligent </li>
                        </ol>
                    </li>
                </ol>
            </li>
        </ol>
    </body>
</html>
```

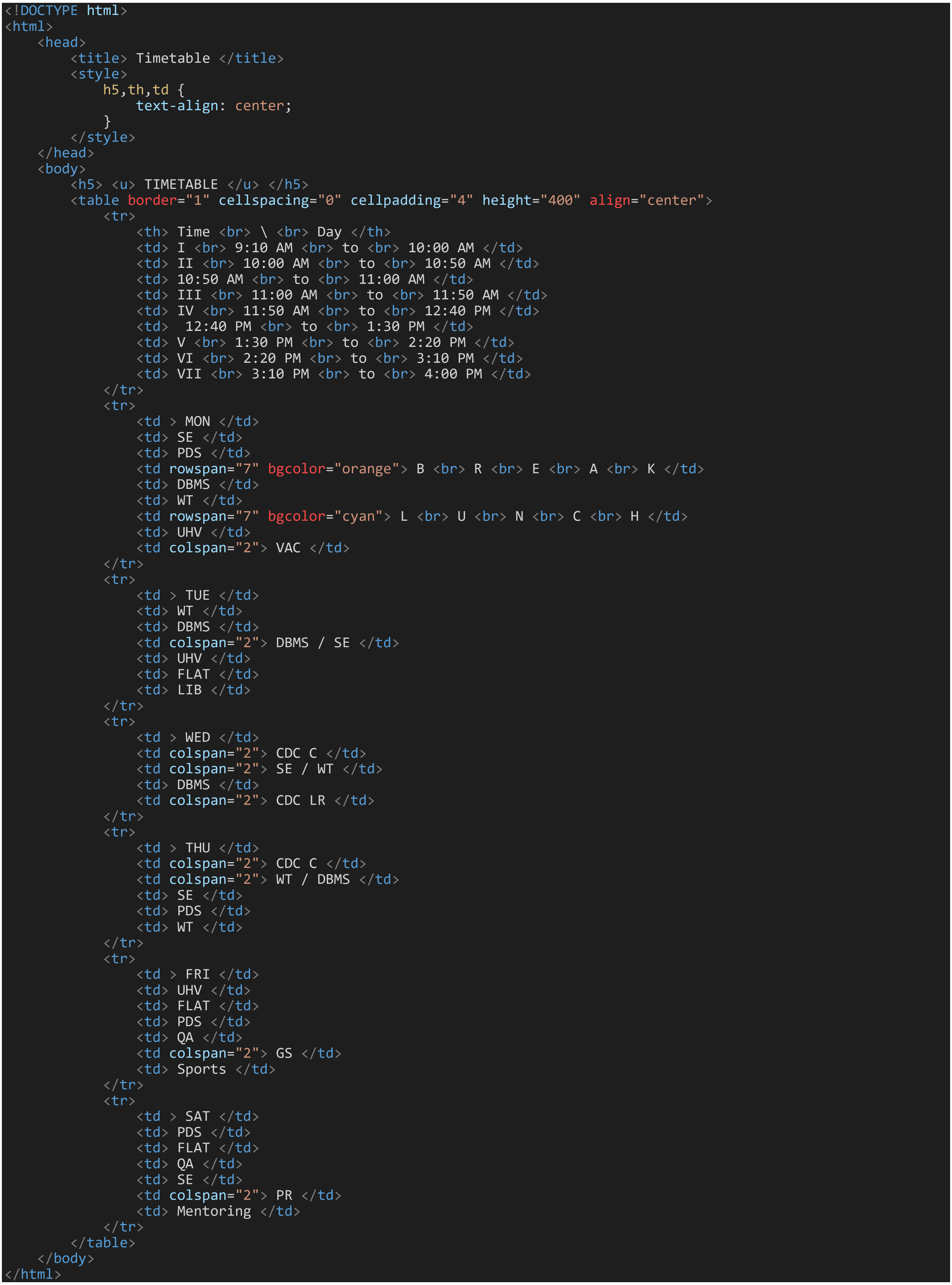
Output:



The Great American Novel

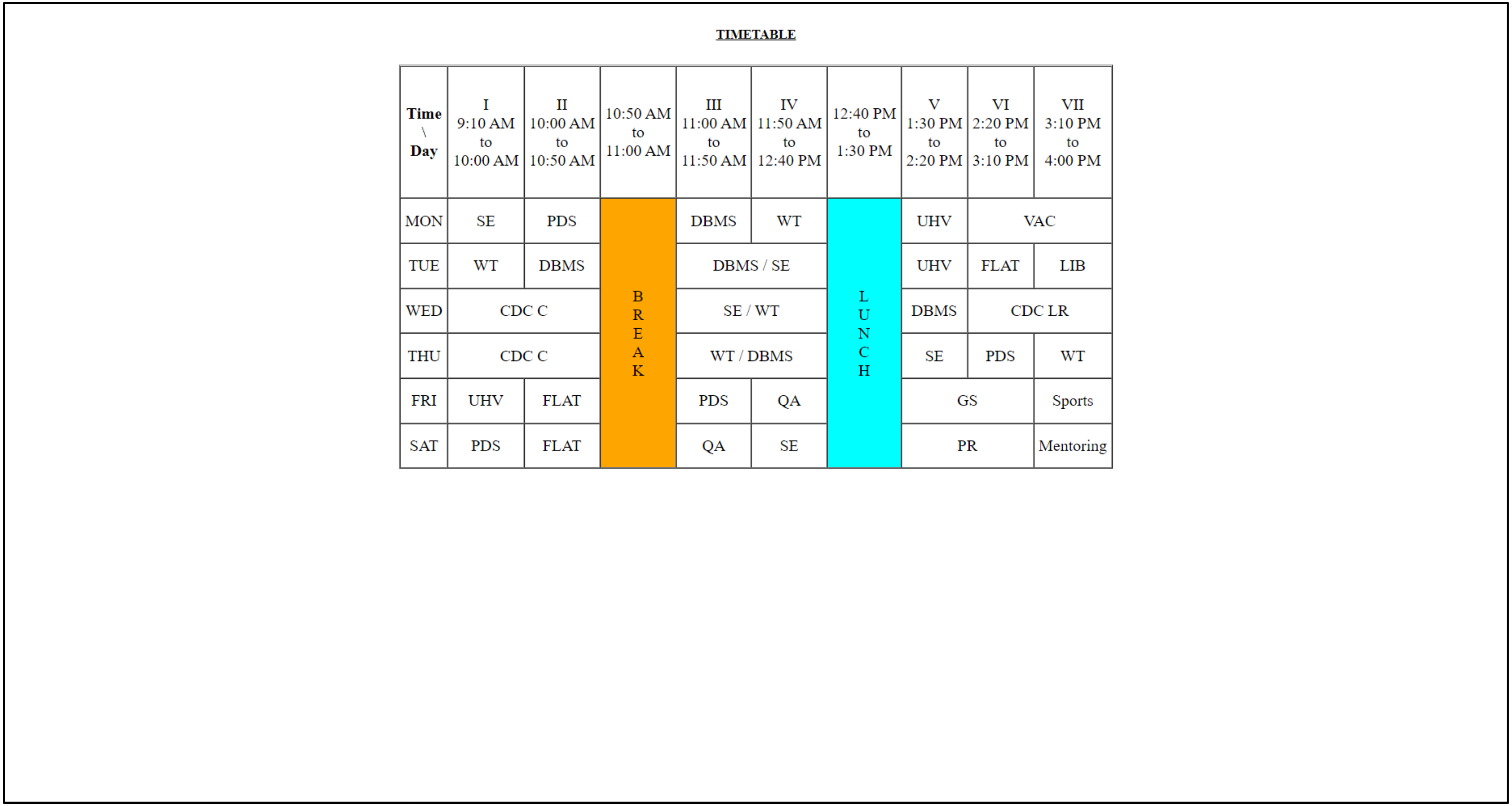
```
  I. Introduction
        A. Boy's Childhood
        B. Girl's Childhood
 II. Development
        A. Boy meets Girl
        B. Girl meets Boy
III. Climax
        A. Boy gives Girl
               1. Girl cant believe her
               2. Boy is intelligent
```

1.2 - HTML to work with tables.



```
<!DOCTYPE html>
<html>
    <head>
        <title> Timetable </title>
        <style>
            h5,th,td {
                text-align: center;
            }
        </style>
    </head>
    <body>
        <h5> <u> TIMETABLE </u> </h5>
        <table border="1" cellspacing="0" cellpadding="4" height="400" align="center">
            <tr>
                <th> Time <br> \ <br> Day </th>
                <td> I <br> 9:10 AM <br> to <br> 10:00 AM </td>
                <td> II <br> 10:00 AM <br> to <br> 10:50 AM </td>
                <td> 10:50 AM <br> to <br> 11:00 AM </td>
                <td> III <br> 11:00 AM <br> to <br> 11:50 AM </td>
                <td> IV <br> 11:50 AM <br> to <br> 12:40 PM </td>
                <td>  12:40 PM <br> to <br> 1:30 PM </td>
                <td> V <br> 1:30 PM <br> to <br> 2:20 PM </td>
                <td> VI <br> 2:20 PM <br> to <br> 3:10 PM </td>
                <td> VII <br> 3:10 PM <br> to <br> 4:00 PM </td>
            </tr>
            <tr>
                <td > MON </td>
                <td> SE </td>
                <td> PDS </td>
                <td rowspan="7" bgcolor="orange"> B <br> R <br> E <br> A <br> K </td>
                <td> DBMS </td>
                <td> WT </td>
                <td rowspan="7" bgcolor="cyan"> L <br> U <br> N <br> C <br> H </td>
                <td> UHV </td>
                <td colspan="2"> VAC </td>
            </tr>
            <tr>
                <td > TUE </td>
                <td> WT </td>
                <td> DBMS </td>
                <td colspan="2"> DBMS / SE </td>
                <td> UHV </td>
                <td> FLAT </td>
                <td> LIB </td>
            </tr>
            <tr>
                <td > WED </td>
                <td colspan="2"> CDC C </td>
                <td colspan="2"> SE / WT </td>
                <td> DBMS </td>
                <td colspan="2"> CDC LR </td>
            </tr>
            <tr>
                <td > THU </td>
                <td colspan="2"> CDC C </td>
                <td colspan="2"> WT / DBMS </td>
                <td> SE </td>
                <td> PDS </td>
                <td> WT </td>
            </tr>
            <tr>
                <td > FRI </td>
                <td> UHV </td>
                <td> FLAT </td>
                <td> PDS </td>
                <td> QA </td>
                <td colspan="2"> GS </td>
                <td> Sports </td>
            </tr>
            <tr>
                <td > SAT </td>
                <td> PDS </td>
                <td> FLAT </td>
                <td> QA </td>
                <td> SE </td>
                <td colspan="2"> PR </td>
                <td> Mentoring </td>
            </tr>
        </table>
    </body>
</html>
```

Output:



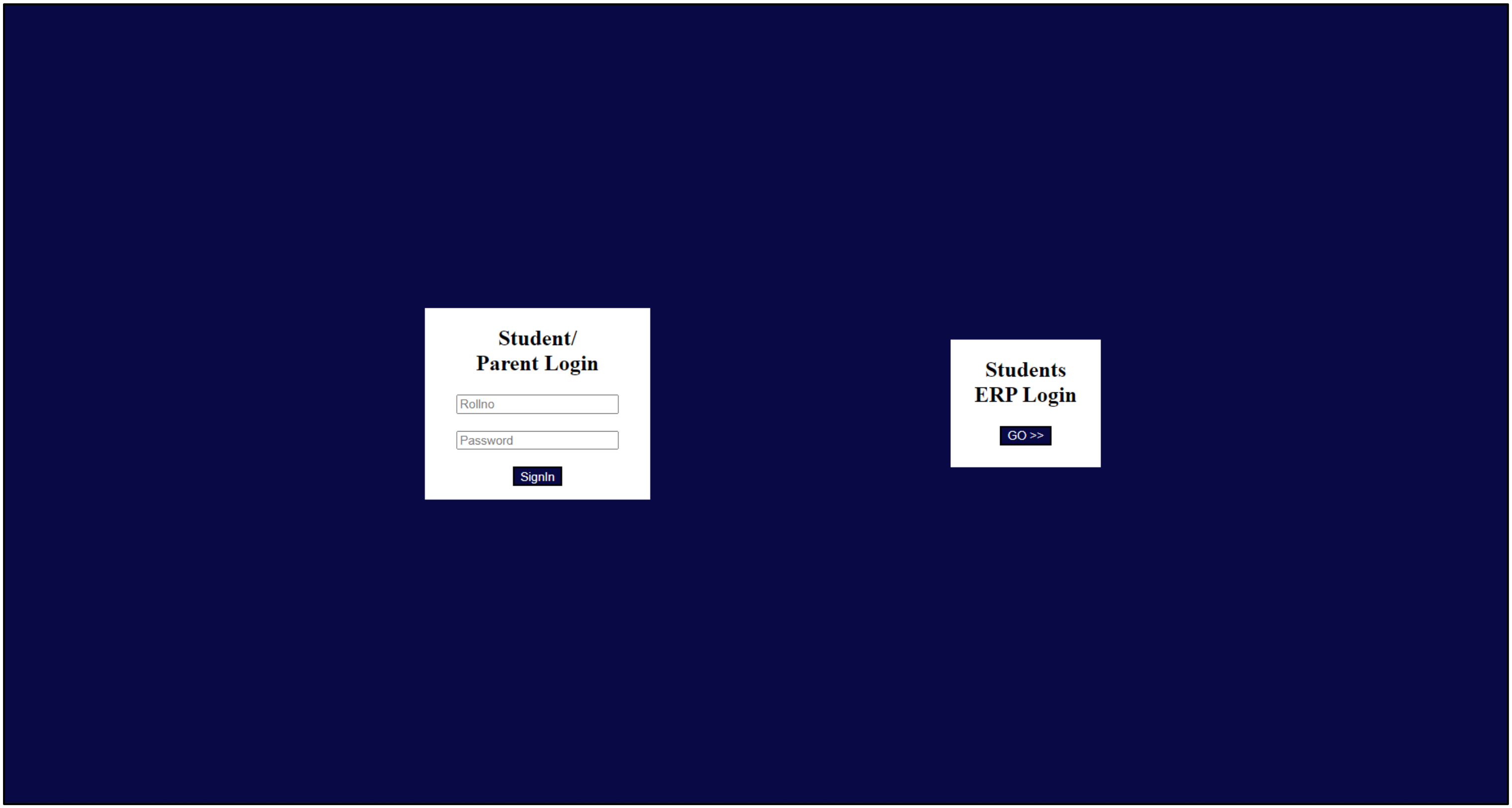
TIMETABLE

Time \\ **Day**	I 9:10 AM to 10:00 AM	II 10:00 AM to 10:50 AM	10:50 AM to 11:00 AM	III 11:00 AM to 11:50 AM	IV 11:50 AM to 12:40 PM	12:40 PM to 1:30 PM	V 1:30 PM to 2:20 PM	VI 2:20 PM to 3:10 PM	VII 3:10 PM to 4:00 PM
MON	SE	PDS	B R E A K	DBMS	WT	L U N C H	UHV	VAC	
TUE	WT	DBMS		DBMS / SE			UHV	FLAT	LIB
WED	CDC C			SE / WT			DBMS	CDC LR	
THU	CDC C			WT / DBMS			SE	PDS	WT
FRI	UHV	FLAT		PDS	QA		GS		Sports
SAT	PDS	FLAT		QA	SE		PR		Mentoring

2.1.1 - HTML Program to design replica of VMEG student login.

```html
<!doctype html>
<html style="background-color:rgb(9, 9, 70);">
    <head>
        <title> Students Corner </title>
    </head>
    <frameset frameborder="0" cols="28%, 15%, 20%, 10%, 27%">
        <frame>
        <frameset rows="38%, 24%, 38%">
            <frame>  <frame src="f1.html">  <frame>
        </frameset>
        <frame>
        <frameset rows="42%, 16%, 42%">
            <frame>  <frame src="f2.html">  <frame>
        </frameset>
        <frame>
    </frameset>
</html>
```

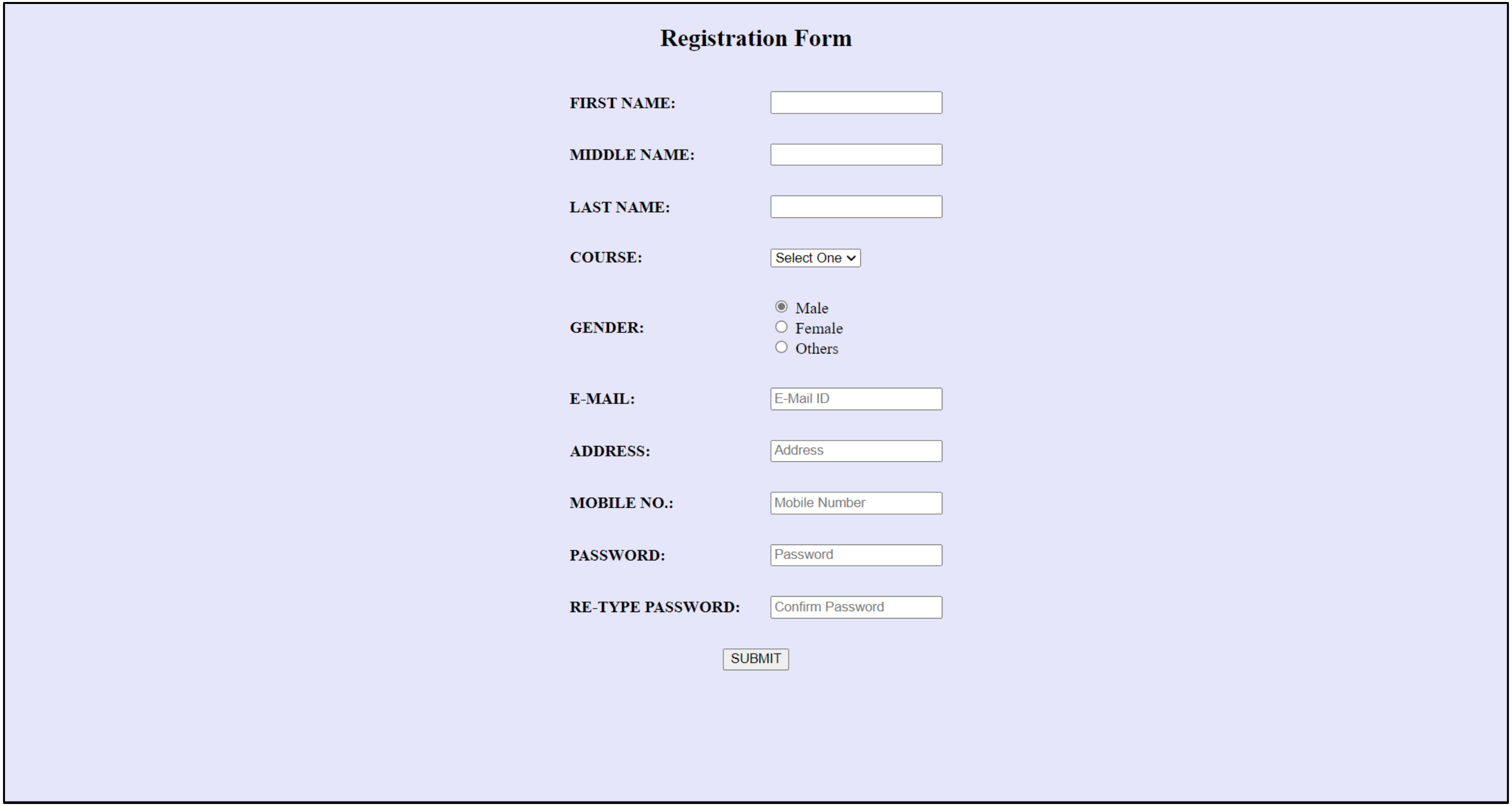
Output:



2.1.2 - HTML Program to design registration form.

```
<!DOCTYPE html>
<html style="background-color:lavender">
<head>
    <title> Students Registration Form </title>
    <style>
        th,td { text-align: left;}
    </style>
</head>
    <h2 align="center"> Registration Form  </h2>
    <table border="0" cellpadding="10" cellspacing="10" align="center">
        <tr>
            <th> FIRST NAME: </th>
            <td> <input type="text"> </td>
        </tr>
        <tr>
            <th> MIDDLE NAME: </th>
            <td> <input type="text"> </td>
        </tr>
        <tr>
            <th> LAST NAME: </th>
            <td> <input type="text"> </td>
        </tr>
        <tr>
            <th> COURSE: </th>
            <td>
                <select>
                    <option hidden="true"> Select One </option>
                    <optgroup label="Branch">
                        <option value="CSE"> CSE </option>
                        <option value="IT"> IT </option>
                        <option value="ECE"> ECE </option>
                    </optgroup>
                </select>
            </td>
        </tr>
        <tr>
            <th> GENDER: </th>
            <td>
                <input type="radio" name="gender" checked="true"> Male <br>
                <input type="radio" name="gender"> Female <br>
                <input type="radio" name="gender"> Others
            </td>
        </tr>
        <tr>
            <th> E-MAIL: </th>
            <td> <input type="email" placeholder="E-Mail ID"> </td>
        </tr>
        <tr>
            <th> ADDRESS: </th>
            <td> <input type="text" placeholder="Address"> </td>
        </tr>
        <tr>
            <th> MOBILE NO.: </th>
            <td> <input type="number" placeholder="Mobile Number"> </td>
        </tr>
        <tr>
            <th> PASSWORD: </th>
            <td> <input type="password" placeholder="Password"> </td>
        </tr>
        <tr>
            <th> RE-TYPE PASSWORD: </th>
            <td> <input type="password" placeholder="Confirm Password"> </td>
        </tr>
        <tr>
            <td colspan="2" style="text-align: center"> <input type="submit" value="SUBMIT"> </td>
        </tr>
    </table>
</html>
```

Output:



Registration Form

FIRST NAME:

MIDDLE NAME:

LAST NAME:

COURSE: Select One

GENDER: ● Male ○ Female ○ Others

E-MAIL: E-Mail ID

ADDRESS: Address

MOBILE NO.: Mobile Number

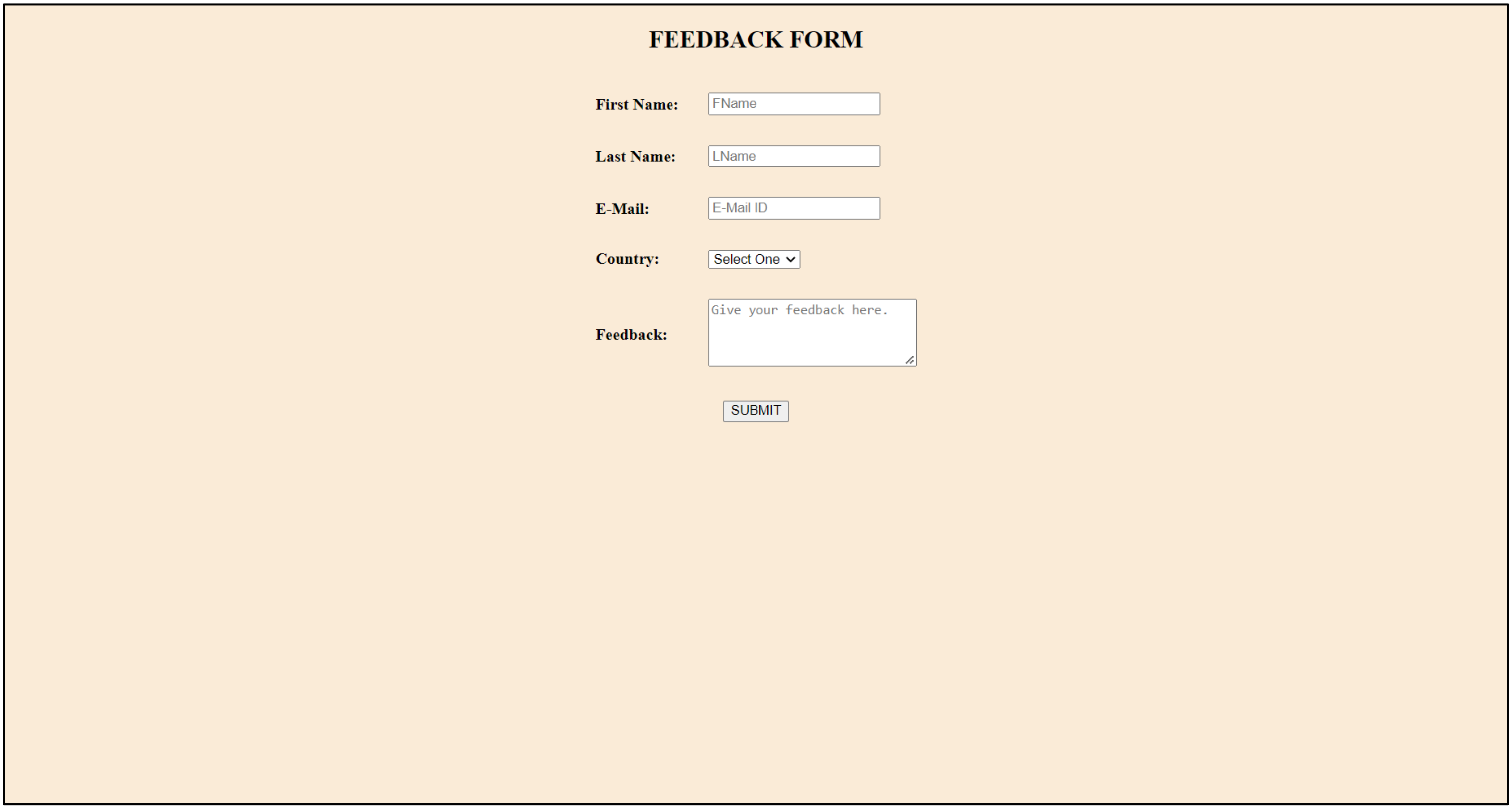
PASSWORD: Password

RE-TYPE PASSWORD: Confirm Password

2.1.3 - HTML Program to design feedback form.

```
<!DOCTYPE html>
<html style="background-color:antiquewhite">
<title> Students Feedback Form </title>
    <h2 align="center"> FEEDBACK FORM </h2>
        <table border="0" cellpadding="10" cellspacing="10" align="center">
            <tr>
                <th align="left"> First Name: </th>
                <td align="left"> <input type="text" placeholder="FName"> </td>
            </tr>
            <tr>
                <th align="left"> Last Name: </th>
                <td align="left"> <input type="text" placeholder="LName"> </td>
            </tr>
            <tr>
                <th align="left"> E-Mail: </th>
                <td align="left"> <input type="email" placeholder="E-Mail ID"> </td>
            </tr>
            <tr>
                <th align="left"> Country: </th>
                <td align="left"> <select>
                        <option hidden="true"> Select One </option>
                        <optgroup label="Country">
                            <option value="Australia"> Australia </option>
                            <option value="China"> China </option>
                            <option value="India"> India </option>
                        </optgroup>
                    </select>
                </td>
            </tr>
            <tr>
                <th align="left"> Feedback: </th>
                <td align="left"> <textarea rows="4" cols="25" placeholder="Give your feedback here."></textarea>
                </td>
            </tr>
            <tr>
                <td colspan="2" align="center"> <input type="submit" value="SUBMIT"> </td>
            </tr>
        </table>
</html>
```

Output:



FEEDBACK FORM

First Name:	FName
Last Name:	LName
E-Mail:	E-Mail ID
Country:	Select One
Feedback:	Give your feedback here.

2.2 - CSS Program to work with background and border properties.

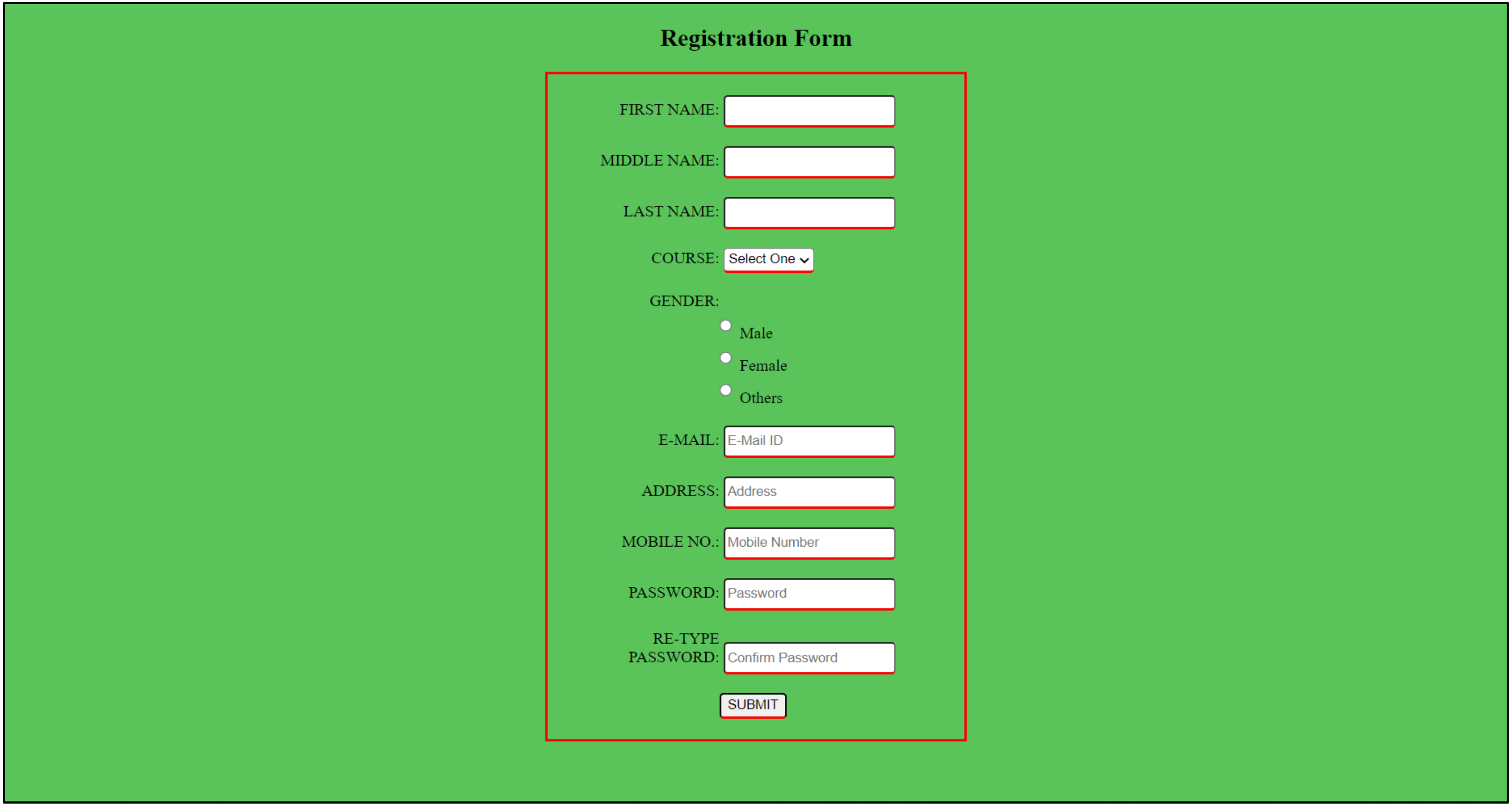
Exercise_2.2.html

```html
<!DOCTYPE html>
<html>
<head>
    <link rel="stylesheet" type="text/css" href="Exercise_2.2.css"/>
</head>
    <h2 align="center"> Registration Form  </h2>
    <body>
    <form>
        <label>FIRST NAME:</label>
            <input type="text"> <br> <br>
        <label>MIDDLE NAME:</label>
            <input type="text"> <br> <br>
        <label>LAST NAME:</label>
            <input type="text"> <br> <br>
        <label>COURSE:</label>
            <select>
                <option hidden="true"> Select One </option>
                <optgroup label="Branch">
                    <option value="CSE"> CSE </option>
                    <option value="IT"> IT </option>
                    <option value="ECE"> ECE </option>
                </optgroup>
            </select> <br> <br>
        <label>GENDER:</label> <br>
            <input type="radio" name="gender" class="gender"> Male <br>
            <input type="radio" name="gender" class="gender"> Female <br>
            <input type="radio" name="gender" class="gender"> Others <br> <br>
        <label>E-MAIL:</label>
            <input type="email" placeholder="E-Mail ID"> <br> <br>
        <label>ADDRESS:</label>
            <input type="text" placeholder="Address"> <br> <br>
        <label>MOBILE NO.:</label>
            <input type="number" placeholder="Mobile Number"> <br> <br>
        <label>PASSWORD:</label>
            <input type="password" placeholder="Password"> <br> <br>
        <label>RE-TYPE PASSWORD:</label>
            <input type="password" placeholder="Confirm Password"> <br> <br>

        <input  style="margin-left:150px" type="submit" value="SUBMIT"   >
    </form>
</body>
</html>
```

Exercise_2.2.css

```css
input,select
{
    border-radius:4px;
    height:25px;
    border-bottom:red solid 3px;
}
body
{
    background-color:rgb(90, 196, 90);
    size: 100%;
}
form
{
    border:red solid 3px;
    padding:20px;
    margin-left:400px;
    margin-right:400px;
    top:100px;
}
label
{
    text-align:right;
    width:150px;
    display:inline-block;
}
.gender
{
    margin-left:150px
}
```

Output:



Registration Form

FIRST NAME:

MIDDLE NAME:

LAST NAME:

COURSE: Select One

GENDER:

○ Male

○ Female

○ Others

E-MAIL: E-Mail ID

ADDRESS: Address

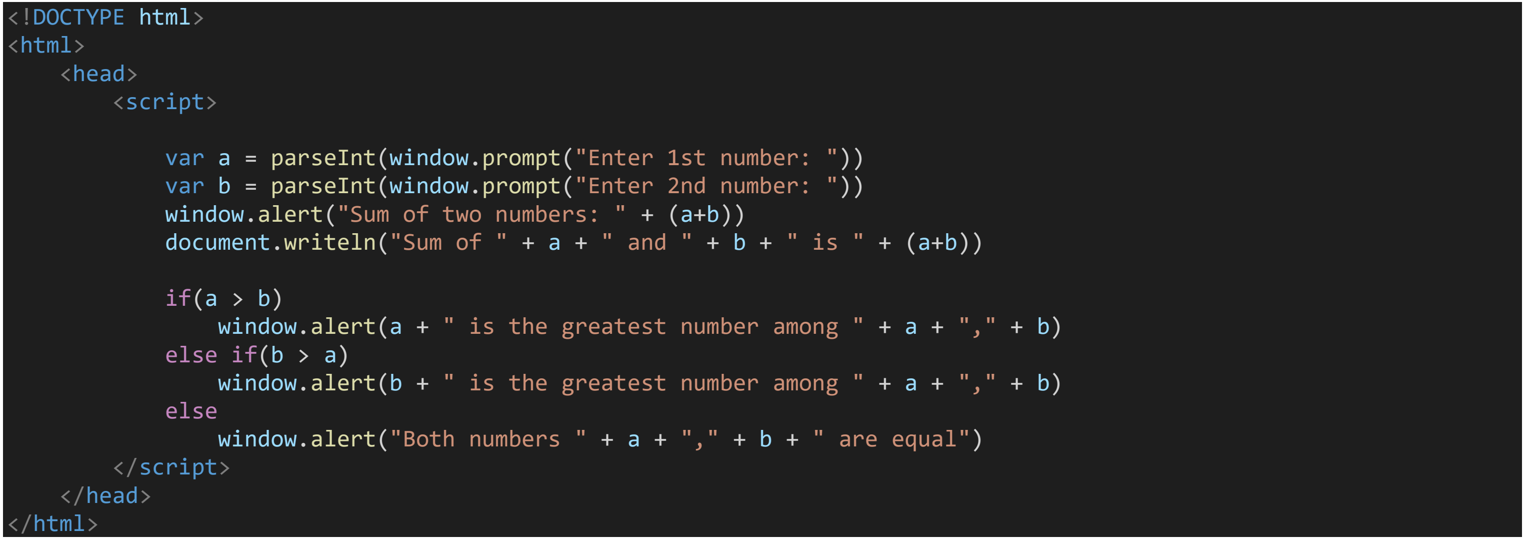
MOBILE NO.: Mobile Number

PASSWORD: Password

RE-TYPE PASSWORD: Confirm Password

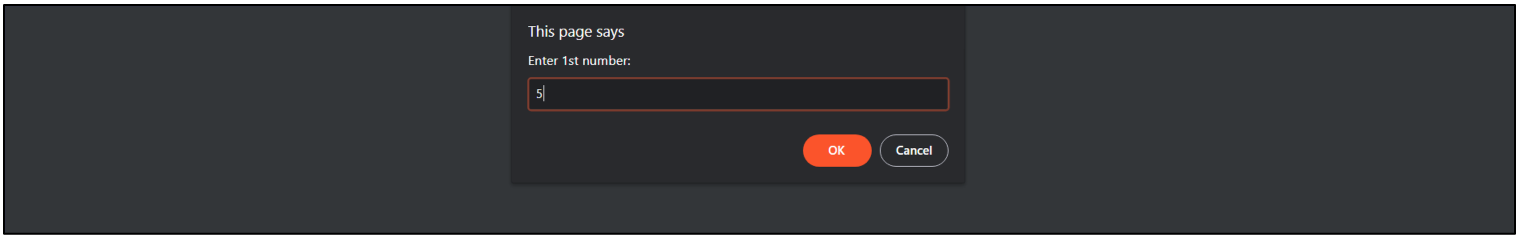
SUBMIT

3.0 - JavaScript program to find out sum of two numbers & greatest of them.



```
<!DOCTYPE html>
<html>
    <head>
        <script>

            var a = parseInt(window.prompt("Enter 1st number: "))
            var b = parseInt(window.prompt("Enter 2nd number: "))
            window.alert("Sum of two numbers: " + (a+b))
            document.writeln("Sum of " + a + " and " + b + " is " + (a+b))

            if(a > b)
                window.alert(a + " is the greatest number among " + a + "," + b)
            else if(b > a)
                window.alert(b + " is the greatest number among " + a + "," + b)
            else
                window.alert("Both numbers " + a + "," + b + " are equal")
        </script>
    </head>
</html>
```

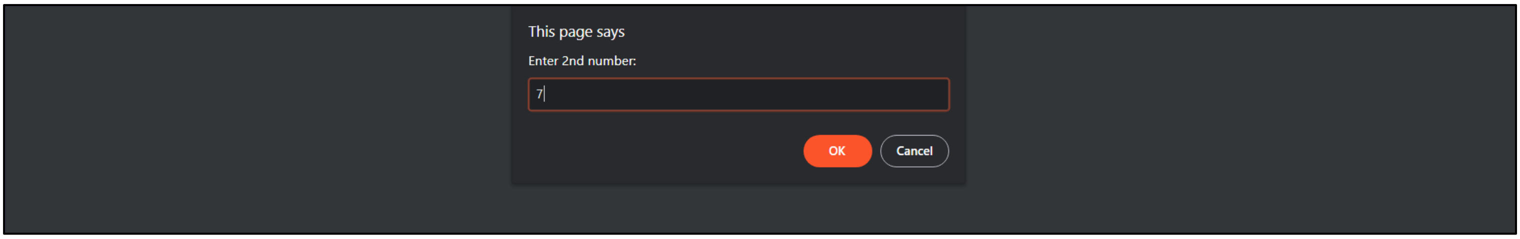
Output:



This page says

Enter 1st number:

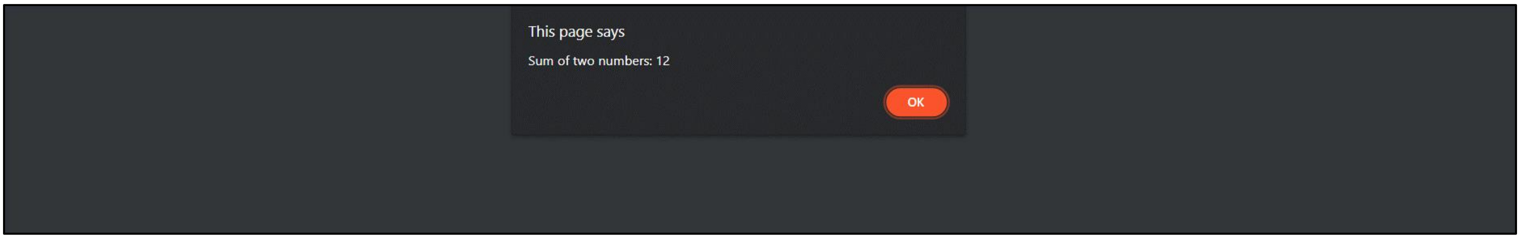
5



This page says

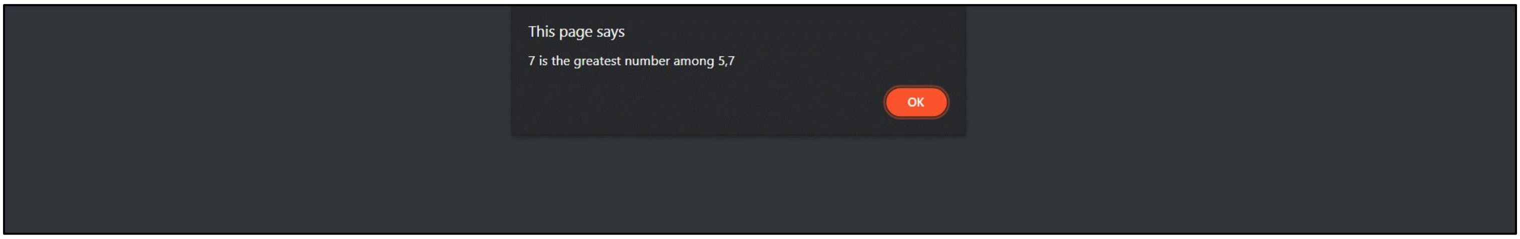
Enter 2nd number:

7



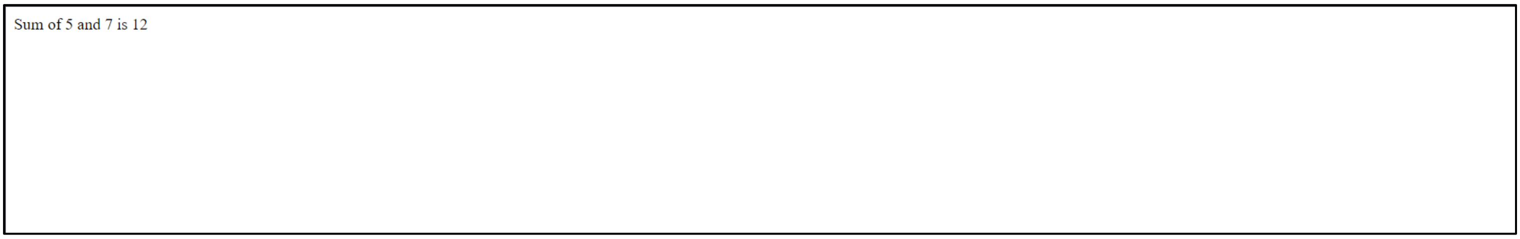
This page says

Sum of two numbers: 12



This page says

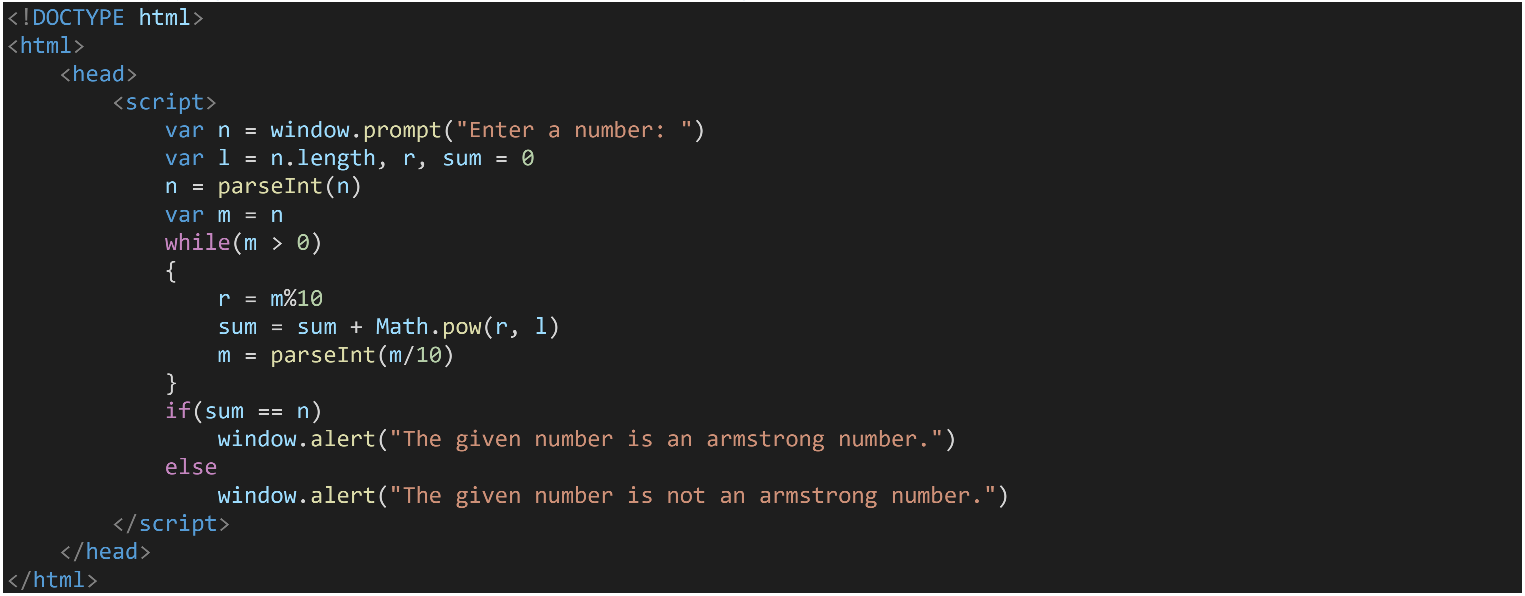
7 is the greatest number among 5,7



Sum of 5 and 7 is 12

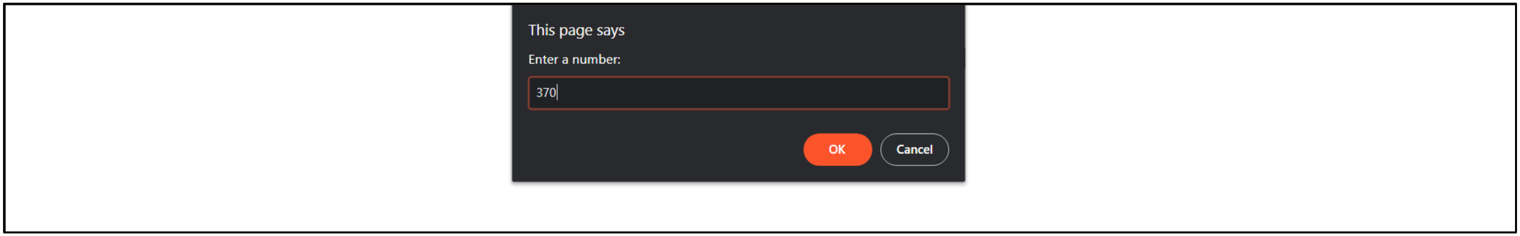
* Output images were cropped to fit in 1 page.

3.1 - JavaScript program to check whether a number is an Armstrong number or not.



```
<!DOCTYPE html>
<html>
    <head>
        <script>
            var n = window.prompt("Enter a number: ")
            var l = n.length, r, sum = 0
            n = parseInt(n)
            var m = n
            while(m > 0)
            {
                r = m%10
                sum = sum + Math.pow(r, l)
                m = parseInt(m/10)
            }
            if(sum == n)
                window.alert("The given number is an armstrong number.")
            else
                window.alert("The given number is not an armstrong number.")
        </script>
    </head>
</html>
```

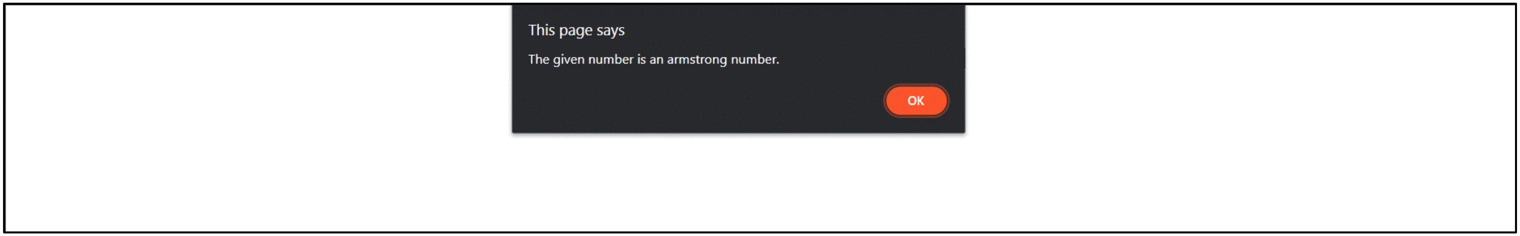
Output:



This page says

Enter a number:

370

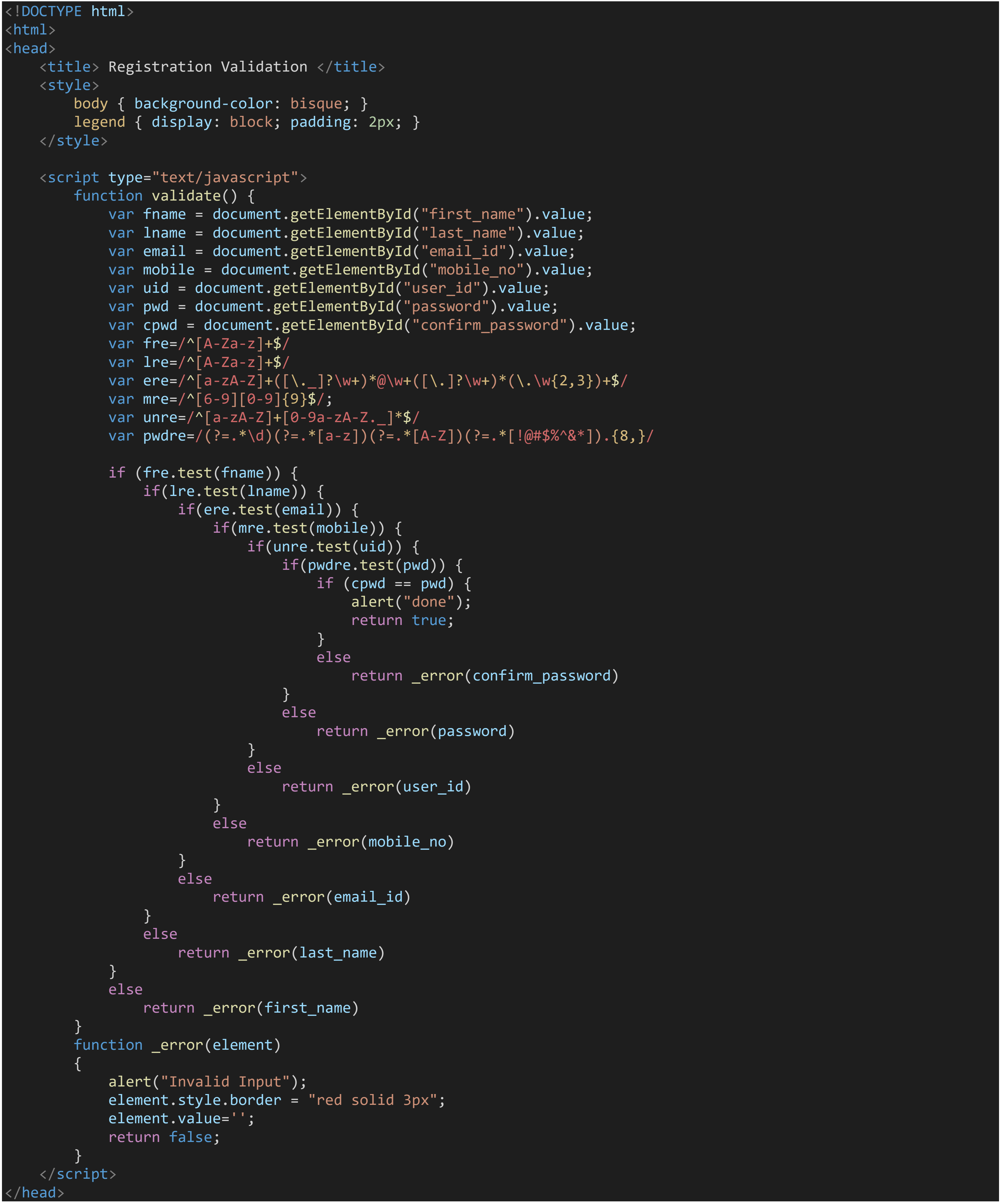


This page says

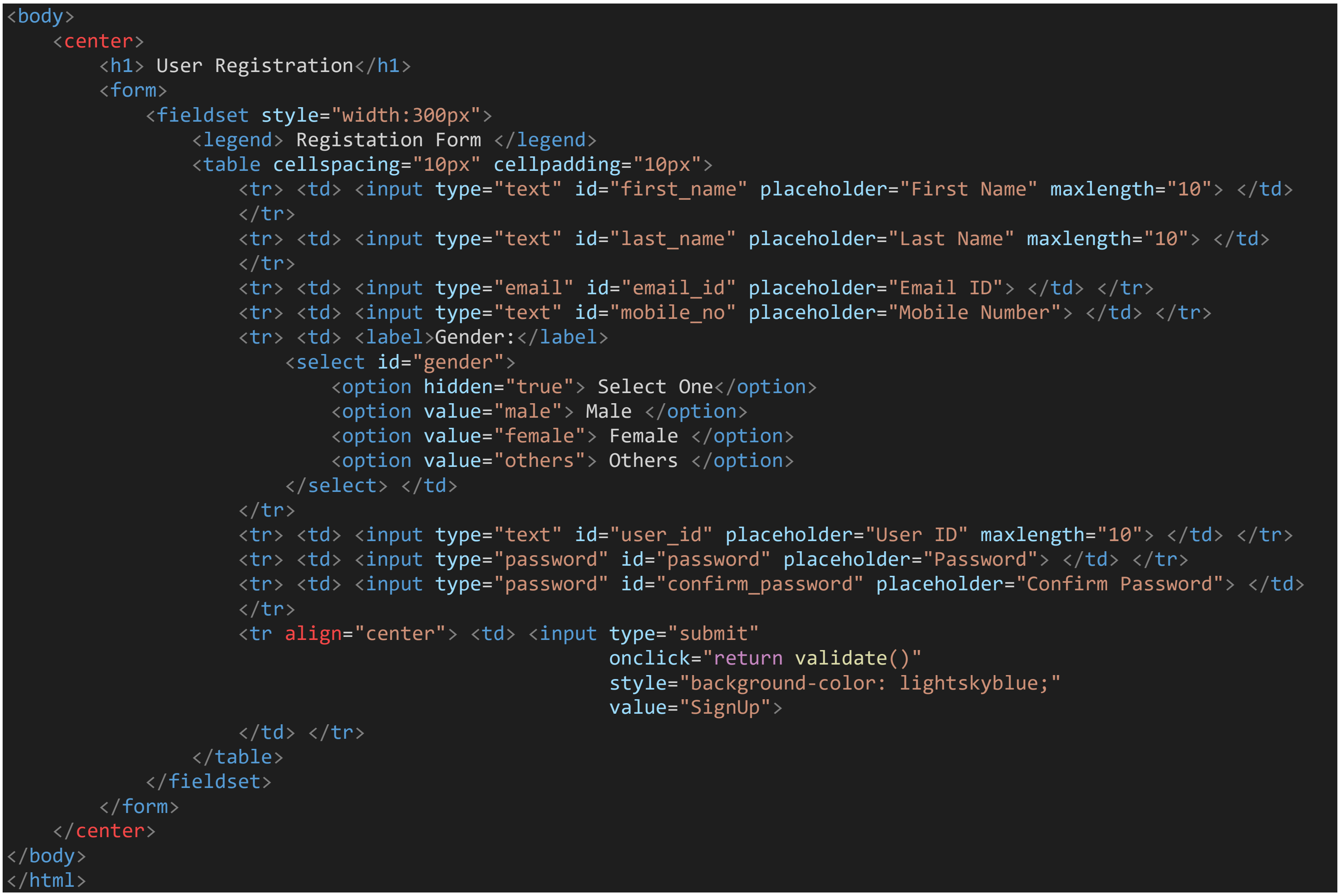
The given number is an armstrong number.

* Output images were cropped to fit in 1 page.

3.2 - JavaScript program to validate username, password, mobile number and email.

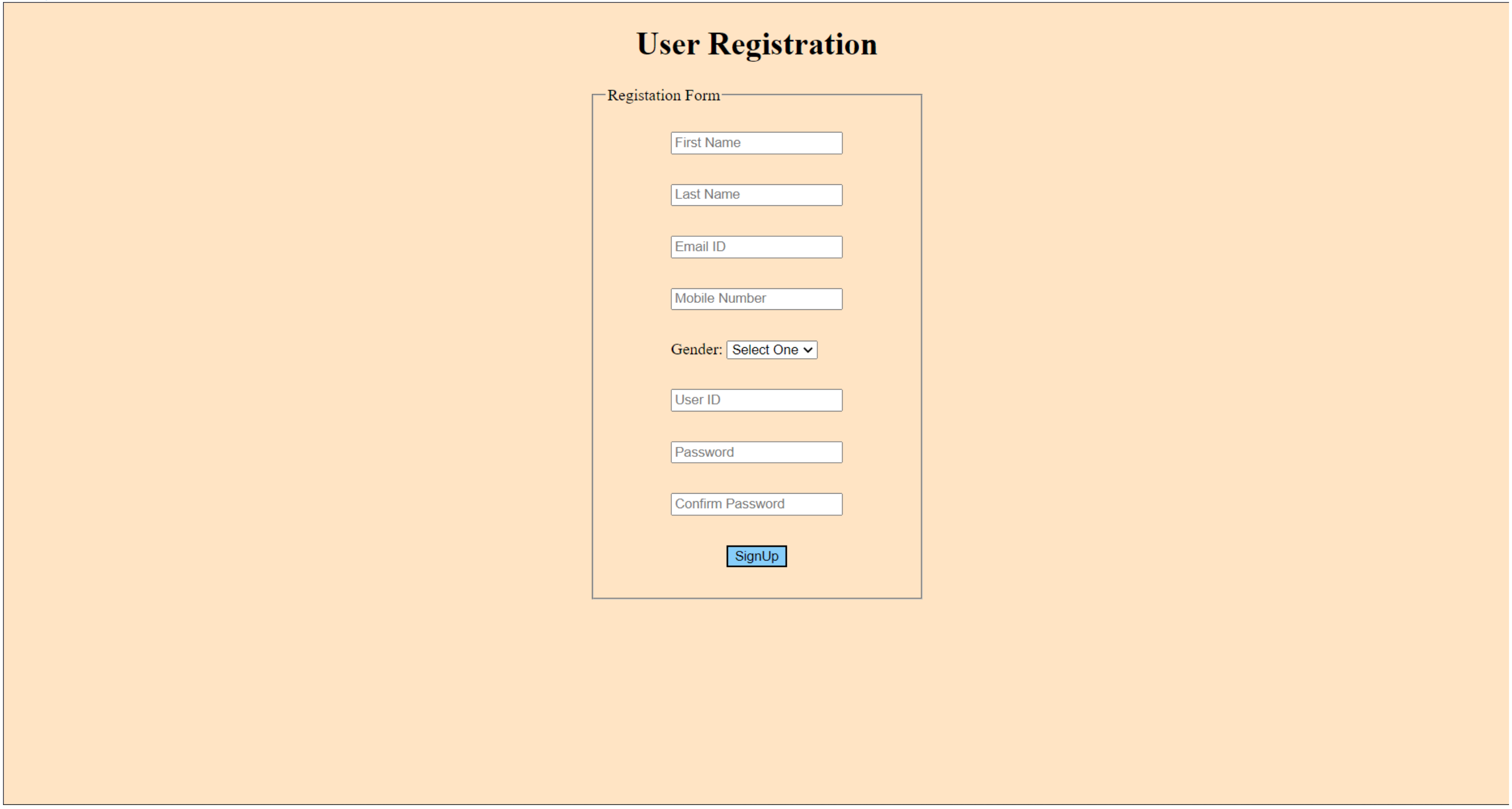


```
<!DOCTYPE html>
<html>
<head>
    <title> Registration Validation </title>
    <style>
        body { background-color: bisque; }
        legend { display: block; padding: 2px; }
    </style>

    <script type="text/javascript">
        function validate() {
            var fname = document.getElementById("first_name").value;
            var lname = document.getElementById("last_name").value;
            var email = document.getElementById("email_id").value;
            var mobile = document.getElementById("mobile_no").value;
            var uid = document.getElementById("user_id").value;
            var pwd = document.getElementById("password").value;
            var cpwd = document.getElementById("confirm_password").value;
            var fre=/^[A-Za-z]+$/
            var lre=/^[A-Za-z]+$/
            var ere=/^[a-zA-Z]+([\._]?\w+)*@\w+([\.]?\w+)*(\.\w{2,3})+$/
            var mre=/^[6-9][0-9]{9}$/;
            var unre=/^[a-zA-Z]+[0-9a-zA-Z._]*$/
            var pwdre=/(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?=.*[!@#$%^&*]).{8,}/

            if (fre.test(fname)) {
                if(lre.test(lname)) {
                    if(ere.test(email)) {
                        if(mre.test(mobile)) {
                            if(unre.test(uid)) {
                                if(pwdre.test(pwd)) {
                                    if (cpwd == pwd) {
                                        alert("done");
                                        return true;
                                    }
                                    else
                                        return _error(confirm_password)
                                }
                                else
                                    return _error(password)
                            }
                            else
                                return _error(user_id)
                        }
                        else
                            return _error(mobile_no)
                    }
                    else
                        return _error(email_id)
                }
                else
                    return _error(last_name)
            }
            else
                return _error(first_name)
        }
        function _error(element)
        {
            alert("Invalid Input");
            element.style.border = "red solid 3px";
            element.value='';
            return false;
        }
    </script>
</head>
```



```html
<body>
    <center>
        <h1> User Registration</h1>
        <form>
            <fieldset style="width:300px">
                <legend> Registation Form </legend>
                <table cellspacing="10px" cellpadding="10px">
                    <tr> <td> <input type="text" id="first_name" placeholder="First Name" maxlength="10"> </td>
                    </tr>
                    <tr> <td> <input type="text" id="last_name" placeholder="Last Name" maxlength="10"> </td>
                    </tr>
                    <tr> <td> <input type="email" id="email_id" placeholder="Email ID"> </td> </tr>
                    <tr> <td> <input type="text" id="mobile_no" placeholder="Mobile Number"> </td> </tr>
                    <tr> <td> <label>Gender:</label>
                        <select id="gender">
                            <option hidden="true"> Select One</option>
                            <option value="male"> Male </option>
                            <option value="female"> Female </option>
                            <option value="others"> Others </option>
                        </select> </td>
                    </tr>
                    <tr> <td> <input type="text" id="user_id" placeholder="User ID" maxlength="10"> </td> </tr>
                    <tr> <td> <input type="password" id="password" placeholder="Password"> </td> </tr>
                    <tr> <td> <input type="password" id="confirm_password" placeholder="Confirm Password"> </td>
                    </tr>
                    <tr align="center"> <td> <input type="submit"
                                                    onclick="return validate()"
                                                    style="background-color: lightskyblue;"
                                                    value="SignUp">
                    </td> </tr>
                </table>
            </fieldset>
        </form>
    </center>
</body>
</html>
```

Output:



User Registration

Registation Form

First Name

Last Name

Email ID

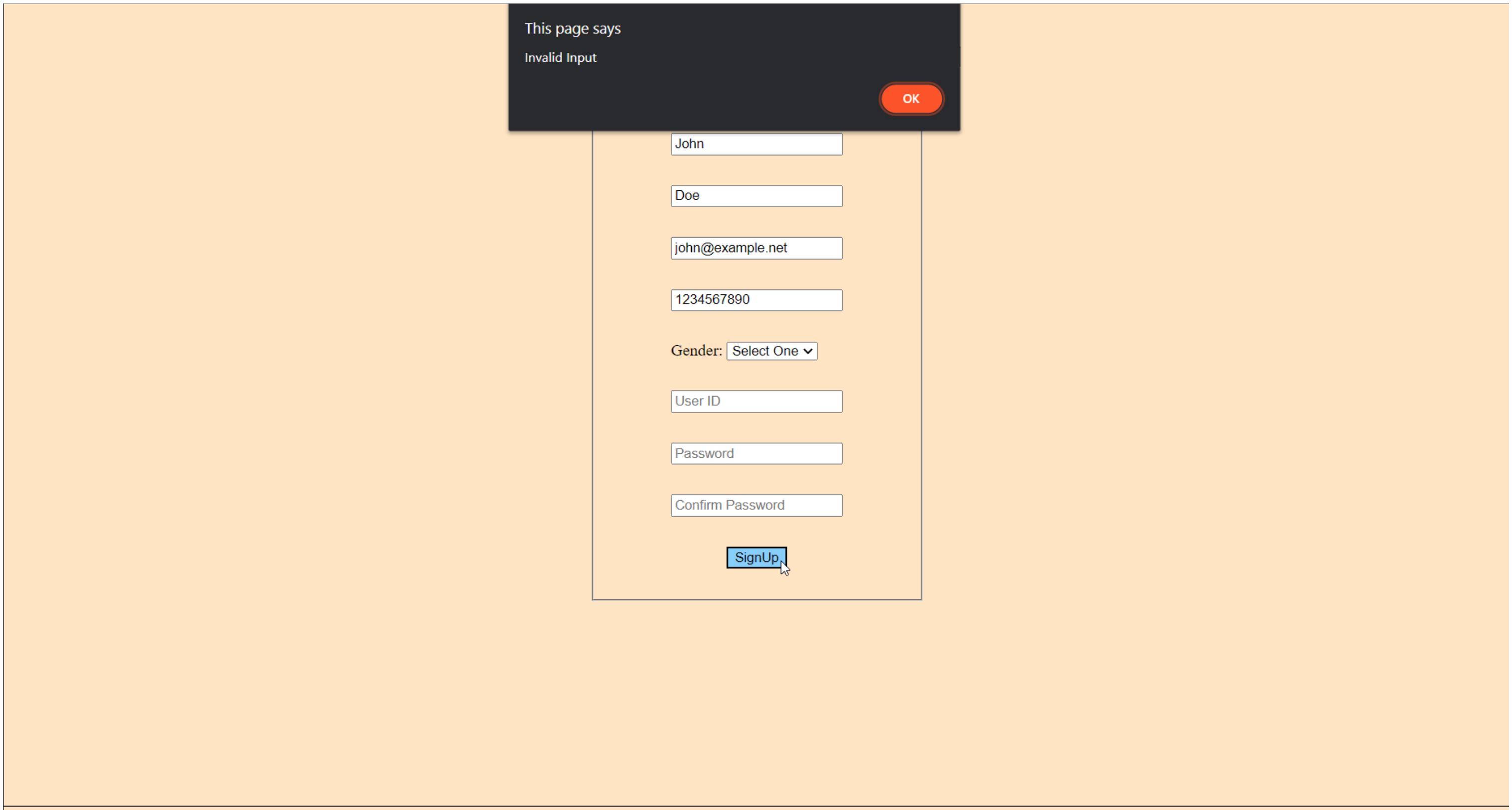
Mobile Number

Gender: Select One

User ID

Password

Confirm Password



This page says

Invalid Input

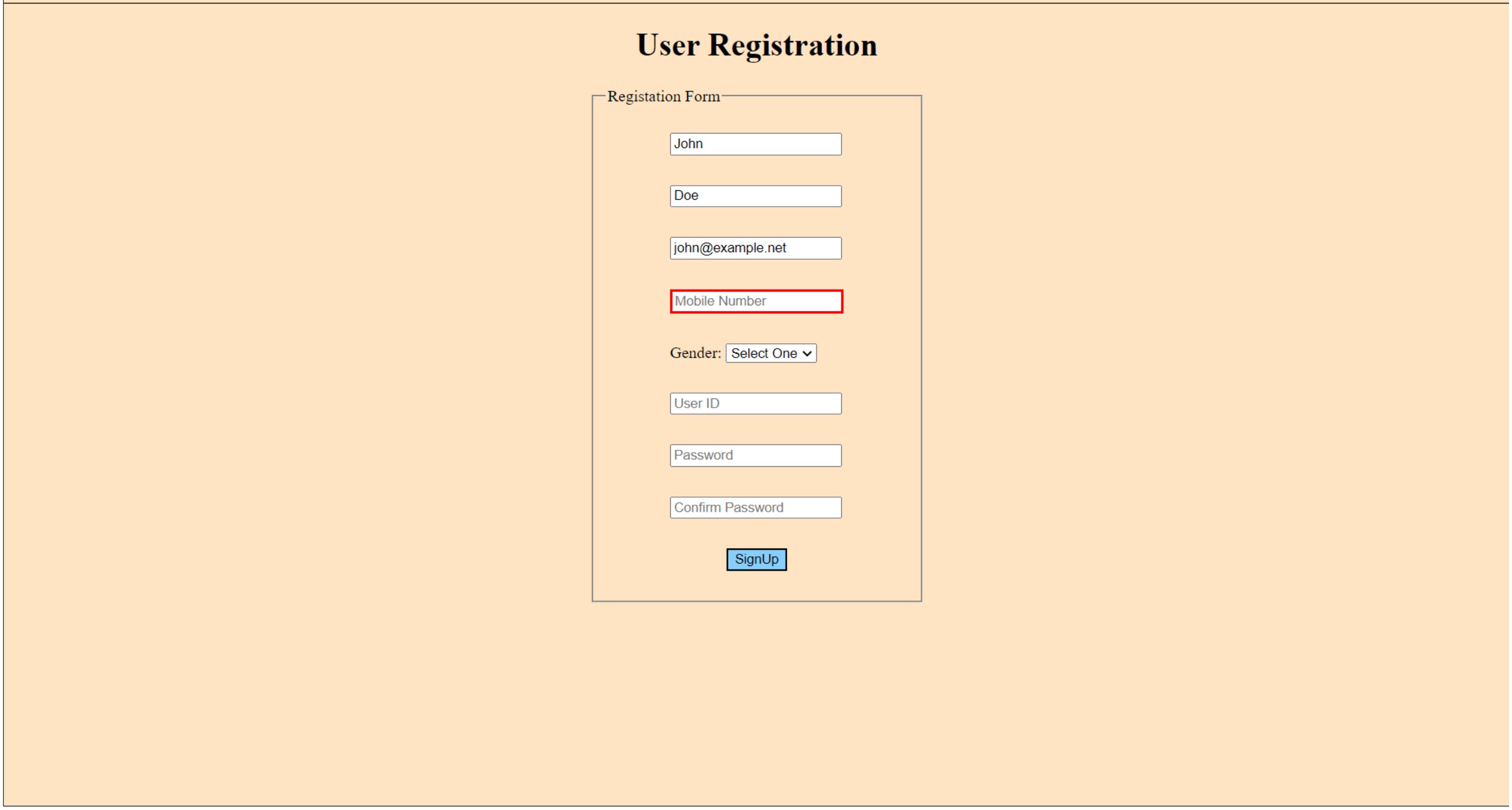
John

Doe

john@example.net

1234567890

Gender: Select One



User Registration

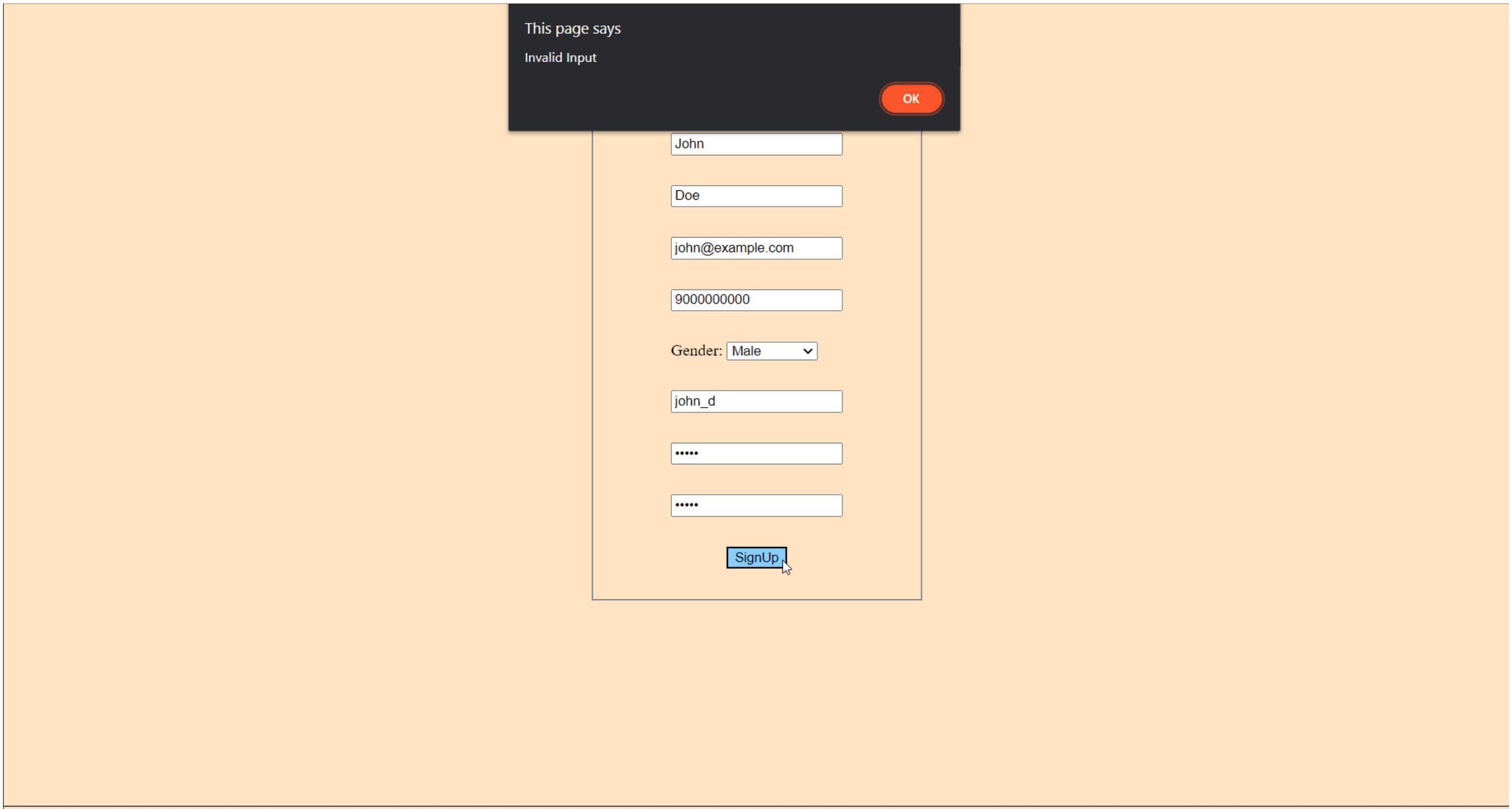
Registation Form

John

Doe

john@example.net

Gender: Select One



This page says

Invalid Input

John

Doe

john@example.com

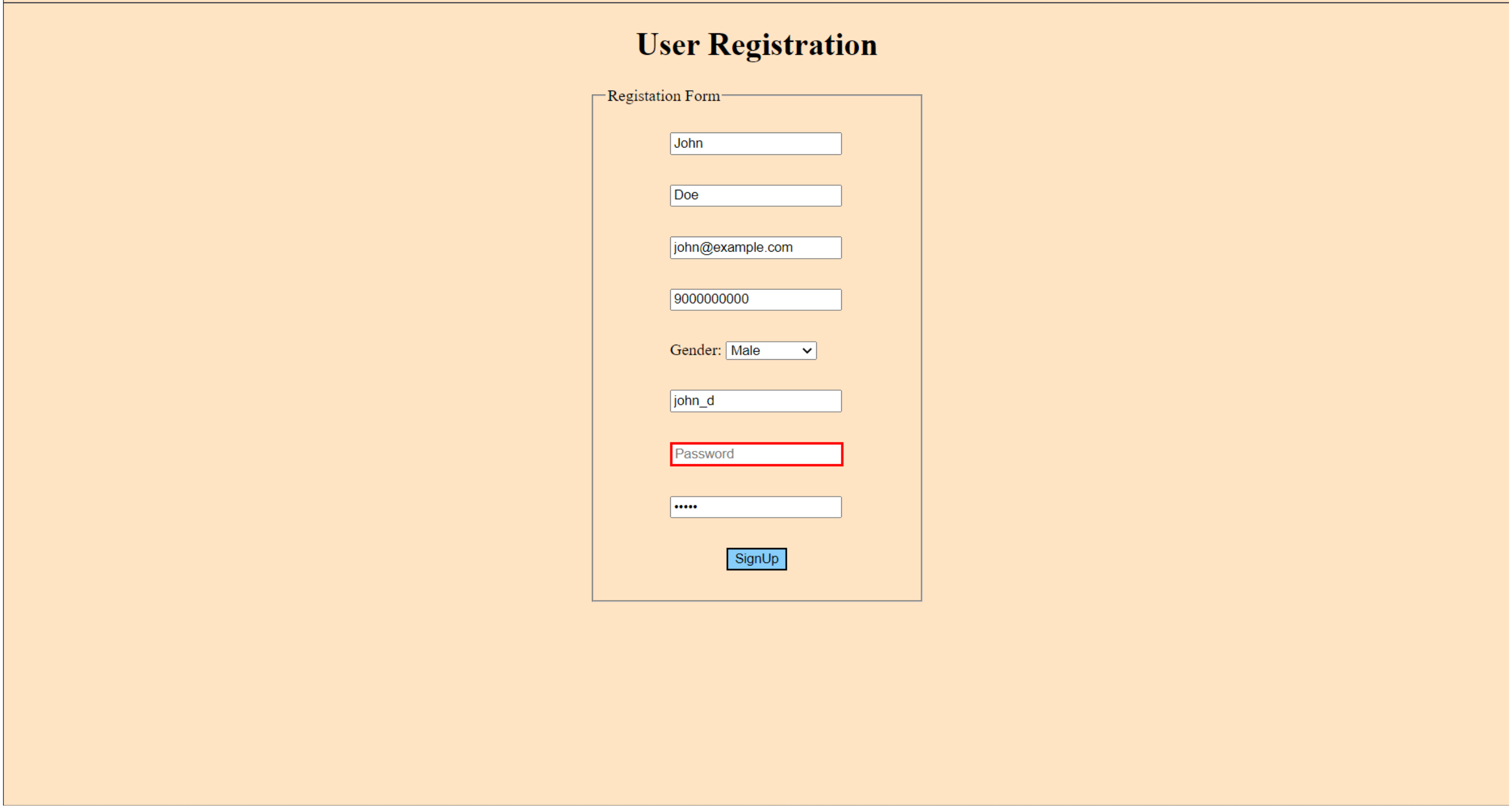
9000000000

Gender: Male

john_d

•••••

•••••



User Registration

Registation Form

John

Doe

john@example.com

9000000000

Gender: Male

john_d

Password

•••••

*Password entered was 12345

4.1 - Create a DTD document to validate an XML document.

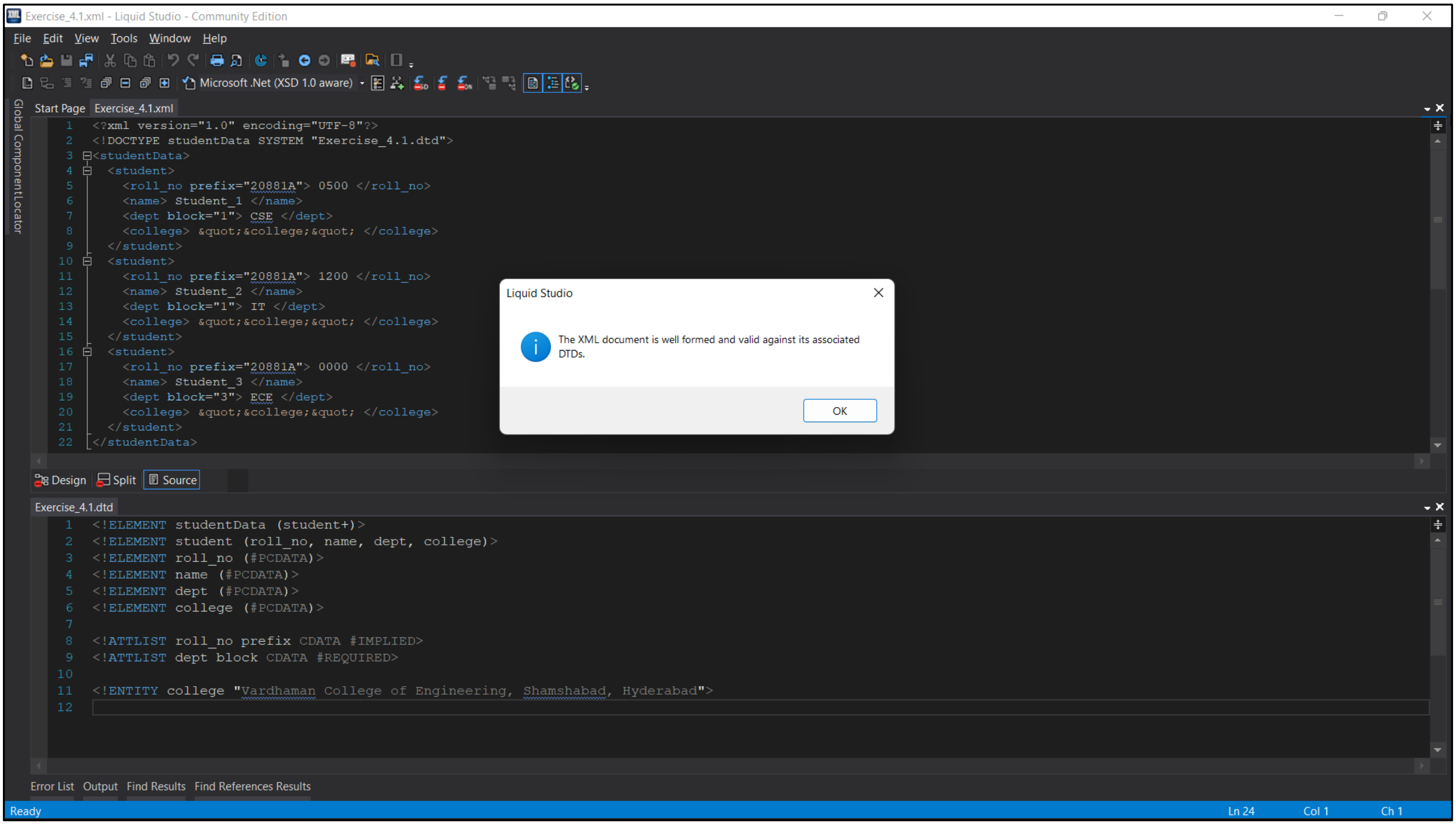
Exercise_4.1.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE studentData SYSTEM "Exercise_4.1.dtd">
<studentData>
  <student>
    <roll_no prefix="20881A"> 0500 </roll_no>
    <name> Student_1 </name>
    <dept block="1"> CSE </dept>
    <college> &quot;&college;&quot; </college>
  </student>
  <student>
    <roll_no prefix="20881A"> 1200 </roll_no>
    <name> Student_2 </name>
    <dept block="1"> IT </dept>
    <college> &quot;&college;&quot; </college>
  </student>
  <student>
    <roll_no prefix="20881A"> 0000 </roll_no>
    <name> Student_3 </name>
    <dept block="3"> ECE </dept>
    <college> &quot;&college;&quot; </college>
  </student>
</studentData>
```

Exercise_4.1.dtd

```
<!ELEMENT studentData (student+)>
<!ELEMENT student (roll_no, name, dept, college)>
<!ELEMENT roll_no (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT dept (#PCDATA)>
<!ELEMENT college (#PCDATA)>

<!ATTLIST roll_no prefix CDATA #IMPLIED>
<!ATTLIST dept block CDATA #REQUIRED>

<!ENTITY college "Vardhaman College of Engineering, Shamshabad, Hyderabad">
```

Output:


Exercise_4.1.xml - Liquid Studio - Community Edition
File Edit View Tools Window Help
Microsoft .Net (XSD 1.0 aware)
Global Component Locator
Start Page Exercise_4.1.xml
Design Split Source
Exercise_4.1.dtd
Error List Output Find Results Find References Results

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE studentData SYSTEM "Exercise_4.1.dtd">
<studentData>
  <student>
    <roll_no prefix="20881A"> 0500 </roll_no>
    <name> Student_1 </name>
    <dept block="1"> CSE </dept>
    <college> &quot;&college;&quot; </college>
  </student>
  <student>
    <roll_no prefix="20881A"> 1200 </roll_no>
    <name> Student_2 </name>
    <dept block="1"> IT </dept>
    <college> &quot;&college;&quot; </college>
  </student>
  <student>
    <roll_no prefix="20881A"> 0000 </roll_no>
    <name> Student_3 </name>
    <dept block="3"> ECE </dept>
    <college> &quot;&college;&quot; </college>
  </student>
</studentData>
```

Liquid Studio

The XML document is well formed and valid against its associated DTDs.

OK

```
<!ELEMENT studentData (student+)>
<!ELEMENT student (roll_no, name, dept, college)>
<!ELEMENT roll_no (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT dept (#PCDATA)>
<!ELEMENT college (#PCDATA)>

<!ATTLIST roll_no prefix CDATA #IMPLIED>
<!ATTLIST dept block CDATA #REQUIRED>

<!ENTITY college "Vardhaman College of Engineering, Shamshabad, Hyderabad">
```

Ready Ln 24 Col 1 Ch 1

4.2 - Create an XSD (XML Schema) document to validate an XML document.

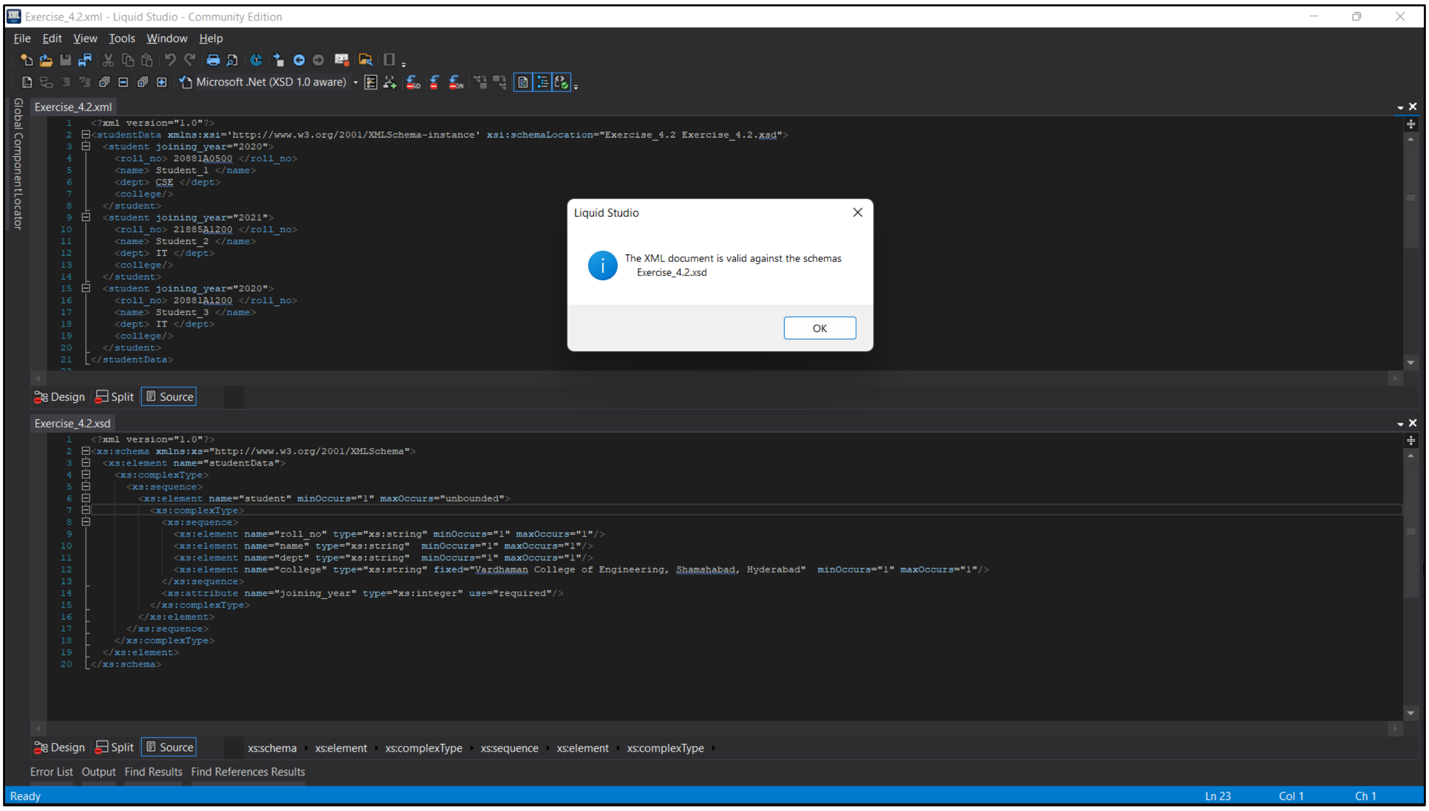
Exercise_4.2.xml

```
<?xml version="1.0"?>
<studentData xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance' xsi:schemaLocation="Exercise_4.2
Exercise_4.2.xsd">
  <student joining_year="2020">
    <roll_no> 20881A0500 </roll_no>
    <name> Student_1 </name>
    <dept> CSE </dept>
    <college/>
  </student>
  <student joining_year="2021">
    <roll_no> 21885A1200 </roll_no>
    <name> Student_2 </name>
    <dept> IT </dept>
    <college/>
  </student>
  <student joining_year="2020">
    <roll_no> 20881A1200 </roll_no>
    <name> Student_3 </name>
    <dept> IT </dept>
    <college/>
  </student>
</studentData>
```

Exercise_4.2.xsd

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="studentData">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="student" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="roll_no" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="name" type="xs:string"  minOccurs="1" maxOccurs="1"/>
              <xs:element name="dept" type="xs:string"  minOccurs="1" maxOccurs="1"/>
              <xs:element name="college" type="xs:string"
                          fixed="Vardhaman College of Engineering, Shamshabad, Hyderabad"
                          minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
            <xs:attribute name="joining_year" type="xs:integer" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Output:



Exercise_4.2.xml - Liquid Studio - Community Edition
File Edit View Tools Window Help
Microsoft .Net (XSD 1.0 aware)
Global ComponentLocator
Exercise_4.2.xml
Design Split Source
Exercise_4.2.xsd
Design Split Source
xs:schema › xs:element › xs:complexType › xs:sequence › xs:element › xs:complexType ›
Error List Output Find Results Find References Results
Ready
Ln 23 Col 1 Ch 1

```
<?xml version="1.0"?>
<studentData xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance' xsi:schemaLocation="Exercise_4.2 Exercise_4.2.xsd">
  <student joining_year="2020">
    <roll_no> 20881A0500 </roll_no>
    <name> Student_1 </name>
    <dept> CSE </dept>
    <college/>
  </student>
  <student joining_year="2021">
    <roll_no> 21885A1200 </roll_no>
    <name> Student_2 </name>
    <dept> IT </dept>
    <college/>
  </student>
  <student joining_year="2020">
    <roll_no> 20881A1200 </roll_no>
    <name> Student_3 </name>
    <dept> IT </dept>
    <college/>
  </student>
</studentData>
```

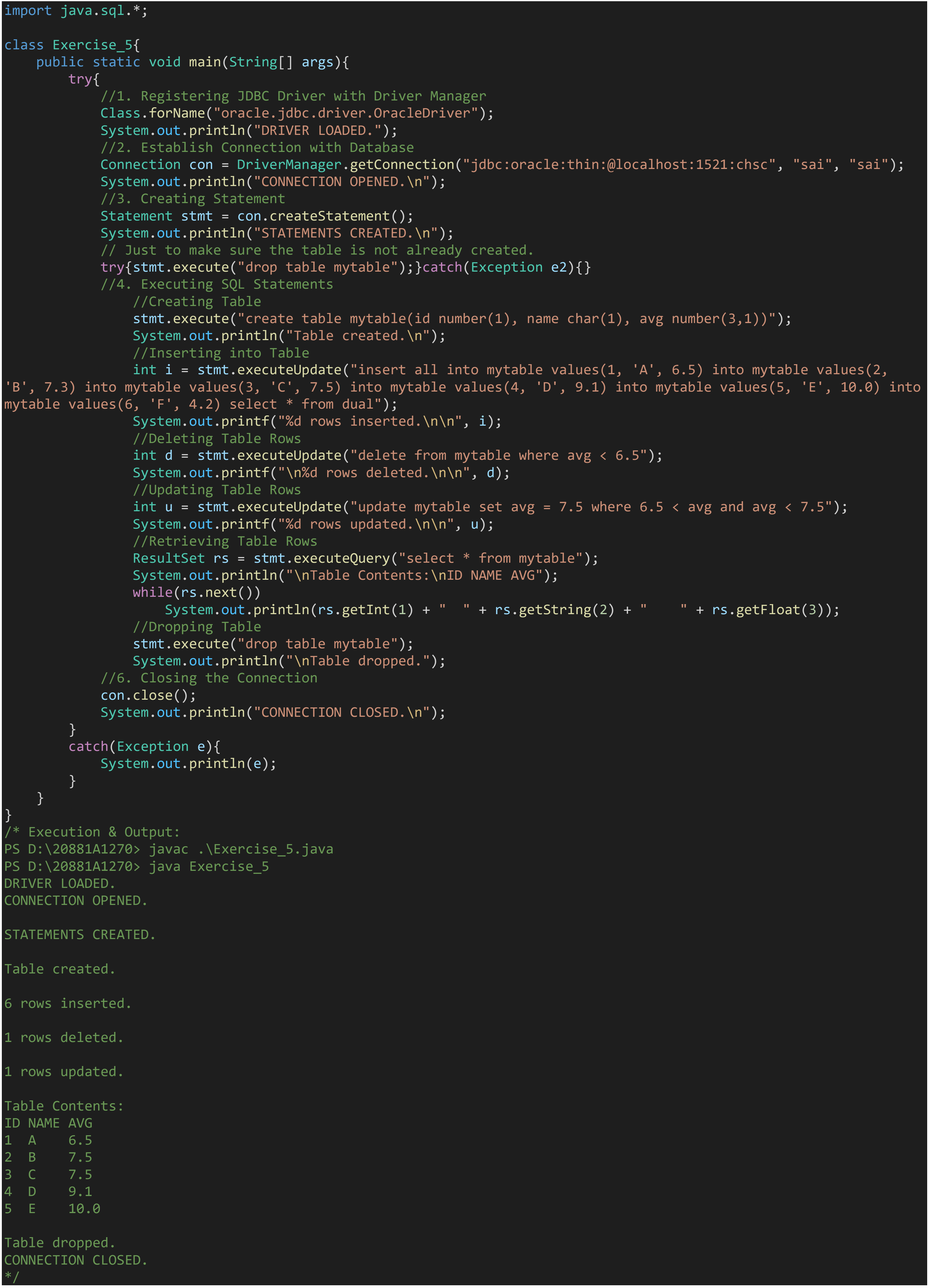
Liquid Studio

The XML document is valid against the schemas
Exercise_4.2.xsd

OK

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="studentData">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="student" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="roll_no" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="name" type="xs:string"  minOccurs="1" maxOccurs="1"/>
              <xs:element name="dept" type="xs:string"  minOccurs="1" maxOccurs="1"/>
              <xs:element name="college" type="xs:string" fixed="Vardhaman College of Engineering, Shamshabad, Hyderabad"  minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
            <xs:attribute name="joining_year" type="xs:integer" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

5 - JDBC Program to create, retrieve, insert, delete and update student data.



```
import java.sql.*;

class Exercise_5{
    public static void main(String[] args){
        try{
            //1. Registering JDBC Driver with Driver Manager
            Class.forName("oracle.jdbc.driver.OracleDriver");
            System.out.println("DRIVER LOADED.");
            //2. Establish Connection with Database
            Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:chsc", "sai", "sai");
            System.out.println("CONNECTION OPENED.\n");
            //3. Creating Statement
            Statement stmt = con.createStatement();
            System.out.println("STATEMENTS CREATED.\n");
            // Just to make sure the table is not already created.
            try{stmt.execute("drop table mytable");}catch(Exception e2){}
            //4. Executing SQL Statements
                //Creating Table
                stmt.execute("create table mytable(id number(1), name char(1), avg number(3,1))");
                System.out.println("Table created.\n");
                //Inserting into Table
                int i = stmt.executeUpdate("insert all into mytable values(1, 'A', 6.5) into mytable values(2,
'B', 7.3) into mytable values(3, 'C', 7.5) into mytable values(4, 'D', 9.1) into mytable values(5, 'E', 10.0) into
mytable values(6, 'F', 4.2) select * from dual");
                System.out.printf("%d rows inserted.\n\n", i);
                //Deleting Table Rows
                int d = stmt.executeUpdate("delete from mytable where avg < 6.5");
                System.out.printf("\n%d rows deleted.\n\n", d);
                //Updating Table Rows
                int u = stmt.executeUpdate("update mytable set avg = 7.5 where 6.5 < avg and avg < 7.5");
                System.out.printf("%d rows updated.\n\n", u);
                //Retrieving Table Rows
                ResultSet rs = stmt.executeQuery("select * from mytable");
                System.out.println("\nTable Contents:\nID NAME AVG");
                while(rs.next())
                    System.out.println(rs.getInt(1) + "  " + rs.getString(2) + "    " + rs.getFloat(3));
                //Dropping Table
                stmt.execute("drop table mytable");
                System.out.println("\nTable dropped.");
            //6. Closing the Connection
            con.close();
            System.out.println("CONNECTION CLOSED.\n");
        }
        catch(Exception e){
            System.out.println(e);
        }
    }
}
/* Execution & Output:
PS D:\20881A1270> javac .\Exercise_5.java
PS D:\20881A1270> java Exercise_5
DRIVER LOADED.
CONNECTION OPENED.

STATEMENTS CREATED.

Table created.

6 rows inserted.

1 rows deleted.

1 rows updated.

Table Contents:
ID NAME AVG
1  A    6.5
2  B    7.5
3  C    7.5
4  D    9.1
5  E    10.0

Table dropped.
CONNECTION CLOSED.
*/
```

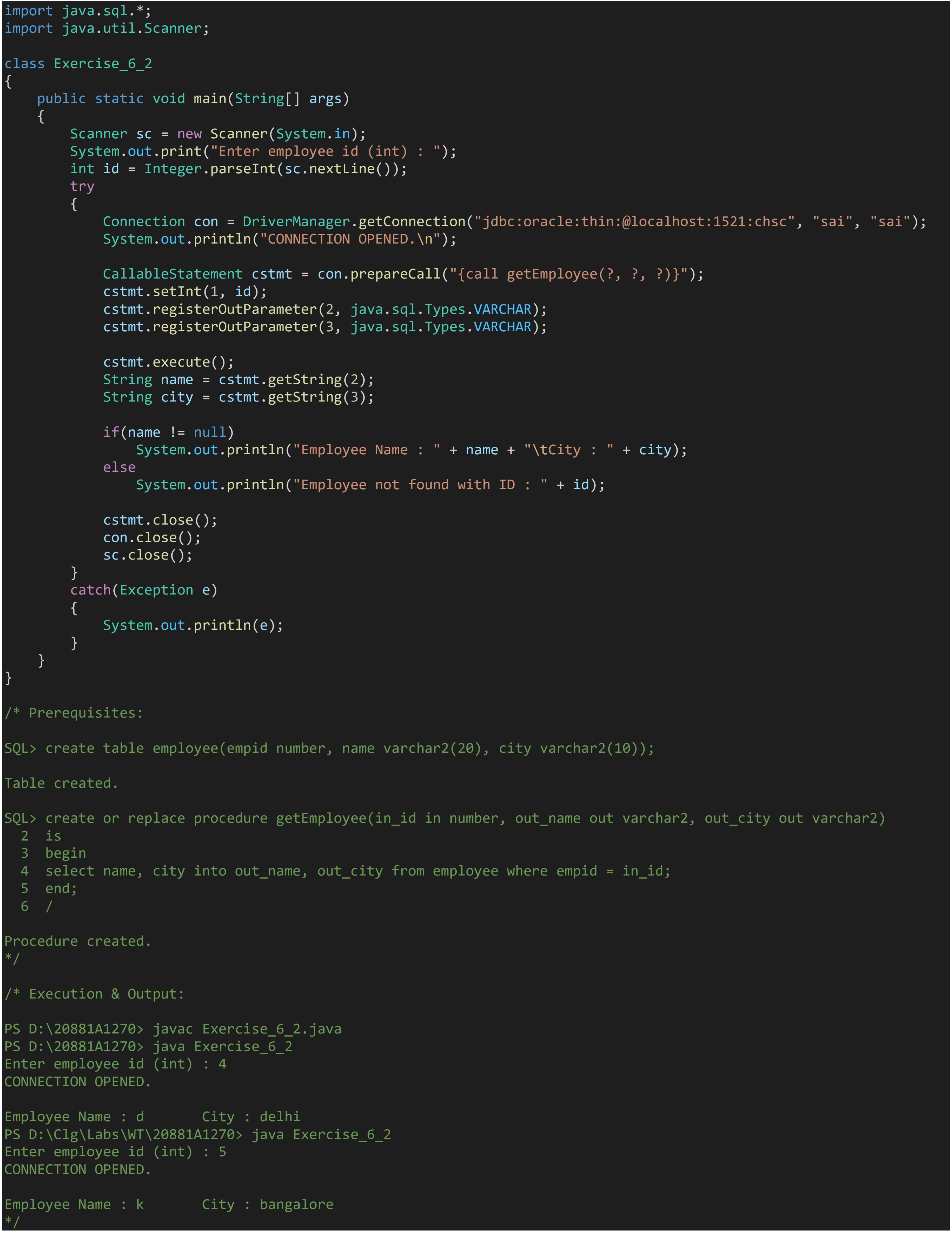
```
 * executeQuery: Returns one ResultSet object.
 * executeUpdate: Returns an integer representing the number of rows affected by the SQL statement. Use this
method if you are using INSERT, DELETE, or UPDATE SQL statements.
 * execute: Returns true if the first object that the query returns is a ResultSet object. Use this method if the
query could return one or more ResultSet objects. Retrieve the ResultSet objects returned from the query by
repeatedly calling Statement.getResultSet.
```

6.1 - JDBC program to create, retrieve, insert, delete and update student data using PreparedStatement.

```
import java.sql.*;
import java.util.Scanner;

class Exercise_6_1{
    public static void main(String[] args){
        try{
            Scanner sc = new Scanner(System.in);
            int k, rno, marks, c = 1;
            String name;
            Class.forName("oracle.jdbc.driver.OracleDriver");
            System.out.println("DRIVER LOADED.");
            Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:chsc", "sai", "sai");
            System.out.println("CONNECTION OPENED.\n");
            PreparedStatement pstmt;
            System.out.println("PREPARED STATEMENT CREATED.\n");
                pstmt = con.prepareStatement("create table mytable(id number(1), name char(1), marks number(3))");
                pstmt.execute();
                System.out.println("Table created.\n");
                while(c != 5)
                {
                    System.out.print("\nChoose an option:\t1.INSERT\t2.DELETE\t3.UPDATE\t4.SELECT\t5.EXIT\nEnter
your option: ");
                    c = sc.nextInt();
                    switch (c)
                    {
                        case 1:
                            pstmt = con.prepareStatement("insert into mytable values(?, ?, ?)");
                            System.out.println("Enter the details of student to be inserted:");
                            System.out.print("rno : "); rno = sc.nextInt(); System.out.print("name : "); name =
sc.next(); System.out.print("marks : "); marks = sc.nextInt();
                            pstmt.setInt(1, rno); pstmt.setString(2, name); pstmt.setInt(3, marks);
                            k = pstmt.executeUpdate();
                            System.out.printf("%d rows inserted.\n\n", k);
                            break;
                        case 2:
                            pstmt = con.prepareStatement("delete from mytable where marks < ?");
                            System.out.print("delete from mytable where marks < "); marks = sc.nextInt();
                            pstmt.setInt(1, marks);
                            k = pstmt.executeUpdate();
                            System.out.printf("\n%d rows deleted.\n\n", k);
                            break;
                        case 3:
                            pstmt = con.prepareStatement("update mytable set marks = ?");
                            System.out.print("update mytable set marks = "); marks = sc.nextInt();
                            pstmt.setInt(1, marks);
                            k = pstmt.executeUpdate();
                            System.out.printf("%d rows updated.\n\n", k);
                            break;
                        case 4:
                            ResultSet rs = pstmt.executeQuery("select * from mytable");
                            System.out.println("\nTable Contents:\nID NAME AVG");
                            while(rs.next())
                                System.out.println(rs.getString(1) + "  " + rs.getString(2) + "    " +
rs.getString(3));
                            break;
                        default:
                            break;
                    }
                }
                pstmt.execute("drop table mytable");
                System.out.println("\nTable dropped.");
            con.close();
            System.out.println("CONNECTION CLOSED.\n");
            sc.close();
        }
        catch(Exception e){
            System.out.println(e);
        }
    }
}
```

```
/* Execution & Output:
PS D:\20881A1270> javac .\Exercise_6.1.java
PS D:\20881A1270> java Exercise_6_1
DRIVER LOADED.
CONNECTION OPENED.

PREPARED STATEMENT CREATED.

Table created.

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 1
Enter the details of student to be inserted:
rno : 1
name : a
marks : 91
1 rows inserted.

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 3
update mytable set marks = 96
1 rows updated.

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 1
Enter the details of student to be inserted:
rno : 2
name : c
marks : 31
1 rows inserted.

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 1
Enter the details of student to be inserted:
rno : 3
name : g
marks : 78
1 rows inserted.

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 2
delete from mytable where marks < 35

1 rows deleted.

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 4

Table Contents:
ID NAME AVG
1  a    96
3  g    78

Choose an option:       1.INSERT        2.DELETE        3.UPDATE        4.SELECT        5.EXIT
Enter your option: 5

Table dropped.
CONNECTION CLOSED.
*/
```

6.2 - JDBC program to retrieve student data using CallableStatement.



```java
import java.sql.*;
import java.util.Scanner;

class Exercise_6_2
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter employee id (int) : ");
        int id = Integer.parseInt(sc.nextLine());
        try
        {
            Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:chsc", "sai", "sai");
            System.out.println("CONNECTION OPENED.\n");

            CallableStatement cstmt = con.prepareCall("{call getEmployee(?, ?, ?)}");
            cstmt.setInt(1, id);
            cstmt.registerOutParameter(2, java.sql.Types.VARCHAR);
            cstmt.registerOutParameter(3, java.sql.Types.VARCHAR);

            cstmt.execute();
            String name = cstmt.getString(2);
            String city = cstmt.getString(3);

            if(name != null)
                System.out.println("Employee Name : " + name + "\tCity : " + city);
            else
                System.out.println("Employee not found with ID : " + id);

            cstmt.close();
            con.close();
            sc.close();
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

/* Prerequisites:

SQL> create table employee(empid number, name varchar2(20), city varchar2(10));

Table created.

SQL> create or replace procedure getEmployee(in_id in number, out_name out varchar2, out_city out varchar2)
  2  is
  3  begin
  4  select name, city into out_name, out_city from employee where empid = in_id;
  5  end;
  6  /

Procedure created.
*/

/* Execution & Output:

PS D:\20881A1270> javac Exercise_6_2.java
PS D:\20881A1270> java Exercise_6_2
Enter employee id (int) : 4
CONNECTION OPENED.

Employee Name : d       City : delhi
PS D:\Clg\Labs\WT\20881A1270> java Exercise_6_2
Enter employee id (int) : 5
CONNECTION OPENED.

Employee Name : k       City : bangalore
*/
```

7.1 - Servlet program to read the parameters from user form (name, contact no) and display using servlet response.

Exercise_7_1\main.html

```html
<html>
    <body style="background-color: beige;">

        <form action="./userform" align="center">
            <br><br><br>
            <h1>User Verification</h1>
            <br><br>
            Username: <input type="text" name="username">
            <br><br>
            Password: <input type="text" name="password">
            <br><br><br>
            <input type="submit" value="Submit">
        </form>
    </body>
 </html>
```

Exercise_7_1\WEB-INF\web.xml

```xml
<web-app>
    <servlet>
        <servlet-name>Exercise_7_1</servlet-name>
        <servlet-class>GetUserDetails</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>Exercise_7_1</servlet-name>
        <url-pattern>/userform</url-pattern>
    </servlet-mapping>
</web-app>
```

Exercise_7_1\WEB-INF\classes\GetUserDetails.java

```java
import javax.servlet.GenericServlet;
import javax.servlet.ServletRequest;
import javax.servlet.ServletResponse;
import javax.servlet.ServletException;
import java.io.PrintWriter;
import java.io.IOException;
public class GetUserDetails extends GenericServlet
{
    public  void service(ServletRequest request, ServletResponse response) throws IOException
    {
        String uname = request.getParameter("username");
        String pwd = request.getParameter("password");
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();
        out.println("<html>");
        out.println("<body  style=\"background-color: beige;\" align =\"center\">");
        out.println("<br><br><h1>Welcome to the site.</h1><br><br>");
        out.println("<h3>Your username is: " + uname + "</h1>");
        out.println("<h3>Your password is: " + pwd + "</h1>");
        out.println("</body>");
        out.println("</html>");
        out.close();
    }
}
```

7.2 - Servlet program to read initialization parameters using ServletConfig.

Exercise_7_2\WEB-INF\web.xml

```
<web-app>
    <servlet>
        <servlet-name> Exercise_7_2  </servlet-name>
        <servlet-class> getInitParameter </servlet-class>
        <init-param>
            <param-name> name </param-name>
            <param-value> CHSC </param-value>
        </init-param>
    </servlet>
    <servlet-mapping>
        <servlet-name> Exercise_7_2 </servlet-name>
        <url-pattern> /param </url-pattern>
    </servlet-mapping>
</web-app>
```

Exercise_7_2\WEB-INF\classes\getInitParameter.java

```
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Enumeration;
import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class getInitParameter extends HttpServlet
{
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, 
IOException
    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        ServletConfig config=getServletConfig();
        String r_name = config.getInitParameter("name");
        out.print("The Retrieved Initialized Parameter: Name = " + r_name);
        out.close();
    }
}
```

8.1 - Servlet program to demonstrate cookie management.

Exercise_8_1\index.html

```html
<html>
    <body bgcolor="beige" align="center">
        <h1>Cookie  Store and Read</h1>
        <form action="./call1" method ="post">
            Username:<input type="text"  name="username"/>
            <br><br>
            Password:<input type="text"  name="password"/>
            <br><br>
            <input type="submit" value="Submit"/>
        </form>
    </body>
</html>
```

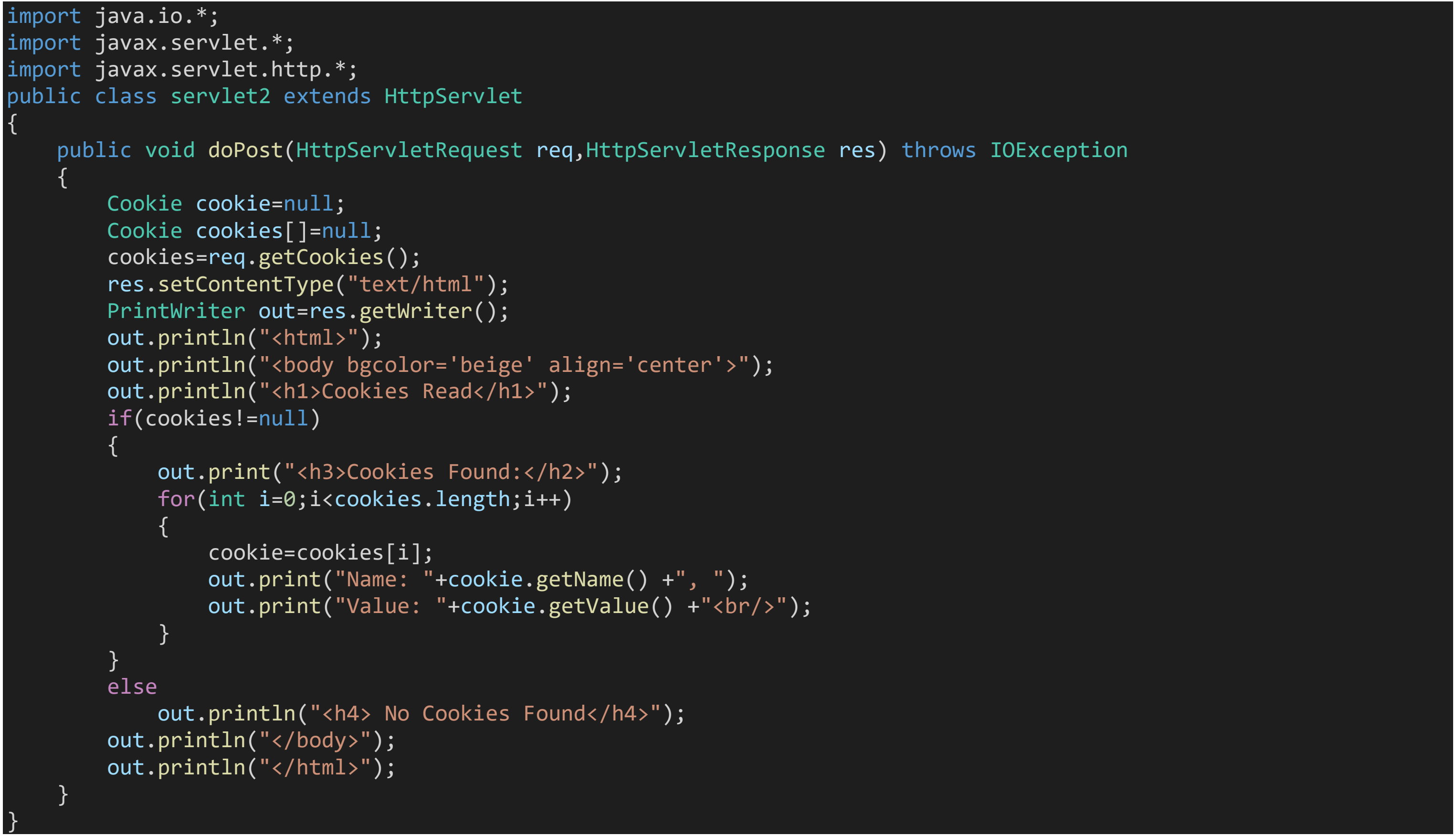
Exercise_8_1\WEB-INF\web.xml

```xml
<web-app>
   <servlet>
        <servlet-name>one</servlet-name>
        <servlet-class>servlet1</servlet-class>
   </servlet>
     <servlet-mapping>
        <servlet-name>one</servlet-name>
        <url-pattern>/call1</url-pattern>
   </servlet-mapping>
 <servlet>
        <servlet-name>two</servlet-name>
        <servlet-class>servlet2</servlet-class>
   </servlet>
     <servlet-mapping>
        <servlet-name>two</servlet-name>
        <url-pattern>/call2</url-pattern>
   </servlet-mapping>
</web-app>
```

Exercise_8_1\WEB-INF\classes\servlet1.java

```java
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class servlet1 extends HttpServlet
{
    public void doPost(HttpServletRequest req,HttpServletResponse res)
    {
        try
        {
            String un = req.getParameter("username"), pwd = req.getParameter("password");
            Cookie c1 = new Cookie("Username",un), c2 = new Cookie("Password",pwd);
            c1.setMaxAge(60*60*24);
            c2.setMaxAge(60*60*24);
            res.addCookie(c1);
            res.addCookie(c2);
            res.setContentType("text/html");
            PrintWriter out=res.getWriter();
            out.print("<html>");
            out.print("<body bgcolor='beige' align='center'>");
            out.print("<h4>Cookies Created</h4>");
            out.print("<form action='./call2' method='post'>");
            out.print("<input type='submit'  value='List Cookies'>");
            out.print("</form>");
            out.print("</body>");
            out.print("</html>");
            out.close();
        }
        catch(Exception e){System.out.println(e);}
    }
}
```

Exercise_8_1\WEB-INF\classes\servlet2.java



```java
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class servlet2 extends HttpServlet
{
    public void doPost(HttpServletRequest req,HttpServletResponse res) throws IOException
    {
        Cookie cookie=null;
        Cookie cookies[]=null;
        cookies=req.getCookies();
        res.setContentType("text/html");
        PrintWriter out=res.getWriter();
        out.println("<html>");
        out.println("<body bgcolor='beige' align='center'>");
        out.println("<h1>Cookies Read</h1>");
        if(cookies!=null)
        {
            out.print("<h3>Cookies Found:</h2>");
            for(int i=0;i<cookies.length;i++)
            {
                cookie=cookies[i];
                out.print("Name: "+cookie.getName() +", ");
                out.print("Value: "+cookie.getValue() +"<br/>");
            }
        }
        else
            out.println("<h4> No Cookies Found</h4>");
        out.println("</body>");
        out.println("</html>");
    }
}
```

8.2 - Servlet program to insert data into student table.

Exercise_8_2\index.html

```html
<html>
    <body bgcolor="beige" align="center">
        <h1>Data Base Connection</h1>
        <form action="./call" method="post">
        First Name:<input type="text"  name="username"/>
        <br><br>
        Roll No<input type="number"  name="rollno"/>
        <br><br>
        <input type="submit" value="Go"/>
        </form>
    </body>
</html>
```

Exercise_8_2\WEB-INF\web.xml

```xml
<web-app>
   <servlet>
      <servlet-name>ServletDB</servlet-name>
      <servlet-class>servlet_DB</servlet-class>
   </servlet>
   <servlet-mapping>
      <servlet-name>ServletDB</servlet-name>
      <url-pattern>/call</url-pattern>
   </servlet-mapping>
</web-app>
```

Exercise_8_2\WEB-INF\classes\servlet_DB.java

```java
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.*;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class servlet_DB extends HttpServlet
{
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException
    {
        response.setContentType("text/html");
        PrintWriter pw=response.getWriter();
        String name = request.getParameter("username");
        int rno = Integer.parseInt(request.getParameter("rollno"));
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:chsc","sai","sai");
            PreparedStatement pstmt=con.prepareStatement("insert into student1 values(?,?)");
            pstmt.setString(1,name);
            pstmt.setInt(2,rno);
            int x=pstmt.executeUpdate();
            if(x==1)
                pw.println("Values Inserted Successfully.");
            else
                pw.println("Values Insertion Failed.");
            con.close();
        }
        catch(Exception e){pw.println("Exception Caught: " + e);}
        pw.close();
    }
}
```

Add WEB-INF\lib\ojdbc.jar

9.1 - JSP Program to print multiplication table.

Exercise_9_1\index.html

```
<html align="center" style="margin-top: 100px; font-family: monospace; font-size: 30px; font-weight: bold; 
background-color: lavender;">
    <body>
        <form action="multiplication_table.jsp">
            Enter a number to print its multiplication table:
            <input type="text" name="n">
            &nbsp;&nbsp;
            <input type="submit" value="Submit">
        </form>
        <br>
    </body>
</html>
```

Exercise_9_1\multiplication_table.jsp

```
<%@ include file="index.html" %>
<%
    int n = Integer.parseInt(request.getParameter("n"));
    for(int i = 1; i <= 10; i++)
    {
        out.println(n + " * " + i + " = " + (n*i));
        %><br><%
    }
%>
```

9.2 - JSP Program to handle the exceptions.

Exercise_9_2\index.html

```
<html align="center" style="margin-top: 200px; font-family: monospace; font-size: large; font-weight: bold;
background-color: lavender;">
    <body>
        <form action="divide.jsp">
            Enter two numbers to perform division:
            <br><br>
            Number 1: <input type="text" name="n1">
            <br>
            Number 2: <input type="text" name="n2">
            <br><br>
            <input type="submit" value="Submit">
        </form>
        <br>
    </body>
</html>
```

Exercise_9_2\divide.jsp

```
<%@ page errorPage="error.jsp" %>
<html align="center" style="margin-top: 200px; font-family: monospace; font-size: large; font-weight: bold;
background-color: lavender;">
    <body>
        <%
            int n1 = Integer.parseInt(request.getParameter("n1"));
            int n2 = Integer.parseInt(request.getParameter("n2"));
            out.println(n1 + " / " + n2 + " = " + n1/n2);
        %>
    </body>
</html>
```

Exercise_9_2\error.jsp

```
<%@ page isErrorPage="true" %>
<html align="center" style="margin-top: 200px; font-family: monospace; font-size: large; font-weight: bold;
background-color: lavender;">
    <body>
        <%
            out.print("Invalid Input.");
            %><br><br><%
            out.print("Exception:  ");
        %>
        <%= exception %>
    </body>
</html>
```

9.3 - JSP Program to retrieve the student data from database.

Exercise_9_3\index.jsp

```
<%@ page import = "java.sql.*" %>
<%@ page import = "java.io.*" %>
<html align="center" style="margin-top: 200px; font-family: monospace; font-size: large; font-weight: bold;
background-color: lavender;">
    <body>
        <%
            try
            {
                Class.forName("oracle.jdbc.driver.OracleDriver");
                Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:chsc", "sai",
"sai");
                Statement stmt = con.createStatement();
                ResultSet rs = stmt.executeQuery("select * from student");
                out.println("ID Name");
                %><br><%
                while(rs.next())
                {
                    out.print(rs.getString(1));
                    %>&nbsp;<%
                    out.println(rs.getString(2));
                    %><br><%
                }
                con.close();
            }
            catch(Exception e){out.println("Exception:  " + e);}
        %>
    </body>
</html>
```

Exercise_9_3\WEB-INF\web.xml

```
<web-app>
</web-app>
```

Add WEB-INF\lib\ojdbc.jar

10.1 - JSP Program to access a java bean component.

Exercise_10_1\index.jsp

```
<form action="welcome.jsp" align="center">
    Username:<input type="text" name="uname"/><br><br>
    Password:<input type="text" name="pwd"/><br><br>
    <input type="submit" value="Go" />
</form>
```

Exercise_10_1\welcome.jsp

```
<jsp:useBean id="obj" class="code.User" />
<jsp:setProperty property="*" name="obj"  />
Hello <jsp:getProperty name="obj" property="uname" />, Your Password is <jsp:getProperty name="obj" property="pwd"
/>
```

Exercise_10_1\WEB-INF\classes\User.java

```
package code;

public class User
{
    private String uname;
    private String pwd;

    public void setUsername(String username){
        uname=username;
    }
    public String getUsername(){
        return uname;
    }
    public void setPassword(String password){
        pwd=password;
    }
    public String getPassword(){
        return pwd;
    }
}
```

10.2 - JSP Program to authenticate the login details. If user is valid forward the control to success.html otherwise forward to fail.html.

Exercise_10_2\index.jsp

```
<form action="./call" align="center" method="post">
    Username:<input type="text" name="uname"/><br><br>
    Password:<input type="password" name="pwd"/><br><br>
    <input type="submit" value="Go" />
</form>
```

Exercise_10_2\WEB-INF\web.xml

```
<web-app>
    <servlet>
        <servlet-name> Exercise_10_2 </servlet-name>
        <servlet-class> code.Servlet </servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name> Exercise_10_2 </servlet-name>
        <url-pattern> /call </url-pattern>
    </servlet-mapping>
</web-app>
```

Exercise_10_2\WEB-INF\classes\Servlet.java

```
package code;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Servlet extends HttpServlet{
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException
    {
        response.setContentType("Text/html");
        PrintWriter out = response.getWriter();
        String uname = request.getParameter("uname"), pwd = request.getParameter("pwd");
        ValidationBean bean = new ValidationBean();
        bean.setUsername(uname);
        bean.setPassword(pwd);
        request.setAttribute("bean", bean);
        boolean validation_status = bean.validate();
        RequestDispatcher d;
        if(validation_status)
            d = request.getRequestDispatcher("success.html");
        else
            d = request.getRequestDispatcher("failure.html");
        d.forward(request, response);
    }
}
```

Exercise_10_2\WEB-INF\classes\ValidationBean.java

```
package code;

public class ValidationBean{
    private String uname, pwd;

    public void setUsername(String username){
        uname = username;
    }
    public String getUsername(){
        return uname;
    }
    public void setPassword(String password){
        pwd = password;
    }
    public String getPassword(){
        return pwd;
    }
    public boolean validate(){
        if(pwd.equals("admin")){
            return true;
        }
        return false;
    }
}
```

Exercise_10_2\success.html

```
<html>
    <body align="center">
        <h1> Login Successful. </h1>
    </body>
</html>
```

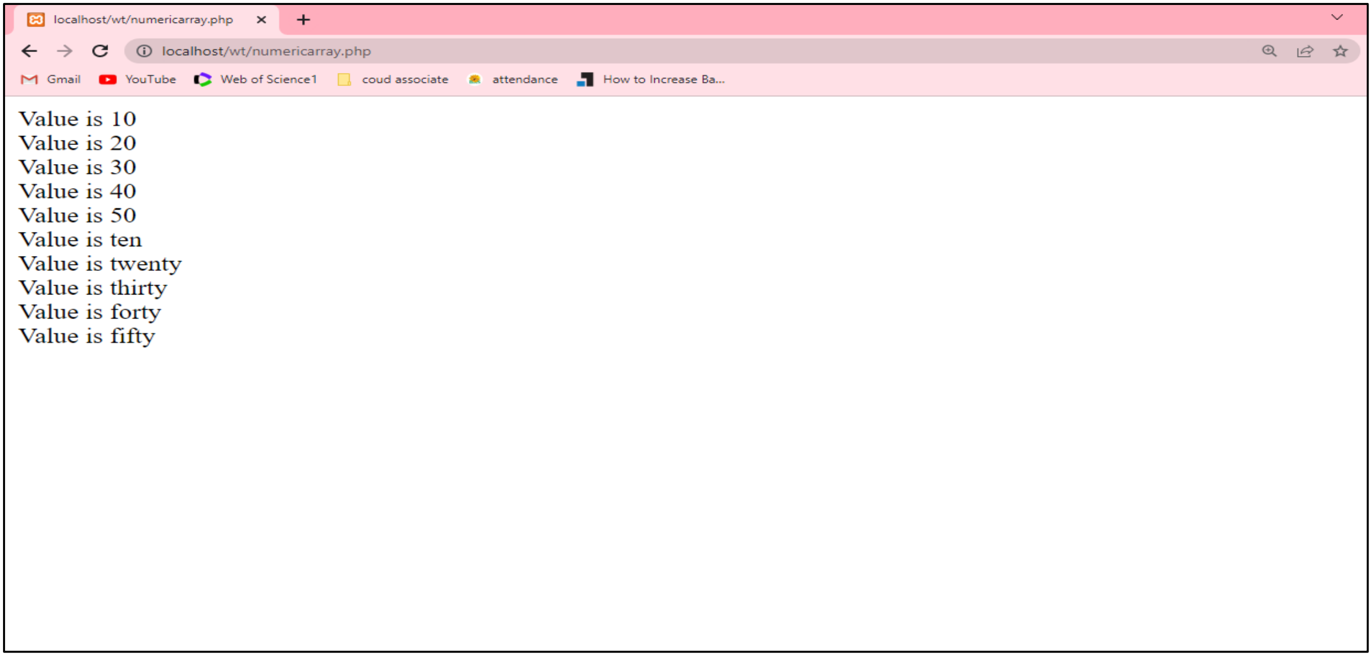
Exercise_10_2\failure.html

```
<html>
    <body align="center">
        <h1> Login Failed. </h1>
    </body>
</html>
```

11.1 - PHP program to work with numeric and associative arrays.

```
<html>
   <body>
        <?php
            /* First method to create array. */
            $numbers = array( 10, 20, 30, 40, 50);
            foreach( $numbers as $value ) {
               echo "Value is $value <br />";
            }
            /* Second method to create array. */
            $numbers[0] = "ten";
            $numbers[1] = "twenty";
            $numbers[2] = "thirty";
            $numbers[3] = "forty";
            $numbers[4] = "fifty";
            foreach( $numbers as $value ) {
               echo "Value is $value <br />";
            }
        ?>
   </body>
</html>
```

Output:

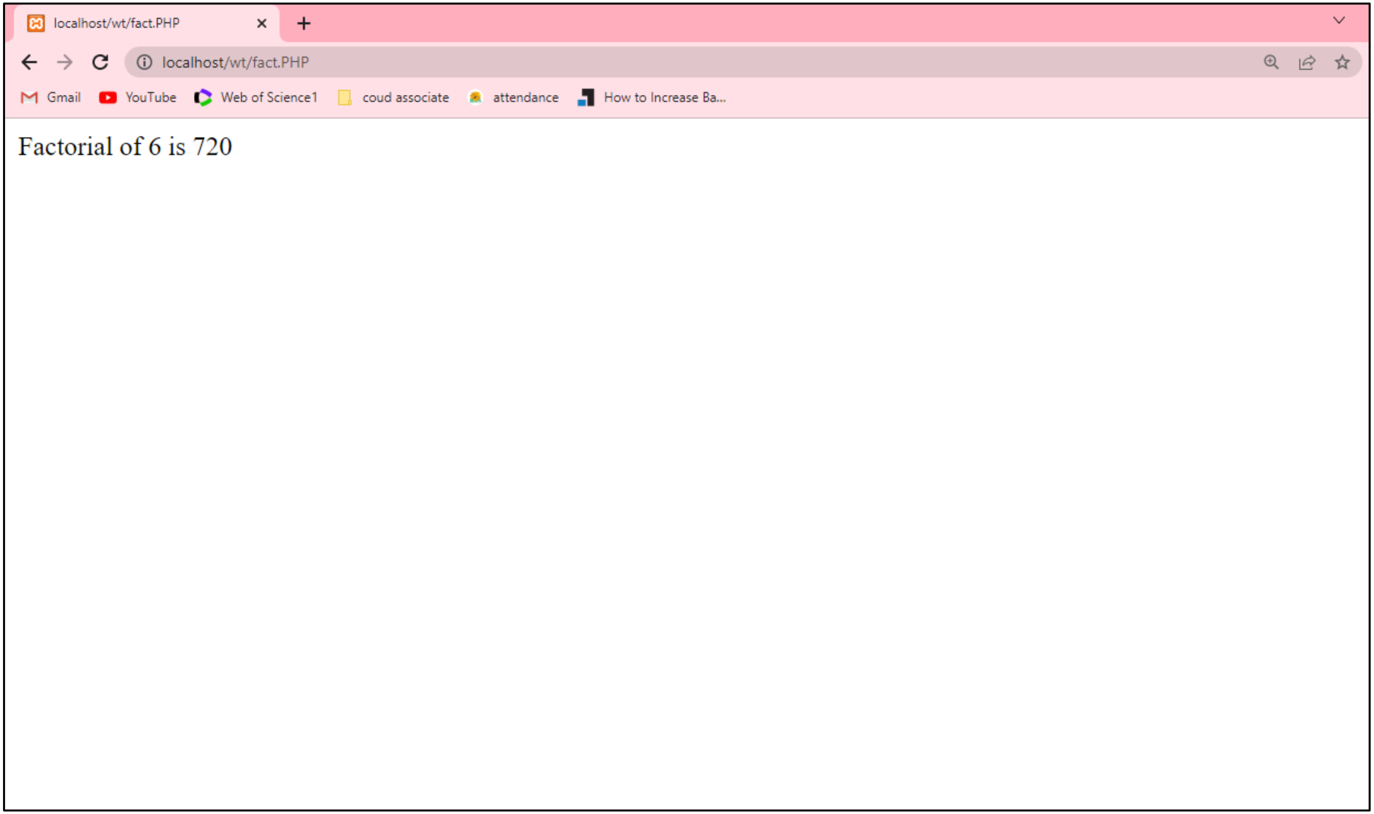


```
Value is 10
Value is 20
Value is 30
Value is 40
Value is 50
Value is ten
Value is twenty
Value is thirty
Value is forty
Value is fifty
```

11.2 - PHP program to find factorial of a number using recursion.

```php
<?php
    function fact ($n)
    {
        if($n <= 1)
        {
            return 1;
        }
        else
        {
            return $n * fact($n - 1);
        }
    }
    echo "Factorial of 6 is " .fact(6);
?>
```

Output:

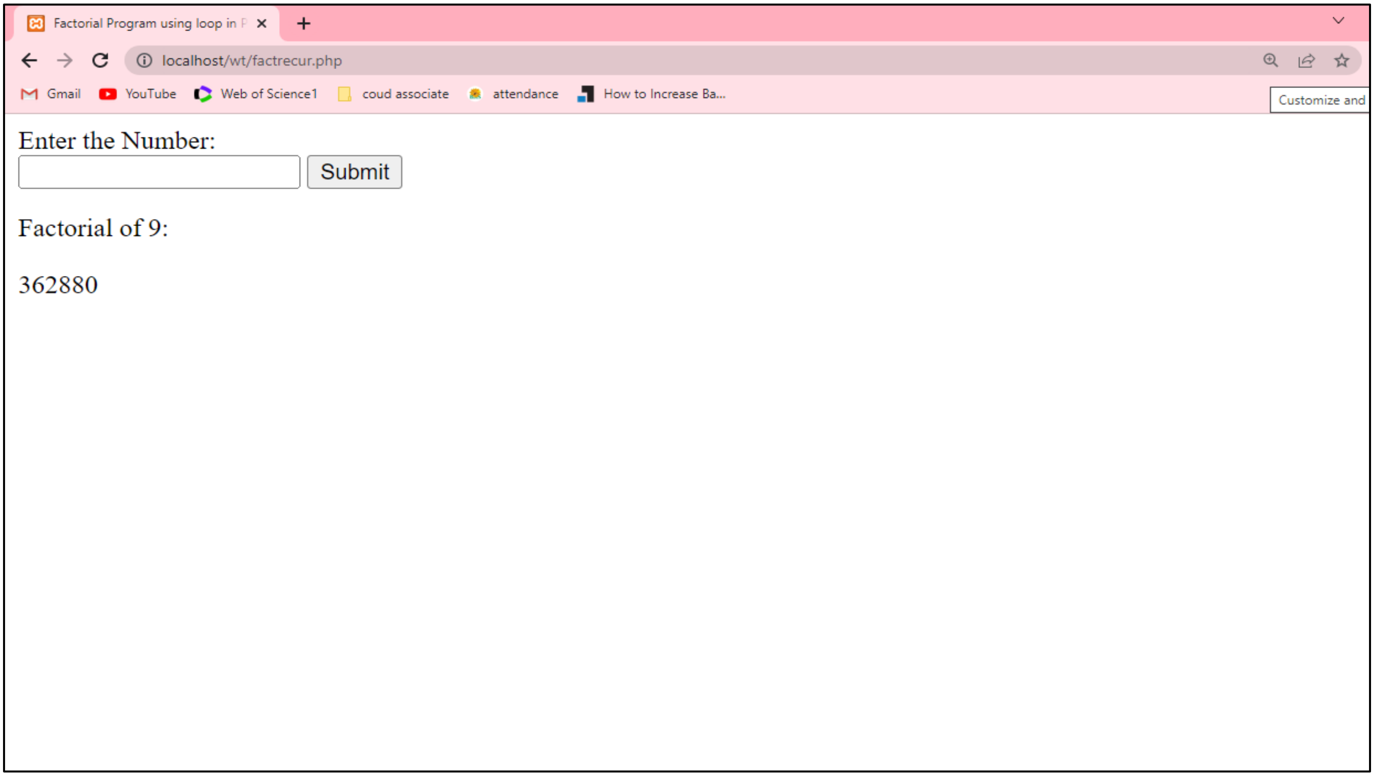


```
Factorial of 6 is 720
```

11.3 - PHP Program to handle form data.

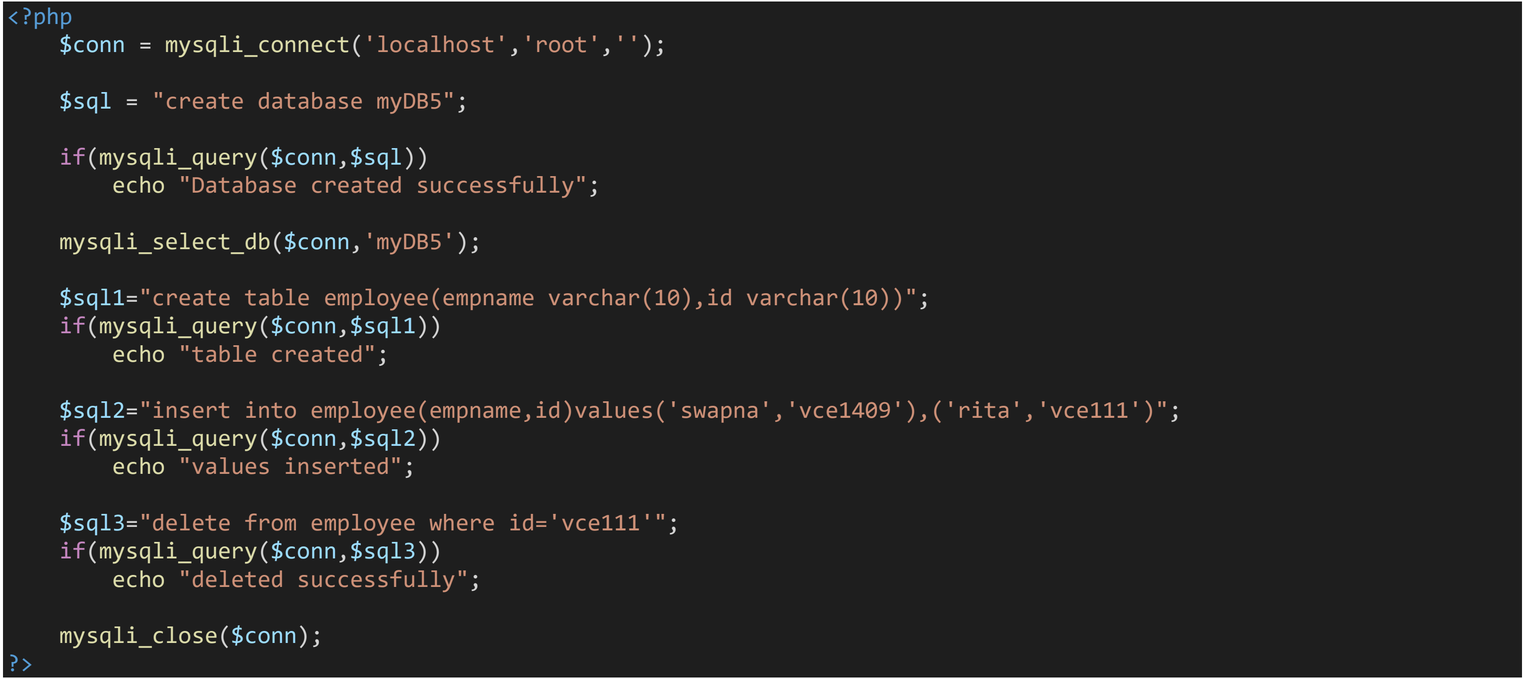
```
<html>
    <head>
        <title>Factorial Program using loop in PHP</title>
    </head>
    <body>
        <form method="post">
            Enter the Number:<br>
            <input type="number" name="number" id="number">
            <input type="submit" name="submit" value="Submit"/>
        </form>
        <?php
            if($_POST) {
                $fact = 1;
                //getting value from input text box 'number'
                $n= $_POST['number'];
                //start loop
                function fact ($n)
                {
                    if($n<= 1)
                    {
                        return 1;
                    }
                    else
                    {
                        return $n * fact($n - 1);
                    }
                }
            }
            echo "Factorial of $n:<br><br>";
            echo fact($n) . "<br>";
        ?>
    </body>
</html>
```

Output:



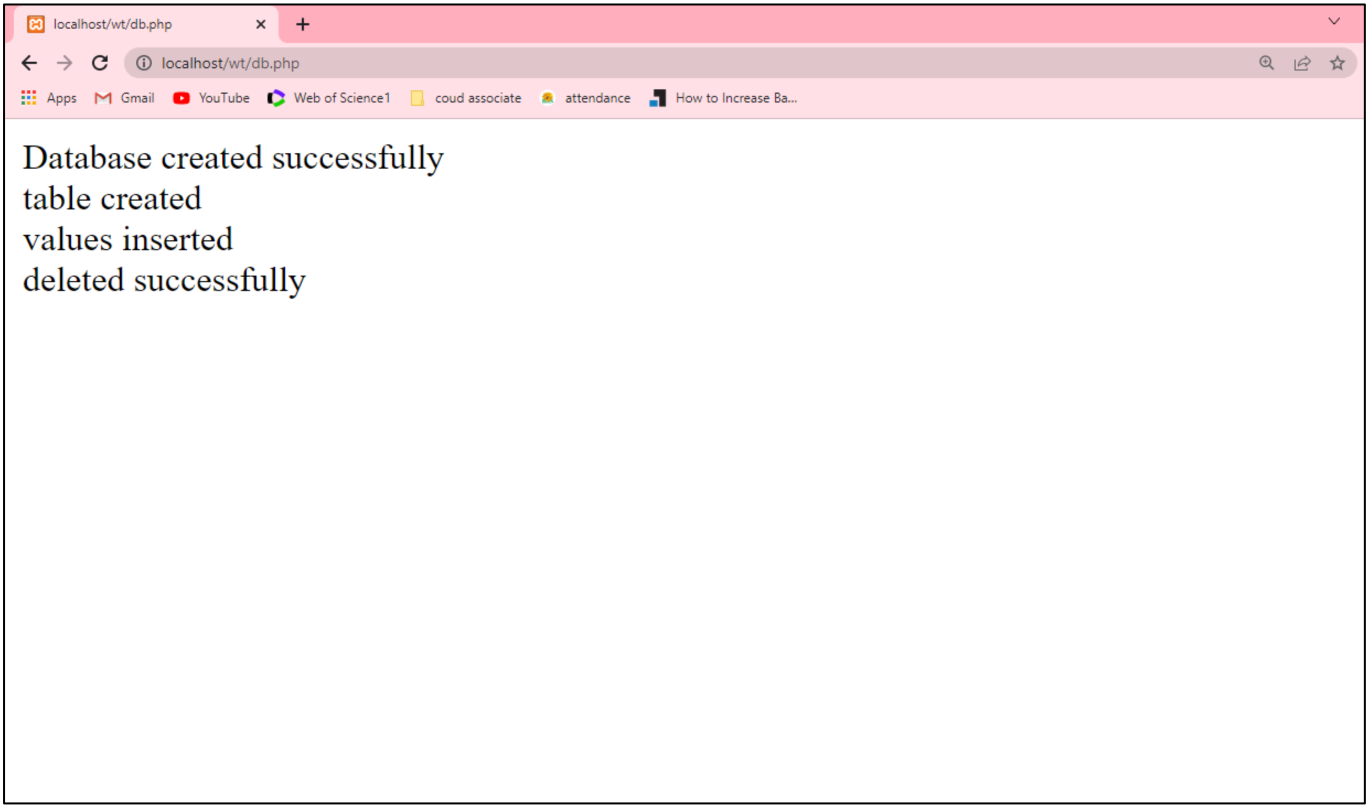
```
Enter the Number:

Factorial of 9:

362880
```

12 - PHP Program to perform database operations like create, insert, update, delete and retrieve on employee data.

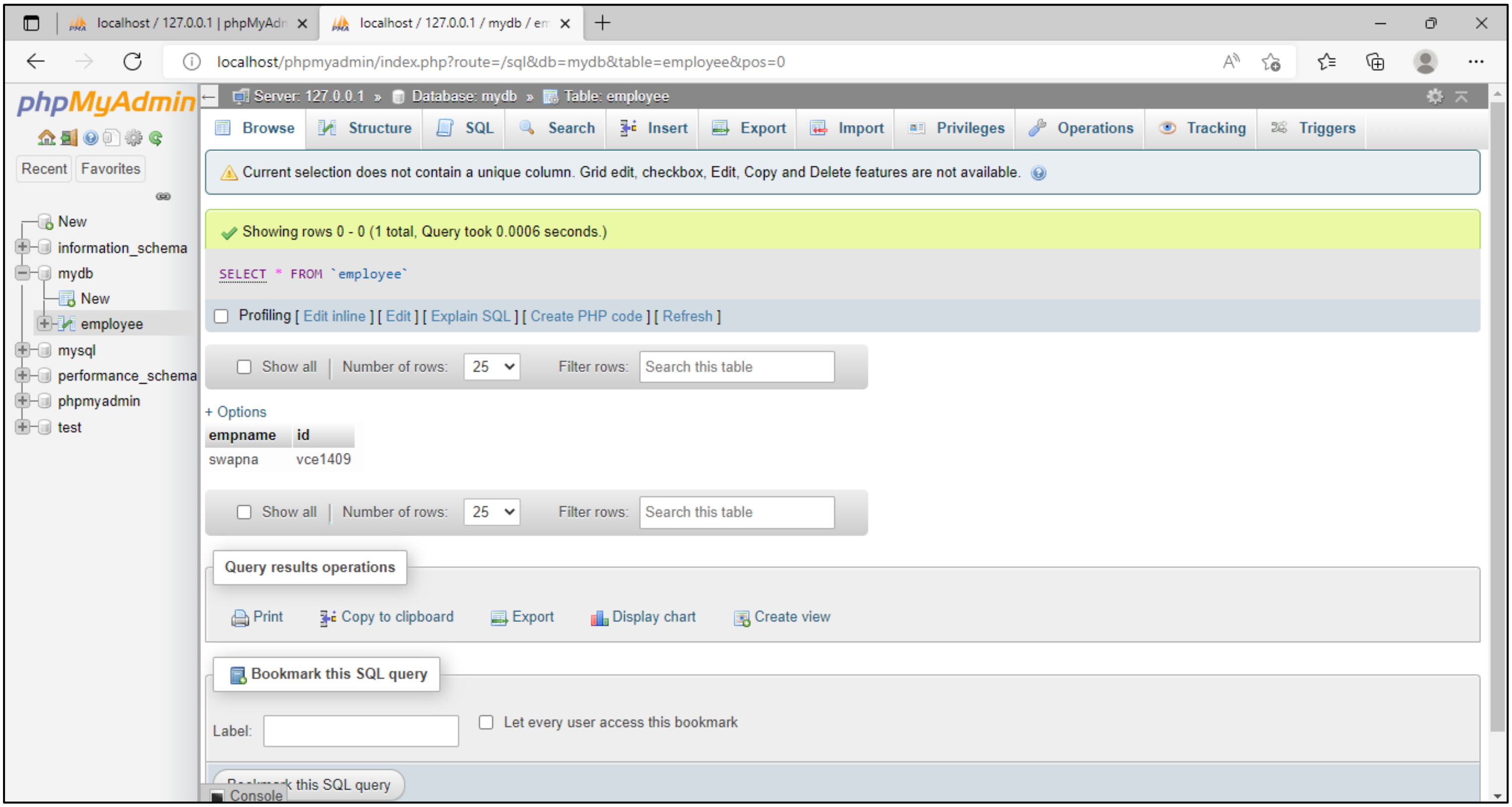


```php
<?php
    $conn = mysqli_connect('localhost','root','');

    $sql = "create database myDB5";

    if(mysqli_query($conn,$sql))
        echo "Database created successfully";

    mysqli_select_db($conn,'myDB5');

    $sql1="create table employee(empname varchar(10),id varchar(10))";
    if(mysqli_query($conn,$sql1))
        echo "table created";

    $sql2="insert into employee(empname,id)values('swapna','vce1409'),('rita','vce111')";
    if(mysqli_query($conn,$sql2))
        echo "values inserted";

    $sql3="delete from employee where id='vce111'";
    if(mysqli_query($conn,$sql3))
        echo "deleted successfully";

    mysqli_close($conn);
?>
```

Output:



```
Database created successfully
table created
values inserted
deleted successfully
```


localhost / 127.0.0.1 | phpMyAdr
localhost / 127.0.0.1 / mydb / en
localhost/phpmyadmin/index.php?route=/sql&db=mydb&table=employee&pos=0
phpMyAdmin
Recent
Favorites
New
information_schema
mydb
New
employee
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: mydb » Table: employee
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Operations
Tracking
Triggers

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 0 (1 total, Query took 0.0006 seconds.)

```
SELECT * FROM `employee`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

+ Options

empname	id
swapna	vce1409

□ Show all | Number of rows: 25 Filter rows: Search this table