

AIM:

To create clickable areas for three fruits in the picture. When you click the fruit, features about the fruits have to be displayed in another web page.

ALGORITHM:

1. Open Netbeans and create a new project.
2. Type the HTML coding for homepage home.html which has an image map using <MAP> tag.
3. Use the area shape attribute to specify the shape of each area in the image map.
4. Create 3 new files for displaying the features of any 3 fruits.
5. Write the codings for all the link webpages.
6. Link those files containing the features of the fruits with home.html.
7. Run the home.html in suitable web browser.
8. Display the output.

PROGRAM:**home.html:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Internet Programming</title>
  </head>
  <body>
    <h1>Fruits</h1>
    
    <map name="Fruits">
      <area shape="rect" coords="381,27,508,127" alt="Image Not Available"
href="Apple.html">
      <area shape="rect" coords="335,396,456,523" alt="Image Not Available"
href="Orange.html">
      <area shape="rect" coords="37,27,223,292" alt="Image Not Available"
href="Green grapes.html">
    </map>
  </body>
</html>
```

Apple.html:

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

```

    <title>Features of Fruits</title>
</head>
<body>
    <h1>Features of Apple....!</h1>
    <p>
        Apple fruit is oval or pear in shape. <br>
        They are incredibly nutritious fruit. <br>
        They are rich in fibre and antioxidants. <br>
        Its outer peel comes in different hues and colours depending upon the
cultivar type. <br>
        Internally, its crispy, juicy pulp is off-white to cream and features a mix of
mildly sweet and tart flavour. <br>
        Its seeds are bitter in taste, and therefore, inedible. <br>
    </p>
</body>
</html>

```

Orange.html:

```

<html>
    <head>
        <title>Features of Fruits</title>
    </head>
    <body>
        <h1>Features of Orange....!</h1>
        <p>
            Orange trees have dark green shiny leaves and small white flowers with five
petals. <br>
            The flowers smell very sweet which attracts many bees. <br>
            An orange has a tough shiny orange skin. <br>
            Inside, the fruit is divided into “segments”, which have thin tough skins that
hold together many little sections with juice inside. <br>
            The vitamin C in oranges helps your body in lots of ways:
            <pre>
                Protects your cells from damage. <br>
                Helps your body make collagen, a protein that heals wounds and gives
you smoother skin. <br>
                Makes it easier to absorb iron to fight anemia.
            </pre>
        </p>
    </body>
</html>

```

Green Grapes.html:

```

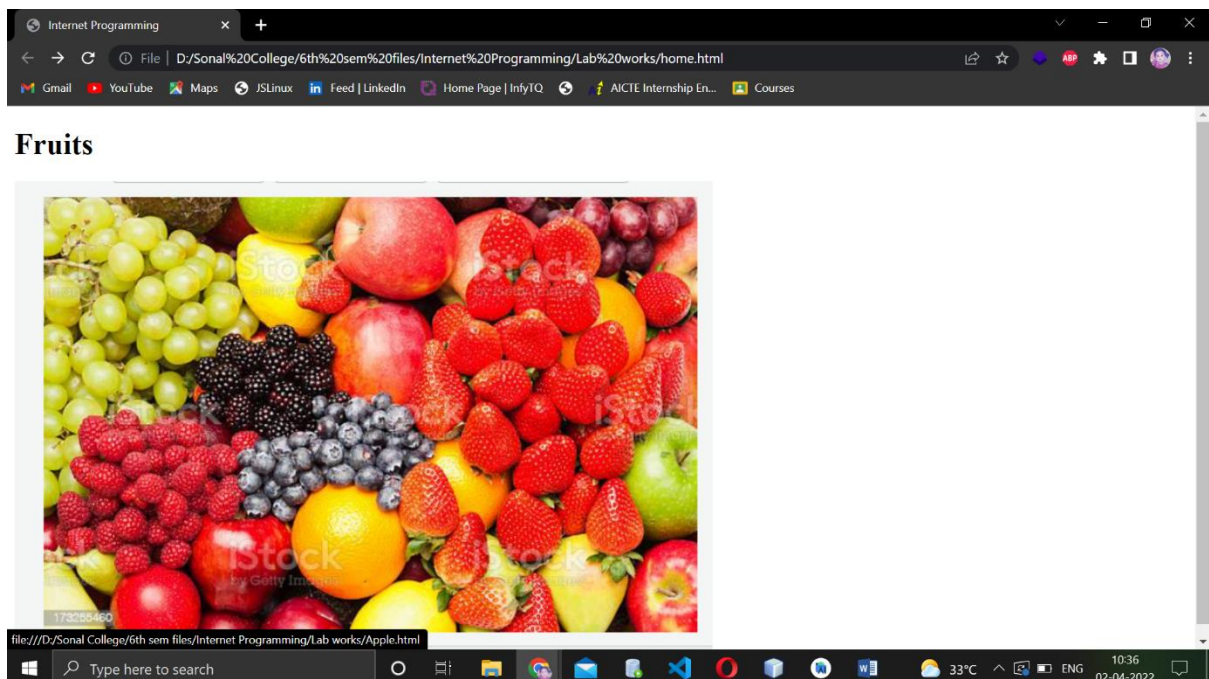
<!DOCTYPE html>
<html>
    <head>
        <title>Features of Fruits</title>
    </head>
    <body>
        <h1>Features of Green Grapes....!</h1>
    </body>
</html>

```

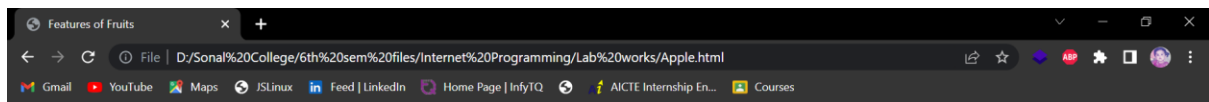
```
<p>
  Grapes Are Actually Berries. <br>
  Table and Wine Grapes Are Different. <br>
  Grapes Have Been Around for 65 Million Years. <br>
  People Have Been Cultivating Grapes for 8,000 Years. <br>
  There Are 8,000 Different Varieties of Grapes. <br>
  Grape Skins Naturally Host Yeast. <br>
  Some properties of grapes are:
  <pre>
    Helps your immune system. <br>
    Prevents cancer. <br>
    Lowers blood pressure. <br>
    Protects against heart disease. <br>
    Reduces high cholesterol. <br>
    Protects against diabetes. <br>
    Helps maintain brain health. <br>
    Improves bone health. <br>
  </pre>
</p>
</body>
</html>
```

OUTPUT:

Home.html:



Apple.html:

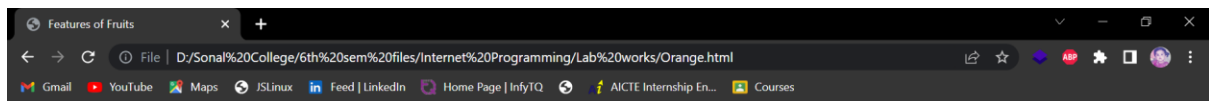


Features of Apples....!

Apple fruit is oval or pear in shape.
They are incredibly nutritious fruit.
They are rich in fibre and antioxidants.
Its outer peel comes in different hues and colors depending upon the cultivar type.
Internally, its crispy, juicy pulp is off-white to cream and features a mix of mildly sweet and tart flavour.
Its seeds are bitter in taste, and therefore, inedible.



Orange.html:



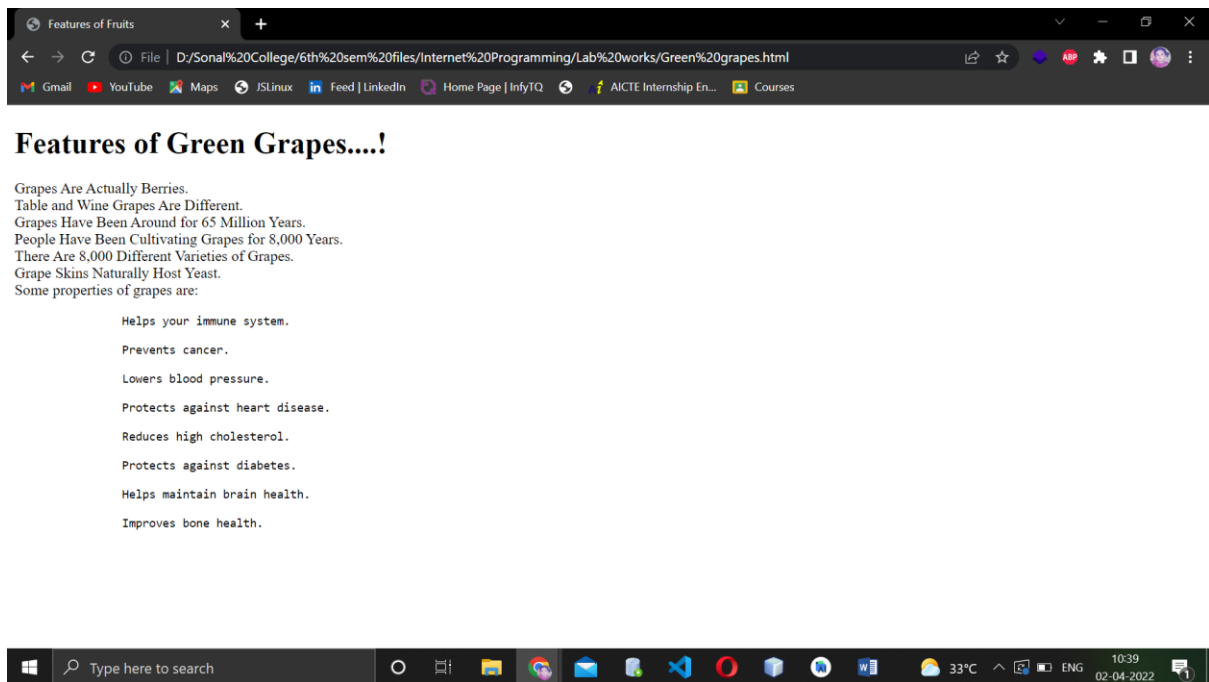
Features of Orange....!

Orange trees have dark green shiny leaves and small white flowers with five petals.
The flowers smell very sweet which attracts many bees.
An orange has a tough shiny orange skin.
Inside, the fruit is divided into "segments", which have thin tough skins that hold together many little sections with juice inside.
The vitamin C in oranges helps your body in lots of ways:

- Protects your cells from damage.
- Helps your body make collagen, a protein that heals wounds and gives you smoother skin.
- Makes it easier to absorb iron to fight anemia.



Green grapes.html:



Observation	
Record	
Total	

RESULT:

Thus, clickable areas for three fruits is created and features about the fruits have been displayed in another web page when clicking the fruit successfully.

AIM:

To create an interactive webpage for Library management system using images, hyperlinks, lists, tables, audio, videos and forms.

ALGORITHM:

1. Open Netbeans and create a new project.
2. Type the HTML coding for homepage index.html.
3. Use the forms for authentication purpose.
4. Create a background image to attach image in the background.
5. Create hyperlinks using <a> tag to search the books according to the genre.
6. Create lists using and tags.
7. Attach related images also if needed.
8. Create separate tables for books and student details using <table> tag.
9. Attach some audio and video using <audio> and <video> tags.
10. Run the index.html in suitable web browser.
11. Display the output.

PROGRAM:

index.html:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Internet Programming</title>
    <style>
      body{
        background-image: url('bg.jpg');
        background-repeat: no-repeat;
        background-attachment: fixed;
        background-size: 100%;
      }
      table,th,td{
        border: 1px solid black;
        border-collapse: collapse;
      }
      th,td{
        background-color: aquamarine;
      }
    </style>
  </head>
  <body>
    <h1>Library Management System</h1>
```


Fantasy

Comics

<h2>Few books available in the library</h2>

<h2>Books Details</h2>

<table style="width:50%">

<tr>

<th>S.NO</th>

<th>Book Name</th>

<th>Author Name</th>

</tr>

<tr>

<td>1</td>

<td>2 States</td>

<td>Chetan Bhagat</td>

</tr>

<tr>

<td>2</td>

<td>The notebook</td>

<td>Nicholas Spark</td>

</tr>

<tr>

<td>3</td>

<td>Harry Potter and the Philosopher's Stone</td>


```

        <td>J K Rowling</td>
    </tr>
    <tr>
        <td>4</td>
        <td>Dear Girls</td>
        <td>Ali Wong</td>
    </tr>
    <tr>
        <td>5</td>
        <td>Alice's Adventures in Wonderland</td>
        <td>Lewis Carroll</td>
    </tr>
</table>

```

<h2>Students Details</h2>

```

<table style="width:100%">
    <tr>
        <th>S.NO</th>
        <th>Student Name</th>
        <th>Department</th>
        <th>Roll No</th>
        <th>Book Borrowed</th>
        <th>Issue date</th>
        <th>Return date</th>
    </tr>
    <tr>
        <th>1</th>
        <th>Sonal L R</th>
        <th>CSE</th>
        <th>19cse087</th>
        <th>2 States</th>
        <th>19/07/2021</th>
        <th>21/08/2021</th>
    </tr>
    <tr>
        <th>2</th>
        <th>Indhumathi K V</th>
        <th>ECE</th>
        <th>19ece072</th>
        <th>Tinkle Magazine</th>
        <th>15/03/2022</th>
        <th>24/03/2022</th>
    </tr>
    <tr>
        <th>3</th>
        <th>Jeya Roshini S</th>
        <th>IT</th>
        <th>19it032</th>
        <th>Around the world in 80 days</th>
        <th>12/08/2020</th>
        <th>23/10/2020</th>
    </tr>

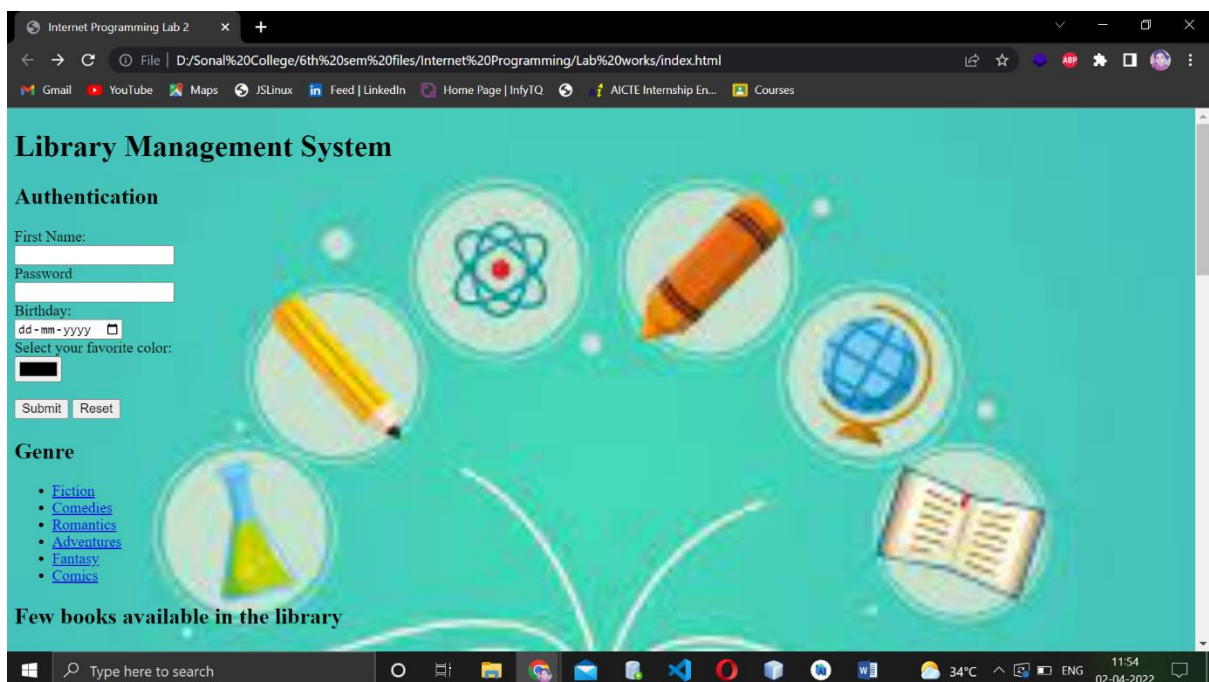
```

```

</tr>
<tr>
  <th>4</th>
  <th>Varsha</th>
  <th>Civil</th>
  <th>19civ059</th>
  <th>Treasure Island</th>
  <th>Issue date</th>
  <th>Return date</th>
</tr>
</table><br>
<audio controls autoplay>
  <source src="htmltut.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio><br>
<video width="500" height="500" controls>
  <source src="song.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>
</body>
</html>

```

OUTPUT:



Internet Programming Lab 2

File | D:/Sonal%20College/6th%20sem%20files/Internet%20Programming/Lab%20works/index.html

Gmail YouTube Maps JSLinux Feed | LinkedIn Home Page | InfyTQ AICTE Internship En... Courses

Few books available in the library

Books Details

S.NO	Book Name	Author Name
1	2 States	Chetan Bhagat

Type here to search

34°C 11:54 02-04-2022

Internet Programming Lab 2

File | D:/Sonal%20College/6th%20sem%20files/Internet%20Programming/Lab%20works/index.html

Gmail YouTube Maps JSLinux Feed | LinkedIn Home Page | InfyTQ AICTE Internship En... Courses

Books Details

S.NO	Book Name	Author Name
1	2 States	Chetan Bhagat
2	The notebook	Nicholas Spark
3	Harry Potter and the Philosopher's Stone	J K Rowling
4	Dear Girls	Ali Wong
5	Alice's Adventures in Wonderland	Lewis Carroll

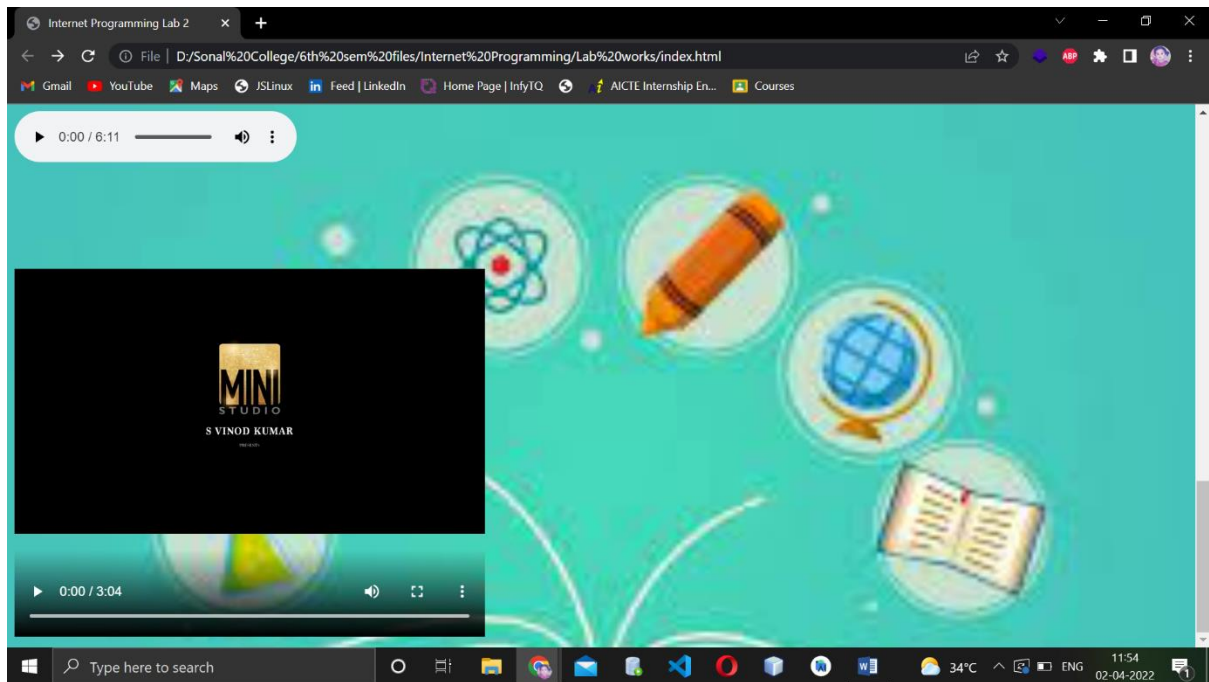
Students Details

S.NO	Student Name	Department	Roll No	Book Borrowed	Issue date	Return date
1	Sonal L R	CSE	19esc087	2 States	19/07/2021	21/08/2021
2	Indhumathi K V	ECE	19ece072	Tinkle Magazine	15/03/2022	24/03/2022
3	Jeya Roshini S	IT	19it032	Around the world in 80 days	12/08/2020	23/10/2020
4	Varsha	Civil	19civ059	Treasure Island	Issue date	Return date

0:00 / 6:11

Type here to search

34°C 11:54 02-04-2022



Observation	
Record	
Total	

RESULT:

Thus, an interactive webpage for Library management system using images, hyperlinks, lists, tables, audio, videos and forms was created successfully.

QUESTION:

Create an interactive webpage for Library management system using different types of stylesheets (use inline, internal and external CSS). The webpage should display

1. Name the website like Connemora central library
2. Website name have to be displayed at the top center of the webpage.
Apply some text effects to the website name.
3. At the top right corner of the webpage, create a link for Signup/Login, when you click this login form has to open.
4. Display some description about your webpage. Set suitable font, text and colour properties for the content
5. List of streams of books available have to be displayed in table form using images. Apply border properties for the table.
6. Fix an image in the webpage at a particular position. When you scroll the page, that image must be visible. When you click the image, it should go to the library registration form. The registration form has to displayed within the webpage itself in the right side. Form has to be displayed within a border.
7. In the left side of the webpage at a particular part use animation and transformations to display the good books that are available in the library. The images of the books have to change at a particular interval

AIM:

To create an interactive webpage for Library management system using different types of stylesheets (use inline, internal and external CSS).

ALGORITHM:

9. Open Netbeans and create a new project.
10. Type the HTML coding for homepage index.html.
11. The top center of the page will be the title which is given by shadow effect.
12. A small description about the library is given with some styling.
13. Left corner are some random images of the books that are available in the library.
14. Right corner is a login button which redirects to the login page.
15. And a logo will be present at the same place while scrolling.
16. A registration form is present in the front page to register for booking a new book.
17. A table is present which shows the contents about the books in the library.
18. Display the output.

PROGRAM:

index.html:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Connemora Central Library</title>
    <link rel="stylesheet" type="text/css" href="css/style.css">
    <style>
      h1{
        text-shadow: 2px 2px 5px black;
        text-align: center;
        background-color: rgb(41, 41, 229);
      }
      img {
        display: block;
        margin-left: auto;
        margin-right: auto;
        margin-top: 10%;
      }
      h3{
        color: black;
      }
      table,th{
        color: black;
      }
    </style>
  </head>
  <body>
    <div class="heading">
      <h1 class="name">Connemora Central Library</h1>
      <button type="button" class="sign"
onclick="window.location.href='login.html'">Sign Up / Login</button>
    </div>
    <div class="content">
      <div class="slider">
        <img src="" alt="" id="image">
        <script type="text/javascript">
          let image=document.getElementById('image');
          let images=['2 States.jpg','The notebook.jpg','Harry potter.jpg','Treasure
island.jpg','The time machine.jpg'];
          setInterval(function(){
            let random=Math.floor(Math.random()*5);
            image.src=images[random];
          },1000);
        </script>
      </div>
      <div class="about">
        <h3>Connemora Library</h3>
      </div>
    </div>
  </body>
</html>
```

<p>The Connemara Public Library at Egmore in Chennai, Tamil Nadu, India, is one of the four National Depository

Libraries which receive a copy of all books, newspapers and periodicals published in India.

Established in 1896, the library is a repository of century-old publications, wherein lie some of the most

respected works and collections in the history of the country. It also serves as a depository library for the

United Nations.[citation needed] It is located in the Government Museum Complex on Pantheon Road, Egmore, which

also houses the Government Museum and the National Art Gallery.</p>

<table>

<tr>

<th style="width: 10%;">S.no</th>

<th style="width: 30%;">Books</th>

<th style="width: 30%;">Authors</th>

</tr>

<tr>

<td>1.</td>

<td></td>

<td>Chetan Bhagat</td>

</tr>

<tr>

<td>2.</td>

<td></td>

<td>Mark Twain</td>

</tr>

<tr>

<td>3.</td>

<td></td>

<td>Lewis Carroll</td>

</tr>

<tr>

<td>4.</td>

<td></td>

<td>Anne Frank</td>

</tr>

</table>

</div>

<div class="logo">

<iframe src="iframe.html" name="register"></iframe>

</div>

</div>

</body>

</html>

style.css:

```
@import
url('https://fonts.googleapis.com/css2?family=Libre+Baskerville&display=swap');
.forms{
  text-align: center;
  border: solid 2px black;
  background-color: grey;
  color: rgb(41, 41, 229);
  margin: 8% 30% 8% 30%;
  font-size: xx-large;
  padding: 20px 0px 20px 0px;
}

.submit{
  color: grey;
  background-color: rgb(41, 41, 229);
  font-size: xx-large;
}

.heading{
  display: inline;
  background-color: rgb(41, 41, 229);
}

.name{
  padding-left: 20%;
  padding-right: 20%;
  font-size: xxx-large;
  color: black;
}

.sign{
  background-color: rgb(41, 41, 229);
  color: black;
  float: right;
  height: 40px;
  margin-top: 10px;
  font-size: large;
}

.sign:hover span{
  color: black;
  background-color: rgb(41, 41, 229);
}

.content{
  display: flex;
}
```



```

.logo img{
  height: 100px;
  width: 100px;
  position: fixed;
  padding-left: 30px;
  padding-top: 0px;
  margin-top: 20px;
}

.slider img{
  width: 150px;
  height: 200px;
  margin: 15px;
}

.about{
  width: 60%;
  height: auto;
  padding: 0px 30px 20px 30px;
  background-color: white;
  color: rgb(41, 41, 229);
  margin-top: 20px;
  border: solid 3px black;
}

.about h3{
  font-size: x-large;
  font-style: italic;
}

.about img{
  width: 100px;
  height: 150px;
}

table,th,tr,td{
  padding: 20px;
  border: solid 2px whitesmoke;
  border-collapse: collapse;
}

iframe{
  margin: 50% 0% 10% 15%;
  width: 120%;
  height: 240px;
}

.cregister{
  padding: 20px 20px 60px 0px;
}

```

```
text-align: center;
color: aquamarine;
background-color: rgb(90, 73, 248);
}
```

```
.csubmit{
background-color: aquamarine;
color: rgb(90, 73, 248);
}
```

login.html:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sign In form</title>
    <link rel="stylesheet" href="css/style.css">
    <style>
      h1{
        text-align: center;
      }
    </style>
  </head>
  <body>
    <h1>SIGN IN</h1>
    <div class="forms">
      <form action="#">
        <label for="fname">First Name :</label>
        <input type="text" name="fname" id="fname"><br><br>
        <label for="lname">Last Name :</label>
        <input type="text" name="lname" id="lname"><br><br>
        <label for="number">Roll No :</label>
        <input type="text" name="rollno" id="rollno"><br><br>
        <button class="submit">Submit</button>
      </form>
    </div>
  </body>
</html>
```

register.html:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Reistration</title>
    <link rel="stylesheet" href="css/style.css">
  </head>
  <body>
    <div class="cregister">
      <form action="#">
        <label for="fname">First Name :</label>
        <input type="text" name="fname" id="fname"><br><br>
```

```

<label for="lname">Last Name :</label>
<input type="text" name="lname" id="lname"><br><br>
<label for="number">Roll No :</label>
<input type="text" name="rollno" id="rollno"><br><br>
<label for="cname">Book Name :</label>
<input type="text" name="cname" id="cname"><br><br>
<button class="csubmit">Submit</button>
</form>
</div>
</body>
</html>

```

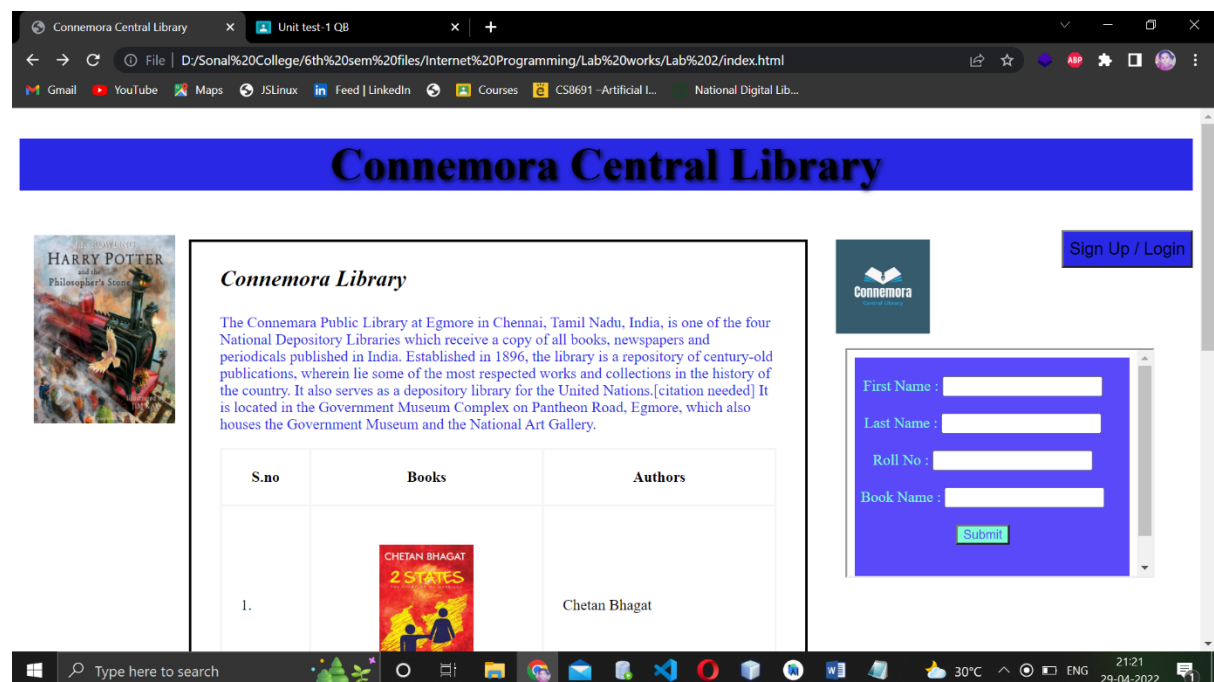
iframe.html:

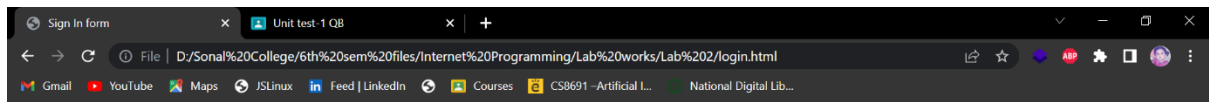
```

<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
  <body>
    <h2>Readers are good Leaders.</h2>
  </body>
</html>

```

OUTPUT:





SIGN IN

First Name :

Last Name :

Roll No :



Observation	
Record	
Total	

RESULT:

Thus, an interactive webpage for Library management system using different types of stylesheets (use inline, internal and external CSS) were created successfully.

AIM:

To write the java script functions for the given questions.

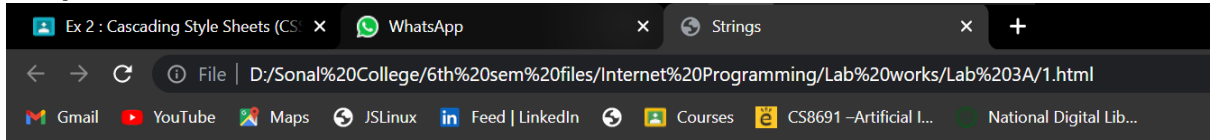
QUESTIONS:**1. Write a java script function that print all combination of strings.****Algorithm:**

19. Open Netbeans and create a new project.
20. For each letter from input start position to end of input string,
 - a. Append the letter to the output string
 - b. Print letters in output string
21. If the current letter isn't the last in the input string, generate the remaining combinations starting at next position with starting at next letter beyond the letter just selected.
22. Delete the last character of the output string.
23. Display the output.

Program:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Strings</title>
  </head>
  <body>
    <script>
      function combinator (s) {
        list_of_strings =[];
        for(i=0; i<s.length; i++) {
          for(j=i+1;j<s.length+1;j++) {
            list_of_strings.push(s.slice(i, j));
          }
        }
        return list_of_strings;
      }
      document.write(combinator("Sonal"));
    </script>
  </body>
</html>
```

Output:



S,So,Son,Sona,Sonal,o,on,ona,onai,n,na,nal,a,al,l

2. Write a java script function that accepts a sentence as input and then it should list out all 3 letter words in it.

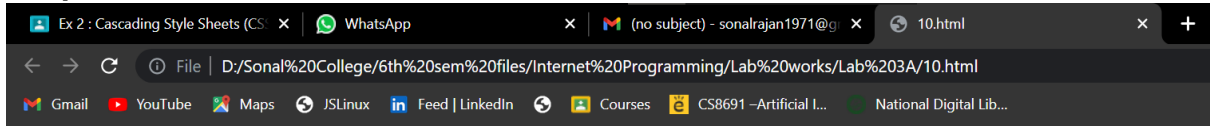
Algorithm:

1. Open netbeans and create a new file.
2. Divide the sentence into separate words.
3. If the words contain 3 letters, then push them into the array.
4. Print all those elements.
5. Display the output.

Program:

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <p id="demo"></p>
    <script>
      function combinator () {
        let s1="How are you and what are you doing";
        arr=s1.split(' ');
        new_arr=[];
        for(i=0;i<arr.length;i++) {
          if(arr[i].length==3) {
            new_arr.push(arr[i]);
          }
        }
        document.getElementById("demo").innerHTML =new_arr;
      }
      combinator();
    </script>
  </body>
</html>
```

Output:



How,are,you,and,are,you

3. Write a java script program to find out duplicate elements in an array.

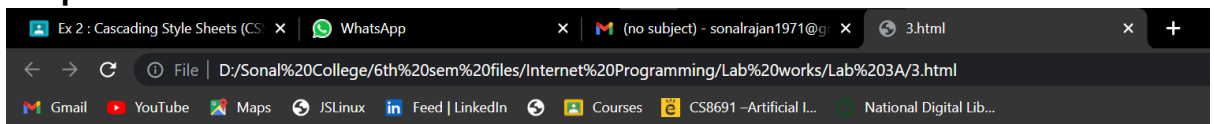
Algorithm:

1. Open netbeans and create a new file.
2. Give some random numbers in the array.
3. Check for every element in the array.
4. Compare the elements with other elements.
5. Display the output if a number is repeated.

Program:

```
<!DOCTYPE html>
<html>
  <head>
    <script>
      const numbers = [1, 2, 3, 2, 4, 5, 5, 6, 19, 21, 19, 22];
      const set = new Set(numbers);
      const duplicates = numbers.filter(item => {
        if (set.has(item)) {
          set.delete(item);
        }
        else {
          return item;
        }
      });
      document.write("The numbers are: " + numbers + "\n");
      document.write("And the duplicate elements are: " + duplicates);
    </script>
  </head>
</html>
```

Output:



The numbers are: 1,2,3,2,4,5,5,6,19,21,19,22 And the duplicate elements are: 2,5,19

4. Write a java script function that prints second largest and smallest element in an array.

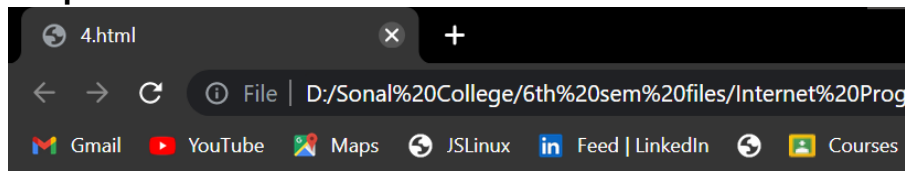
Algorithm:

1. Open netbeans and create a new file.
2. Declare an array and initialize the array.
3. Sort the elements in the array.
4. Print the second element in the array for second smallest element.
5. Print the last before element in the array for second largest element.
6. Display the output.

Program:

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <p id ="demo"></p>
    <p id ="demo1"></p>
    <script>
      const a = [4,3,6,1,5,2,7]
      function second(a){
        s1=[];
        s1=a.sort();
        len=a.length;
        n=len-1;
        document.getElementById("demo").innerHTML ="Second smallest
number is: "+s1[1];
        document.getElementById("demo1").innerHTML ="Second largest
number is: "+s1[n];
      }
      second(a);
    </script>
  </body>
</html>
```

Output:



Second smallest number is: 2

Second largest number is: 7

5. Write a java script function to count the number of occurrences of a particular word in a sentence

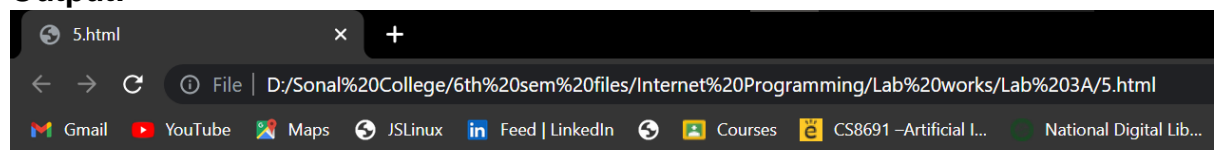
Algorithm:

1. Open netbeans and create a new file.
2. Split the string by spaces.
3. Take a variable count = 0 and in every true condition we increment the count by 1.
4. Now run a loop at 0 to length of string and check if our string is equal to the word
5. If the condition is true then increment the value of count by 1 and in the end, print the value of count.
6. Display the output.

Program:

```
<!DOCTYPE html>
<html>
  <head>
    <script>
      function countOccurrences(str,word) {
        let a = str.split(" ");
        let count = 0;
        for (let i = 0; i < a.length; i++) {
          if (word==(a[i]))
            count++;
        }
        return count;
      }
      let str = "The world is so big and the world in filled with land and
water.";
      let word = "world";
      document.write("The given string is: "+ str);
      document.write(" The number of '" + word + "' in the string is: "+
countOccurrences(str, word));
    </script>
  </head>
  <body></body>
</html>
```

Output:



6. Write a java script function to add two complex numbers [create complex number as objects]

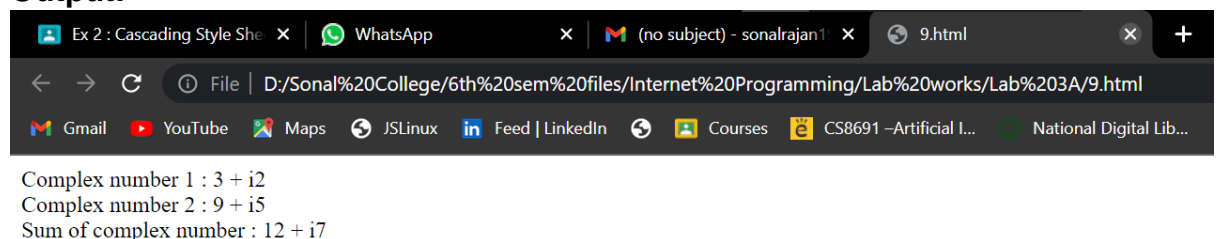
Algorithm:

1. Open netbeans and create a new file.
2. Declare a complex number by its complex and number part.
3. First add the original vale and then add the complex value.
4. Now after adding the original and the imaginary value store it in a variable.
5. Display the output.

Program:

```
<!DOCTYPE html>
<html>
  <script>
    class Complex {
      constructor(tempReal, tempImaginary) {
        this.real = tempReal;
        this.imaginary = tempImaginary;
      }
    }
    function addComp(C1,C2) {
      let temp = new Complex();
      temp.real = C1.real + C2.real;
      temp.imaginary = C1.imaginary + C2.imaginary;
      return temp;
    }
    let C1 = new Complex(3, 2);
    document.write("Complex number 1 : "+ C1.real + " + i "+
C1.imaginary+"<br>");
    let C2 = new Complex(9, 5);
    document.write("Complex number 2 : "+ C2.real + " + i "+
C2.imaginary+"<br>");
    let C3 = new Complex();
    C3 = addComp(C1, C2);
    document.write("Sum of complex number : "+ C3.real + " + I "+
C3.imaginary+"<br>");
  </script>
</html>
```

Output:



7. Write a java script function to calculate the experience of an employee in a company when joining date is given as input.

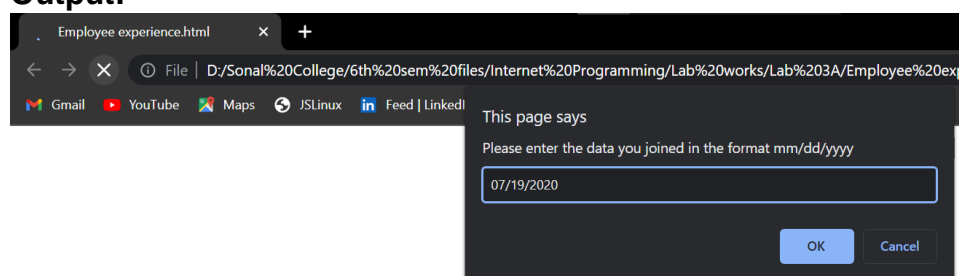
Algorithm:

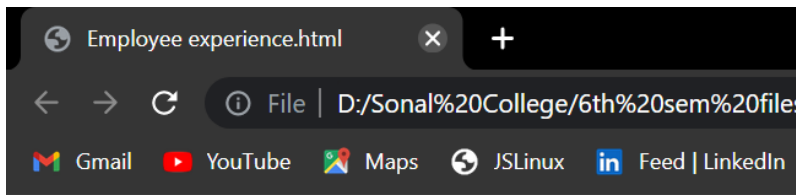
1. Open netbeans and create a new file.
2. Get the input of the joining date of the employee in the format of MM/DD/YY.
3. Calculate the time difference in seconds between the input date and today's date using the inbuilt function.
4. Using the difference found in the seconds, calculate the days and years difference.
5. Display the output.

Program:

```
<!DOCTYPE html>
<html>
  </head>
  <script>
    var customerName = prompt("Please enter the date you joined in the
format mm/dd/yyyy", "<date goes here>");
    function myFunction() {
      document.getElementById("frm1").submit();
    }
  </script>
  <script type = "text/javascript" >
    var date1 = new Date(customerName);
    var date2 = new Date("04/26/2022");
    var Difference_In_Time = date2.getTime() - date1.getTime();
    var Difference_In_Days = Difference_In_Time / (1000 * 3600 * 24);
    var daydiff = (Difference_In_Time/ 31536000000).toFixed(0);
    document.write(" The experience of the employee is: " +
Difference_In_Days + " days");
    document.write(" \n that is " + daydiff + " years");
  </script>
</head>
<body><body>
</html>
```

Output:





The experience of the employee is: 646 days that is 2 years

8. Write a java script to find whether given email address is valid or not [Use Regular expression]

Algorithm:

1. Open netbeans and create a new file.
2. Receive the input form the user of the mail id.
3. Define a regular expression that depicts the normal format of the mail id.
4. It must contain a ".", "@", and "com".
5. If the input satisfies the regular expression.
6. Display the output.

Program:

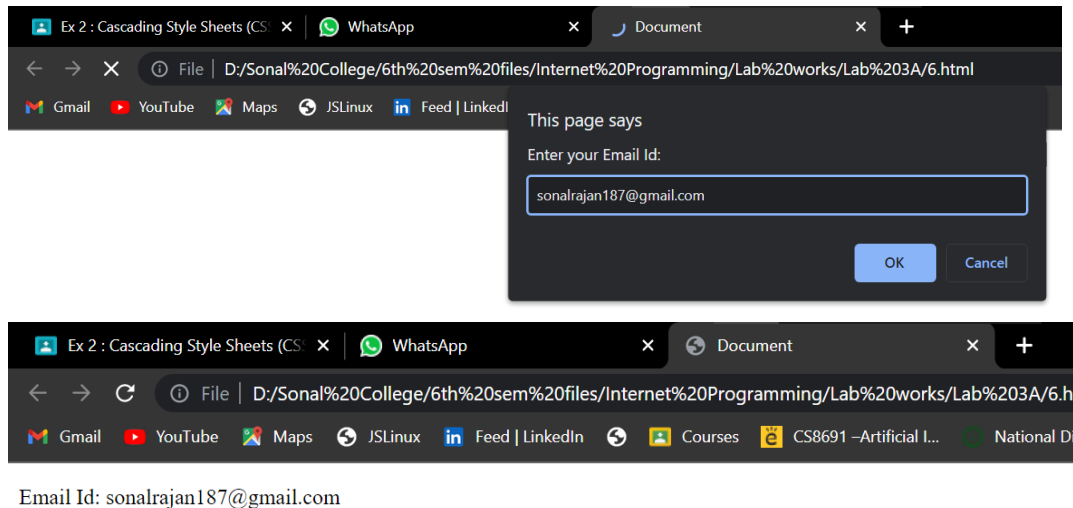
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
  </head>
  <body>
    <p id="demo"></p>
    <p id="demo1"></p>
    <script>
      function myfunc() {
        let get = window.prompt("Enter your Email Id:");
        let regex = /^[a-z A-Z 0-9]@[a-z].[a-z]/;
        let test = regex.test(get);
        if(test == false) {
          window.alert("Invalid Mail Id. Enter the correct Mail Id");
        }
        else {
          window.alert("Valid Mail Id.");
        }
        let get1 = window.prompt("Enter your Email Id:");
        document.getElementById("demo").innerHTML="Email Id: "+ get1;
      }
      myfunc();
    </script>
```

```

    </body>
</html>

```

Output:



9. Write a java script function to find whether given IP address is valid or not [Use Regular expression].

Algorithm:

1. Open netbeans and create a new file.
2. Receive the input form the user of the IP address.
3. Define a regular expression that depicts the normal format of the IP address.
4. It must contain three dots, numbers in range 001 to 255.
5. If the input satisfies the regular expression.
6. Display the output.

Program:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>IP address validation</title>
    <link rel='stylesheet' href='form-style.css' type='text/css' />
  </head>
  <body onload='document.form1.text1.focus()>
    <div class="mail">
      <h2>Input ip address and submit</h2>
      <form name="form1" action="#">
        <ul>
          <li><input type='text' name='text1' /></li>
          <li>&nbsp;</li>

```

```

        <li class="submit"><input type="submit" name="submit"
value="Submit" onclick="ValidateIp(document.form1.text1)"/></li>
        <li>&nbsp;</li>
    </ul>
</form>
</div>
<script>
    function ValidateEmail(inputText){
        var mailformat = /^(25[0-5]|2[0-4][0-9]||[01]?[0-9][0-9]?)\.(25[0-5]|2[0-
4][0-9]||[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]||[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-
9]||[01]?[0-9][0-9]?)$/;
        if(inputText.value.match(mailformat)){
            alert("Valid ip address!");
            document.form1.text1.focus();
            return true;
        }
        else{
            alert("You have entered an invalid ip address!");
            document.form1.text1.focus();
            return false;
        }
    }
</script>
<style>
    li {
        list-style-type: none;
        font-size: 16pt;
    }
    .mail {
        margin: auto;
        padding-top: 10px;
        padding-bottom: 10px;
        width: 400px;
        background : #D8F1F8;
        border: 1px solid silver;
    }
    .mail h2 {
        margin-left: 38px;
    }
    input {
        font-size: 20pt;
    }
    input:focus, textarea:focus{
        background-color: lightyellow;
    }
    input submit {

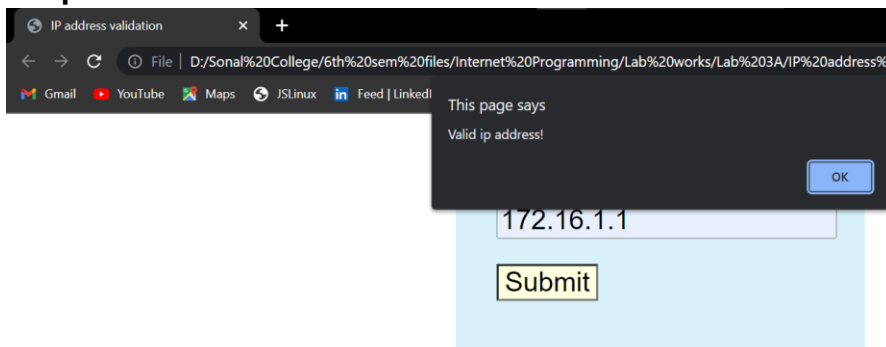
```

```

        font-size: 12pt;
    }
    .rq {
        color: #FF0000;
        font-size: 10pt;
    }
</style>
</body>
</html>

```

Output:



10. Write a java script function that prints the string that begins and end with a vowel [Use Regular expression].

Algorithm:

1. Open netbeans and create a new file.
2. Receive the input from the user of the word.
3. Define a regular expression that depicts the word.
4. It must contain a word that starts and ends with a vowel.
5. If the input satisfies the regular expression.
6. Display the output.

Program:

```

<!DOCTYPE html>
<html>
<script>
    function checkIfStartsWithVowels(str) {
        if (!(str[0] == 'A' || str[0] == 'a' || str[0] == 'E' || str[0] == 'e' || str[0] == 'I' ||
str[0] == 'i' || str[0] == 'O' || str[0] == 'o' || str[0] == 'U' || str[0] == 'u'))
            return 1;
        else
            return 0;
    }
    function check(str) {
        if (checkIfStartsWithVowels(str))
            document.write( str+" : Not Accepted<br>");
    }

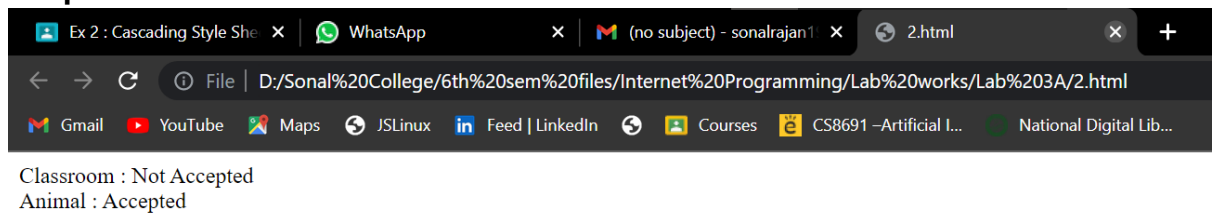
```

```

        else
            document.write(str+" : Accepted<br>");
    }
    var str = "Classroom";
    check(str);
    str = "Animal";
    check(str);
</script>
</html>

```

Output:



Observation	
Record	
Total	

RESULT:

Thus, all the java scrip function programs were executed successfully.

QUESTION:

Test JavaScript Form Validataion

Name*	<input type="text"/>	Please enter your name!
Address	<input type="text"/>	
Zip Code*	<input type="text"/>	
Country*	<input type="text" value="Please select..."/>	
Gender*	<input type="radio"/> Male <input type="radio"/> Female	
Preferences*	<input type="checkbox"/> Red <input type="checkbox"/> Green <input type="checkbox"/> Blue	
Phone*	<input type="text"/>	
Email*	<input type="text"/>	
password (6-8 characters)*	<input type="text"/>	
Verify password*	<input type="text"/>	
	<input type="button" value="SEND"/> <input type="button" value="CLEAR"/>	

1. Ensure that all the fields have some entry.
2. Name should be of minimum 6 characters and maximum 12 characters, should have only alphabets
3. Address field should have some entry
4. Zip code must be a 6-digit number
5. Country field should have some selection
6. Gender and Preferences field should have some selection
7. Phone number must be a 10-digit number
8. Email field should follow email format, should have @ and . symbols, e.g., aaa@gmail.com
9. Password should be 6-8 characters, should have both numbers and alphabets
10. Password and verify password field should have the same content

When you click the send button if all the fields have entries and if it satisfies all the constraints mentioned above, it should display successfully registered message...

If any one of the constraints is not satisfied, it should display the message as not a valid entry nearby that field. Example is also mentioned in the image. (Note the name field)

AIM:

To create a form validation using Java Script.

ALGORITHM:

1. Open Netbeans and create a new project.
2. Use form tag to display the details that are needed to be filled.
3. Give separate div tags for easy modifications at a particular place.
4. Use script tag for validation.
5. Some basics constraints are email id must definitely contain '@' symbol.
6. If all the constraints are satisfied, then display registration successful.
7. Else display the alert message for which the content must be changed.
8. Display the output.

PROGRAM:**index.html:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Form Validation</title>
    <style>
      h1{
        text-align: center;
      }
      div{
        text-align: center;
        border: 2px solid black;
        margin-top: 2%;
      }
      p{
        color: red;
      }
    </style>
  </head>
  <body>
    <h1>Java Script Form Validation</h1>
    <div>
      <form name="form" onsubmit="return validate()" method="post"><br>
        <label for="fname">Name*</label>
        <input type="text" name="fname" id="fname" required><br>
        <p>Please enter your name!</p>
        <label for="address">Address</label>
        <input type="text" name="address" id="address" required><br><br>
        <label for="code">Zip code*</label>
        <input type="number" name="code" id="code" required><br><br>
        <label for="cselect">Country*</label>
        <select name="country" id="country" required><br><br>
```

```

        <option value="select">Please Select..</option>
        <option value="india">India</option>
        <option value="australia">Australia</option>
        <option value="canada">Canada</option>
        <option value="vietnam">Vietnam</option>
        <option value="malaysia">Malaysia</option>
        <option value="singapore">Singapore</option>
        <option value="sri lanka">Sri Lanka</option>
        <option value="usa">USA</option>
    </select><br><br>
    <label for="gender">Gender*</label>
    <input type="radio" name="gender" id="male">
    <label for="male">Male</label>
    <input type="radio" name="gender" id="female">
    <label for="female">Female</label><br><br>
    <label for="prefer">Preferences*</label>
    <input type="checkbox" name="prefer" id="red">
    <label for="red">Red</label>
    <input type="checkbox" name="prefer" id="green">
    <label for="green">Green</label>
    <input type="checkbox" name="prefer" id="blue">
    <label for="blue">Blue</label><br><br>
    <label for="phone">Phone*</label>
    <input type="number" name="number" id="number" required><br><br>
    <label for="mail">Email*</label>
    <input type="text" name="email" id="email" required><br><br>
    <label for="password">Password(6-8 characters)*</label>
    <input type="text" name="password" id="password" required><br><br>
    <label for="verify">Verify Password*</label>
    <input type="text" name="verify" id="verify" required><br><br>
    <input type="submit" value="SEND">
    <input type="submit" value="CLEAR">
</form><br>
</div>

<script>
function validate(){
    var name=document.getElementById("fname").value;
    var code=document.getElementById("code").value;
    var n=document.getElementById("number").value;
    var pw=document.getElementById("password").value;
    var vp=document.getElementById("verify").value;
    var e=document.getElementById("email").value;
    if(name.match(/^[A-Za-z]+$/)){
        if(code.length==6){
            if(n.length==10){
                if(e.match(/^\w+@[a-zA-Z_]+?\.[a-zA-Z]{2,3}$/)){
                    if(pw==vp){
                        if(pw.length>=6||pw.length<=8){
                            alert("Registration Successful");
                        }
                    }
                }
            }
        }
    }
}

```

```
        return true;
    }
    else{
        alert("Use 6-8 characters in passwords");
        return false;
    }
}
else{
    alert("Paswords does not match");
    return false;
}
}
else{
    alert("Enter valid email address");
    return false;
}
}
else{
    alert("Please enter 10 digit phone number");
    return false;
}
}
else{
    alert("Please enter 6 digit zip code");
    return false;
}
}
else{
    alert("Enter valid name");
    return false;
}
}
</script>
</body>
</html>
```

OUTPUT:

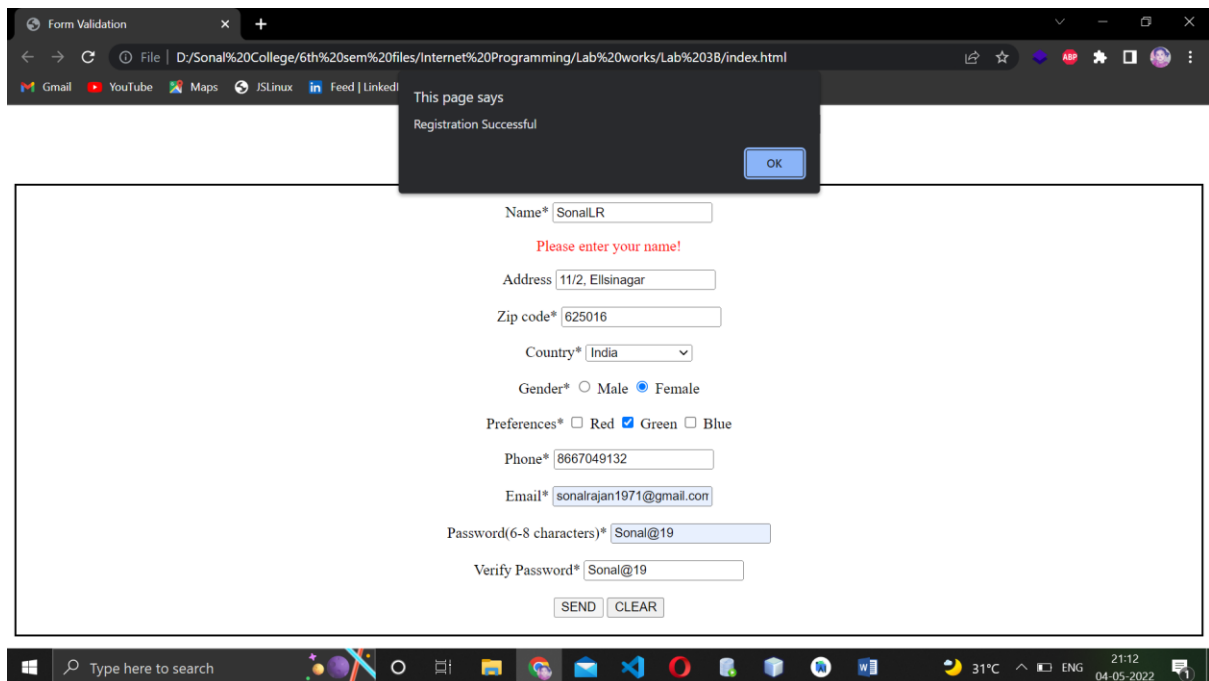
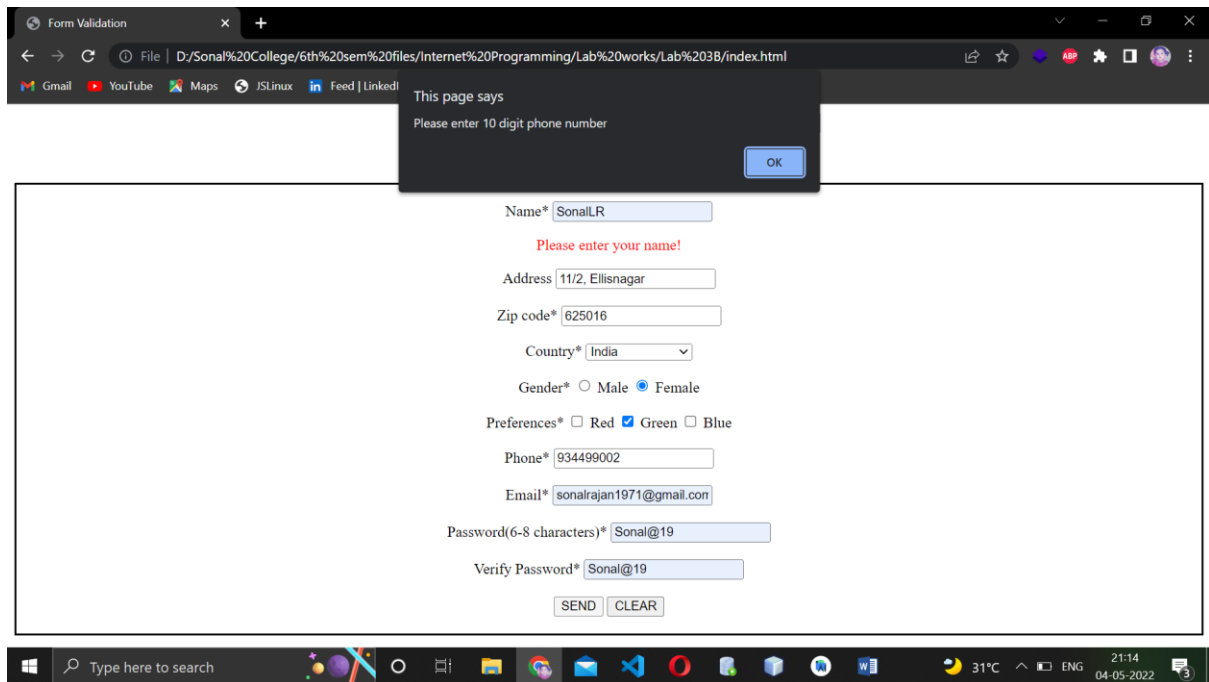
The first screenshot shows a web browser window titled "Form Validation" with the URL "D:/Sonal%20College/6th%20sem%20files/Internet%20Programming/Lab%20works/Lab%203B/index.html". The page displays a form titled "Java Script Form Validation" with the following fields and labels:

- Name*
- Please enter your name!
- Address
- Zip code*
- Country*
- Gender* ☐ Male ☐ Female
- Preferences* ☐ Red ☐ Green ☐ Blue
- Phone*
- Email*
- Password(6-8 characters)*
- Verify Password*
-

The second screenshot shows the same form with the following data entered:

- Name*
- Please enter your name!
- Address
- Zip code*
- Country*
- Gender* ☐ Male ☒ Female
- Preferences* ☐ Red ☒ Green ☐ Blue
- Phone*
- Email*
- Password(6-8 characters)*
- Verify Password*
-

A validation error message is displayed: "This page says Please enter 6 digit zip code". The message box has an "OK" button.



Observation	
Record	
Total	

RESULT:

Thus, form validation using Java Script were created and validated successfully.

AIM:

To write a servlet program to find the gross salary for an employee .Test this by a client program to get the form data inputs (name, empid, basic pay, HRA) and display the result (name, empid, basic pay, HRA along with calculated DA(50% of Basic pay) and gross pay(basic pay+hra+DA)).

ALGORITHM:

1. Create an html page displaying a form to collect the employee details like name, id, basic pay and HRA.
2. Mention the form method as "get" and include a "submit" input type for the form action to perform.
3. Connect the form with a JAVA servlet by mentioning the servlet name in the action attribute of the HTML form.
4. Get the input values by using `getParameter()` method in the request `HttpServletRequest`.
5. Calculate the Gross Pay and DA using the input values in appropriate data types.
6. Display the Gross Pay and DA using `PrintWriter` function.

PROGRAM:**index.html:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Java servlet</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <form action="firstserv" method="get">
      <label>Enter Name:</label>
      <input type="text" name="ename"><br><br>
      <label>Enter Employee id:</label>
      <input type="text" name="empid"><br><br>
      <label>Enter basic pay</label>
      <input type="number" name="bp"><br><br>
      <label>Enter HRA:</label>
      <input type="number" name="hrapay"><br><br>
      <input type="submit" value="GETINFO">
    </form>
  </body>
</html>
```

firstserv.java:

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class firstserv extends HttpServlet{
    protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            String ename=request.getParameter("ename");
            String empid=request.getParameter("empid");
            int bpay=Integer.parseInt(request.getParameter("bp"));
            int hra=Integer.parseInt(request.getParameter("hrapay"));
            float da=0.50f*bpay;
            double grosspay=bpay+hra+da;
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet firstserv</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Employee Information:</h1>");
            out.println("<div><ul><li>Employee Name: "+ename+"</li>"
                + "<li>Employee ID: "+empid+"</li>"
                + "<li>Employee Basic pay: "+bpay+"</li>"
                + "<li>Employee hra: "+hra+"</li>"
                + "<li>Employee da: "+da+"</li>"
                + "<li>Employee gross pay: "+grosspay+"</li></ul></div>");
            out.println("</body>");
            out.println("</html>");
        } finally {
            out.close();
        }
    }
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        processRequest(request, response);
    }
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
```



```

        processRequest(request, response);
    }
    @Override
    public String getServletInfo() {
        return "Short description";
    }
}

```

OUTPUT:

Enter Name:

Enter Employee id:

Enter basic pay:

Enter HRA:

localhost:8080/Lab_Ex_4/firstserv?ename=sonal&empid=emp87&...

Employee Information:

- Employee Name: sonal
- Employee ID: emp87
- Employee Basic pay: 3000
- Employee hra: 500
- Employee da: 1500.0
- Employee gross pay: 5000.0

Observation	
Record	
Total	
Sign	

RESULT:

Thus, the given program to calculate Gross Salary of an employee is written and executed using JAVA servlets successfully.

AIM:

To develop a java application for the following scenario using servlets:
Hamen's Book shop maintains the inventory of books that are being sold at the shop. The list includes details such as author, title, price, publisher and stock position. Whenever a customer wants a book, the sales person inputs the title and author and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed. If it is, then the system displays the book details and requests for the number of copies required. If the requested copies are available, the total cost of the requested copies is displayed; otherwise the message "Required copies not in stock" is displayed.

ALGORITHM:

1. Create an html page displaying a form to collect the details like title and author of the book.
2. Mention the form method as "get" and include a "submit" input type for the form action to perform.
3. Connect the form with a JAVA servlet by mentioning the servlet name in the action attribute of the HTML form.
4. Get the input values by using `getParameter()` method in the request `HttpServletRequest`.
5. Check whether the given book is available in the database where the database is maintained as an array of class objects.
6. If yes, input the number of copies required.
7. If the stock is available, calculate the total amount of the book(s) and display the total bill amount.

PROGRAM:**index.html:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Servlet</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <form action="serv" method="get">
      Enter book name:<input type="text" name="title" id="title"><br><br>
      Enter book author:<input type="text" name="author" id="author"><br><br>
      <input type="submit" value="GET INFO">
    </form>
  </body>
</html>
```

serv.java:

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet(name = "BookServlet", urlPatterns = {"/BookServlet"})

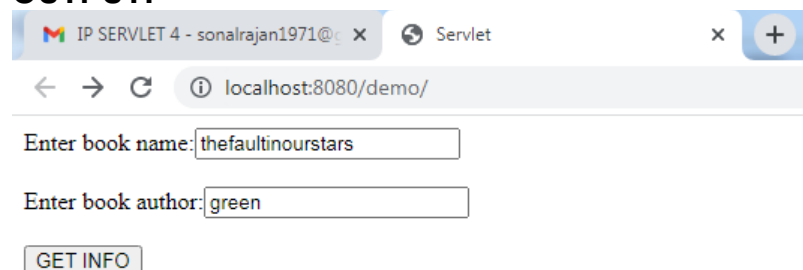
public class serv extends HttpServlet {
    bookdetailsinventory obj[]=new bookdetailsinventory[5];
    protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        int flag=0;
        obj[0]=new bookdetailsinventory("thefaultinourstars","green",500,"abc publications",8);
        obj[1]=new bookdetailsinventory("fivefeetapart","rowling",1500,"xyz publications",12);
        obj[2]=new bookdetailsinventory("2states","chetan bhagat",800,"ko publications",16);
        obj[3]=new bookdetailsinventory("3mistakesofmylife","athvik",200,"keert
publications",10);
        obj[4]=new bookdetailsinventory("fivepointsomeone","chetan",900,"okk
publications",8);
        try {
            String title=request.getParameter("title");
            String author=request.getParameter("author");
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Details of Book</title>");
            out.println("</head>");
            out.println("<body>");
            for(int i=0;i<5;i++){
                if(title.equalsIgnoreCase(obj[i].title) && author.equalsIgnoreCase(obj[i].author)){
                    out.println("<h1>Book "+title+ " found </h1>");
                    out.println("<h3> Title : "+ obj[i].title + "</h3>");
                    out.println("<h3> Author : "+ obj[i].author + "</h3>");
                    out.println("<h3> Price : "+ obj[i].price + "</h3>");
                    out.println("<h3> Publisher : "+ obj[i].publisher + "</h3>");
                    out.println("<br><br><form action='serv' method='post' ><label >Number of
Copies required </label>"
                        + "<span><input type='text' name='count'></span><br><input type='hidden'
name='posn' value='"+i+"'>"
                        + "<br><input type='submit' value='SEARCH'></form>");
                    out.println("</body>");
                    out.println("</html>");
                }} finally {
                    out.close();
                }
            }
        }
        @Override
        protected void doGet(HttpServletRequest request, HttpServletResponse response)
```

```

        throws ServletException, IOException {
    processRequest(request, response);
}
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    //processRequest(request, response);
    int count=Integer.parseInt(request.getParameter("count"));
    int pos=Integer.parseInt(request.getParameter("posn"));
    if(count<obj[pos].stock){
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        out.println("<h1>Total Bill Amount : "+((obj[pos].price)*count)+"</h1>");
        out.println("</html></body>");
    }
}
public String getServletInfo() {
    return "Short description";
}
}
class bookdetailsinventory{
    String title;
    String author;
    int price;
    String publisher;
    int stock;
    bookdetailsinventory(String t,String a, int pr,String pub,int st){
        title=t;
        author=a;
        price=pr;
        publisher=pub;
        stock=st;
    }
}

```

OUTPUT:

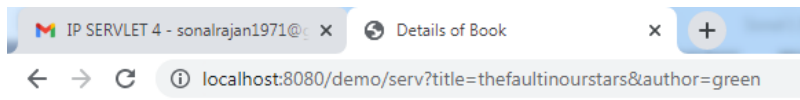


IP SERVLET 4 - sonalrajan1971@ x Servlet x +

localhost:8080/demo/

Enter book name:

Enter book author:



Book thefaultinourstars found

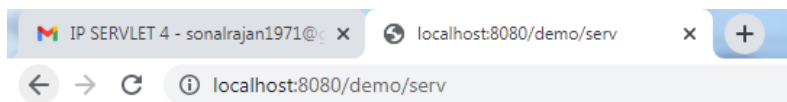
Title : thefaultinourstars

Author : green

Price : 500

Publisher : abc publications

Number of Copies required



Total Bill Amount : 2000

Observation	
Record	
Total	
Sign	

RESULT:

Thus, the given program to calculate Gross Salary of an employee is written and executed using JAVA servlets successfully.

AIM:

To create a JAVA Application using Java Servlets with Database Connectivity for the given scenario: Create a student database which has a table to with student name, rollno, department, CGPA and average.

1. Create a HTML form which gets student roll no and displays the other details of the student using servlet
2. Create another HTML form to get the values about the student and store it in the student database.

ALGORITHM:

1. Create an html page displaying a form to collect the student roll no to display the student details.
2. Mention the form method as "get" and include a "submit" input type for the form action to perform.
3. Connect the form with a JAVA servlet by mentioning the servlet name in the action attribute of the HTML form.
4. Get the input value by using `getParameter()` method in the request `HttpServletRequest`.
5. Fetch the student details using roll no by executing a SQL query after a Java DB (embedded) is created and table is filled.
6. Display the student details fetched from the database table and get appropriate details to add a student details in the table using form method as "post".
7. Insert the details in the table by executing `INSERT INTO` SQL statement.

PROGRAM:**index.html:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Student Database</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <h1>GET STUDENT DETAILS</h1>
    <form method='get' action='NewServlet'>
      <label>Enter student roll number</label><br><br>
      <input type='number' name='rollno' />
      <br><br>
      <input type="submit" value="GET DETAILS" />
    </form>
  </body>
</html>
```

serv.java:

```
package FirstServlet;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

    Connection conn = null;
    Statement st = null;
    ResultSet rs = null;

    protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        int rollno = Integer.parseInt(request.getParameter("rollno"));

        try {

            Class.forName("org.apache.derby.jdbc.ClientDriver");
            String dburl = "jdbc:derby://localhost:1527/StudentDB";
            conn = DriverManager.getConnection(dburl, "iplab", "iplab");
            st = conn.createStatement();
            rs = st.executeQuery("SELECT * FROM STUDENTDETAILS WHERE
ROLLNO=" + rollno);

            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Student Database</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>GET STUDENT DETAILS</h1>");

            while(rs.next()) {
                out.println("<h3>Roll number: " + rs.getString(1) + "</h3>");
                out.println("<h3>Name: " + rs.getString(2) + "</h3>");
                out.println("<h3>Department: " + rs.getString(3) + "</h3>");
                out.println("<h3>CGPA: " + rs.getString(4) + "</h3>");
                out.println("<h3>Average: " + rs.getString(5) + "</h3>");
                out.println("<h1>ADD STUDENT DETAILS</h1>");
            }
        }
    }
}
```

```

        out.println("<form method='post' action='Serv'>"
            + "<label>Enter student roll number</label><br><br>"
            + "<input type='number' name='srollno'><br><br>"
            + "<label>Enter student name</label><br><br>"
            + "<input type='text' name='sname'><br><br>"
            + "<label>Enter student department</label><br><br>"
            + "<input type='text' name='sdept'><br><br>"
            + "<label>Enter student CGPA</label><br><br>"
            + "<input type='number' name='scgpa'><br><br>"
            + "<label>Enter student Average</label><br><br>"
            + "<input type='number' name='savg'><br><br>"
            + "<input type='submit' value='ADD DETAILS'>"
            + "</form>");
    }

    out.println("</body>");
    out.println("</html>");

} catch(Exception e) {
    System.out.println(e.getMessage());
} finally {
    out.close();
}
}

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
    processRequest(request, response);
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {

    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();

    Integer rollno = Integer.parseInt(request.getParameter("srollno"));
    String name = request.getParameter("sname");
    String department = request.getParameter("sdept");
    Integer cgpa = Integer.parseInt(request.getParameter("scgpa"));
    Integer average = Integer.parseInt(request.getParameter("savg"));

    try {

        Class.forName("org.apache.derby.jdbc.ClientDriver");
        String dburl = "jdbc:derby://localhost:1527/StudentDB";

```



```

        conn = DriverManager.getConnection(dburl, "iplab", "iplab");
        st = conn.createStatement();

        String query="INSERT INTO STUDENTDETAILS
(ROLLNO,STUDENTNAME,DEPARTMENT,CGPA,AVERAGE) VALUES
("+rollno+", "+"+"+name+", "+"+department+", "+"+cgpa+", "+"+average+)";
        st.executeUpdate(query);

        rs = st.executeQuery("SELECT * FROM STUDENTDETAILS WHERE
ROLLNO=" + rollno);

        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Student Database</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>ADD STUDENT DETAILS</h1>");
        out.println("<h2>Student details added successfully ! </h2>");

        while(rs.next()) {
            out.println("<h3>Roll number: " + rs.getString(1) + "</h3>");
            out.println("<h3>Name: " + rs.getString(2) + "</h3>");
            out.println("<h3>Department: " + rs.getString(3) + "</h3>");
            out.println("<h3>CGPA: " + rs.getString(4) + "</h3>");
            out.println("<h3>Average: " + rs.getString(5) + "</h3>");
        }

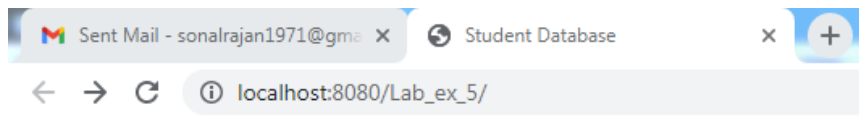
        out.println("</body>");
        out.println("</html>");

    } catch(Exception e) {
        System.out.println(e.getMessage());
    } finally {
        out.close();
    }
}

@Override
public String getServletInfo() {
    return "Short description";
}
}

```

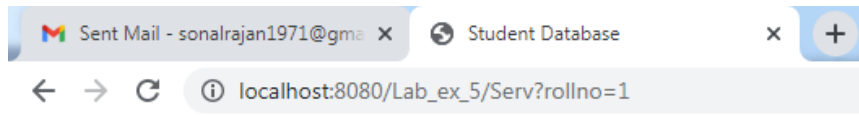
OUTPUT:



GET STUDENT DETAILS

Enter student roll number

GET DETAILS



GET STUDENT DETAILS

Roll number: 1

Name: Sonal

Department: CSE

CGPA: 9

Average: 90

ADD STUDENT DETAILS

Enter student roll number

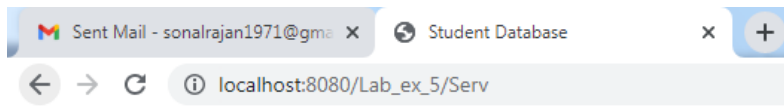
Enter student name

Enter student department

Enter student CGPA

Enter student Average

ADD DETAILS



ADD STUDENT DETAILS

Student details added successfully !

Roll number: 8

Name: Shahasni

Department: ECE

CGPA: 9

Average: 90

Observation	
Record	
Total	
Sign	

RESULT:

Thus, a JAVA Application using Java Servlets with Database Connectivity for the given scenario is written and executed successfully.

AIM:

To create an online examination portal using JSP, database and cookies

1. Create a login form which asks for username and password, when the user enters correct username and password, it should be redirected to a jsp script which checks whether entered credentials are correct using database
2. If the credentials are wrong, display a message to enter correct user name and password
3. If the credentials are correct, display the questions to the user. Store the credentials in a cookie
4. User have to select the answers and press the submit button
5. When the submit button is clicked, a jsp script,
 - a. Have to calculate the score
 - b. Store the score in the database
 - c. Address the user with his/her name using the cookie value as Hello "username".
6. Display all user's score at the end.

ALGORITHM:

1. Create an html page displaying a form to collect the student username and password to login.
2. The form is directed to a JSP page which checks if the given username and password match with the stored details in the database. And, cookies are used store the student's username.
3. If yes, the page will display a form to collect answers for the multiple choice questions displayed. Once the user presses a submit button, the form is directed to another JSP page.
4. In that JSP page, it checks whether the submitted answers are correct and increments the score. Also, the student's score is updated in the database.
5. The page will display the student's username retrieved from the cookie and the score of the student. It will also display the leader board from the database.
6. If the entered username and password doesn't match with the stored details in the database, then error is displayed as "Incorrect username or password".

PROGRAM:

index.html:

```
<!DOCTYPE html>
<html>
  <head>
    <title>ONLINE EXAMINATION </title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body style="text-align:center;margin:auto 50px;">
    <form method="get" action='newjsp.jsp'>
      <h2>Username</h2><br>
      <input type='text' name='username'><br><br>
      <h2>Password</h2><br>
      <input type='password' name='password'><br><br>
      <input type='submit' value="LOG IN" />
    </form>
  </body>
</html>
```

newjsp.jsp:

```
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"
lang='java' import='java.sql.*'>
    <title>ONLINE EXAMINATION</title>
  </head>
  <body>
    <%
      response.setContentType("text/html;charset=UTF-8");
      String username = request.getParameter("username");
      String password = request.getParameter("password");
      try {
        String dburl = "jdbc:derby://localhost:1527/ip-lab-6";
        Connection conn = DriverManager.getConnection(dburl, "sonal", "sonal");
        Statement st = conn.createStatement();
        ResultSet rs = st.executeQuery("SELECT * FROM USERSDETAILS
WHERE USERNAME='" + username + "'");
        while(rs.next())
        {
          String retrievedpassword = rs.getString(1);

          if(retrievedpassword.equals(password)) {
```

```

        Cookie ck1 = new Cookie("username", username);
        Cookie ck2 = new Cookie("password", password);
        response.addCookie(ck1);
        response.addCookie(ck2);

        out.println("<html><head><title>ONLINE
EXANIMATION</title></head>");
        out.println("<body style='text-align:center; margin: auto 50px;'>");
        out.println("<h2>Multiple Choice Questions</h2>");

        out.println("<form action='newjsp1.jsp'>");

        out.println("<h3>Who is the first Prime Minister of India?</h3><br>");
        out.println("<input type='radio' name='ans1'
value='jawaharlalnehru'><label>Jawaharlal Nehru</label><br>");
        out.println("<input type='radio' name='ans1'
value='mahatmagandhi'><label>Mahatma Gandhi</label><br>");
        out.println("<input type='radio' name='ans1'
value='subhashchandraboze'><label>Subhash Chandra Bose</label><br><br>");

        out.println("<h3>Which is the capital of India?</h3><br>");
        out.println("<input type='radio' name='ans2'
value='tamilnadu'><label>Tamil Nadu</label><br>");
        out.println("<input type='radio' name='ans2'
value='newdelhi'><label>New Delhi</label><br>");
        out.println("<input type='radio' name='ans2'
value='maharastra'><label>Maharastra</label><br><br>");

        out.println("<h3>Pick the odd one out</h3><br>");
        out.println("<input type='radio' name='ans3'
value='lovestory'><label>Love story</label><br>");
        out.println("<input type='radio' name='ans3'
value='blankspace'><label>Blank space</label><br>");
        out.println("<input type='radio' name='ans3' value='takitaki'><label>Taki
Taki</label><br><br><br>");
        out.println("<input type='submit' value='SUBMIT'>");
        out.println("</form>");
    } else {
        out.println("<h3>Incorrect Username or Password ! </h3>");
    }
    out.println("</body></html>");
}
} catch(Exception e) {
    System.out.println(e);
} finally {
    out.close();
}
}

```

```

    %>
</body>
</html>

```

newjsp1.jsp:

```

<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8 lang='java'
import='java.sql.*'">
        <title>ONLINE EXAMINATION</title>
    </head>
    <body>
        <%
            Integer score = 0;
            response.setContentType("text/html;charset=UTF-8");
            String ans1 = request.getParameter("ans1");
            String ans2 = request.getParameter("ans2");
            String ans3 = request.getParameter("ans3");
            if (ans1.equals("jawaharlalnehru")) {
                score++;
            }
            if (ans2.equals("newdelhi")) {
                score++;
            }
            if (ans3.equals("takitaki")) {
                score++;
            }
            String dburl = "jdbc:derby://localhost:1527/ip-lab-6";
            Connection conn = DriverManager.getConnection(dburl, "baruni", "baruni");
            Statement st = conn.createStatement();
            Cookie[] cookie = request.getCookies();
            String username = cookie[1].getValue();
            String password = cookie[2].getValue();
            st.executeUpdate("UPDATE USERSDETAILS SET SCORE=" + score + "
WHERE USERNAME='" + username + "'");

            out.println("<html>><head><title>ONLINE EXANIMATION</title></head>");
            out.println("<body style='text-align:center; margin: auto 50px;'>");
            out.println("<h2>Hello " + username + "! </h2>");
            out.println("<h3>Your score is " + score + "</h3>");
            ResultSet rs = st.executeQuery("SELECT USERNAME,SCORE FROM
USERSDETAILS");
            %>
            <table border='1' style='margin:auto;'>

```

```

        <tr>
            <th>USERNAME</th>
            <th>SCORE</th>
        </tr>
        <% while (rs.next()) {%>
        <tr>
            <td> <%= rs.getString(1)%></td>
            <td> <%= rs.getString(2)%></td>
        </tr>
        <% } %>
    </table>
    <%
        out.println("</body>");
        out.println("</html>");
    %>
</body>
</html>

```

OUTPUT:

Username

Password

Multiple Choice Questions

Who is the first Prime Minister of India?

- ☒ Jawaharlal Nehru
- ☐ Mahatma Gandhi
- ☐ Subhash Chandra Bose

Which is the capital of India?

- ☒ Tamil Nadu
- ☐ New Delhi
- ☐ Maharashtra

Pick the odd one out

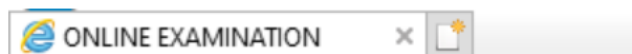
- ☐ Love story
- ☐ Blank space
- ☒ Taki Taki

SUBMIT

Hello sonal!

Your score is 2

USERNAME	SCORE
baruni	2
priya	2
poojha	3
sonal	2



Incorrect Username or Password !

Observation	
Record	
Total	
Sign	

RESULT:

Thus, a JSP with database connectivity and cookies for the given scenario is written and executed successfully.

1. Create a HTML form and validate it using PHP

Name
Name can not be left blank.

Email
Email can not be left blank.

Education
Tell us about your education.

Gender ☐ Male ☐ Female
Specify your gender.

Hobbies ☐ Drawing ☐ Singing ☐ Dancing
What are your hobbies.

Comment
This field is required.

AIM:

To create a HTML form and write a PHP program for form validation.

ALGORITHM:

1. Create an index.php file and fix the form in given format.
2. Use PHP functions to validate the inputs.
3. If every value is validated then submit the form.
4. Else display error message under every input field.
5. Finally, when all values are validated without any errors submit the form.

PROGRAM:

index.php:

```
<html>
  <head>
    <title>PHP Form Validation</title>
    <style>
      .des {
        display: block;
        background-color: lightblue;
        width: 40%;
      }
    </style>
  </head>
  <h1>
    <?php
      error_reporting(0);
      $error = array();
      function cleaninput($input)
      {
        foreach ($input as $key => $value)
        {
          $value = trim($value);
          $value = stripslashes($value);
          $value = htmlspecialchars($value);
        }
        return $input;
      }
      function validateinput($input)
      {
        if (!array_key_exists("gender", $input)) {
          $error["gender"] = "*Specify your gender.";
        }
        if (!array_key_exists("hobby", $input)) {
          $error["hobby"] = "*What are your hobbies?";
        }

        foreach ($input as $key => $value)
        {
          switch ($key)
          {
            case "usr":
              if (empty($value))
                $error["usr"] = "*Name cannot be left blank!";
              break;

            case "email":
              if (!filter_var($value, FILTER_VALIDATE_EMAIL))
                $error["email"] = "*Email cannot be left blank!";
              break;
```

```

        case "edu":
            if ($value == "--")
                $error["edu"] = "*Tell us about your education!";
            break;

        case "comment":
            if ($value=="")
                $error["comment"] = "*This field is required.";
            break;
    }
}
//var_dump($error);
return $error;
}
if (isset($_POST["submit-btn"])) {
    // var_dump($_POST);
    $cleandata = cleaninput($_POST);
    $error = validateinput($cleandata);
}
?>
</h1>
<div class = "des">
<body>
    <form action="index.php" method="post">
        <label>Name</label>
        <input type="text" name="usr"><br>
        <div style="color:red"><?php echo $error["usr"]; ?></div><br>

        <label>Email</label>
        <input type="text" name="email"><br>
        <div style="color:red"><?php echo $error["email"]; ?></div><br>

        <label for="edu">Education</label>
        <select id="edu" name="edu">
            <option value="--">--</option>
            <option value="hsc">HSC</option>
            <option value="sslc">SSLC</option>
            <option value="dip">Diplomo</option>
            <option value="be">B.E</option>
        </select>
        <div style="color:red"><?php echo $error["edu"]; ?></div><br>

        <label for="gender">Gender</label>
        <input type="radio" id="gender" name="gender" value="male">
        <label for="male">Male</label>
        <input type="radio" id="gender" name="gender" value="female">
        <label for="female">Female</label>
        <input type="radio" id="gender" name="gender" value="none">
        <label for="none">Other</label>
        <div style="color:red"><?php echo $error["gender"]; ?></div><br>
    </form>
</body>
</div>

```

```

<label for="hobbies">Hobbies</label>
<input type="checkbox" id="hobby" name="hobby" value="drawing">
<label for="vehicle1">Drawing</label>
<input type="checkbox" id="hobby" name="hobby" value="singing">
<label for="vehicle2">Singing</label>
<input type="checkbox" id="hobby" name="hobby" value="dancing">
<label for="vehicle3">Dancing</label>
<div style="color:red"><?php echo $error["hobby"]; ?></div> <br>

<label for="comment">Comment</label>
<br>
<textarea rows="4" cols="30" name="comment"></textarea>
<div style="color:red"><?php echo $error["comment"];
?>
</div>
<br><br>
<input type="submit" value="Submit" name="submit-btn">
</form>
</body>
</html>

```

OUTPUT:

PHP Form Validation

localhost/ip-lab-7/index.php

Name

Email

Education -- ▾

Gender ☐ Male ☐ Female ☐ Other

Hobbies ☐ Drawing ☐ Singing ☐ Dancing

Comment

Submit

PHP Form Validation

localhost/ip-lab-7/index.php

Name
*Name cannot be left blank!

Email
*Email cannot be left blank!

Education -- ▾
*Tell us about your education!

Gender ☐ Male ☐ Female ☐ Other
*Specify your gender.

Hobbies ☐ Drawing ☐ Singing ☐ Dancing
*What are your hobbies?

Comment
*This field is required.

Submit

PHP Form Validation

localhost/ip-lab-7/index.php

Name

Email

Education

Gender ☐ Male ☐ Female ☐ Other

Hobbies ☐ Drawing ☒ Singing ☐ Dancing

Comment

Submit

PHP Form Validation

localhost/ip-lab-7/index.php

Name

Email

Education

*Tell us about your education!

Gender ☐ Male ☐ Female ☐ Other

*Specify your gender.

Hobbies ☐ Drawing ☐ Singing ☐ Dancing

Comment

*This field is required.

Submit

PHP Form Validation

localhost/ip-lab-7/index.php

Name

Email

Education

*Tell us about your education!

Gender ☐ Male ☒ Female ☐ Other

*Specify your gender.

Hobbies ☐ Drawing ☐ Singing ☒ Dancing

Comment

*This field is required.

Submit

RESULT:

Thus, the HTML form was successfully created and validated using php.

1. Create a login form using PHP. When a user enters correct user name and password, display a message Logged in successfully. Else display an error message "enter correct user name and password". Use MYSQL database to store user name and passwords.

AIM:

To create a login form using PHP.

ALGORITHM:

1. Create an index1.php file and fix the form in given format.
2. Form should contain username and password.
3. Check student database and verify the password.
4. If password is correct, then display login 'successful'.
5. Else print login 'not successful'.

PROGRAM:

Index1.php:

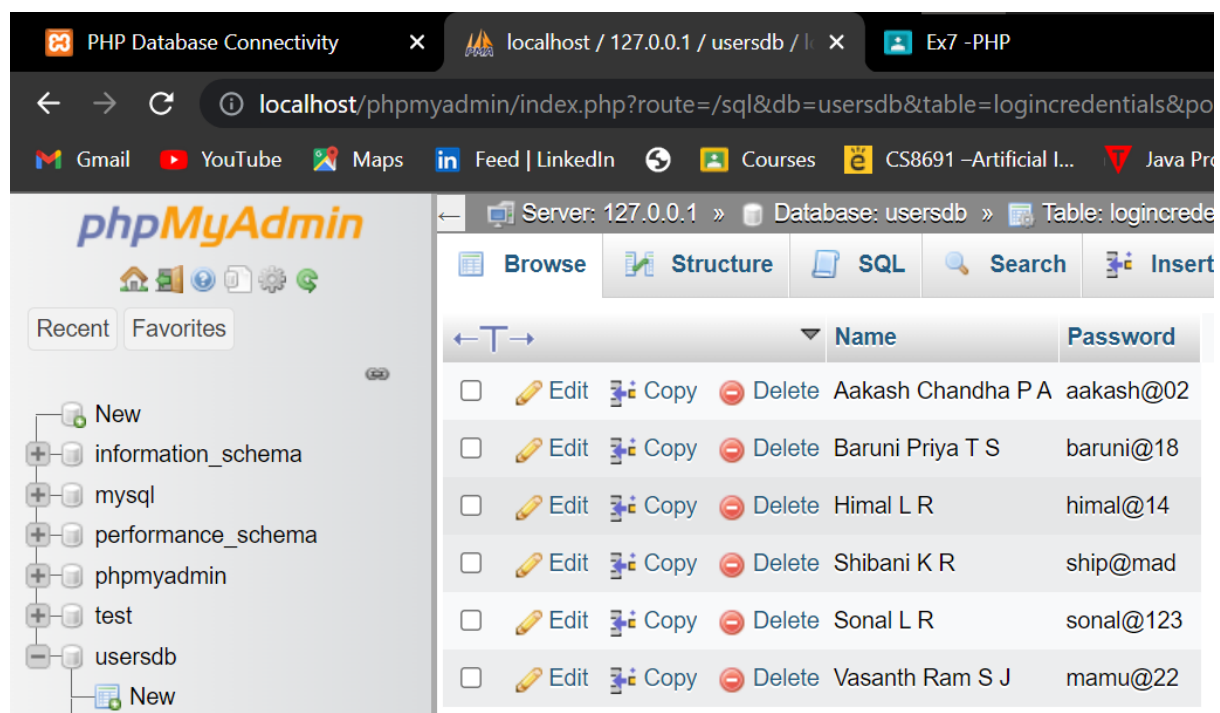
```
<html>
<head>
  <title>PHP Database Connectivity</title>
  <style>
    .des {
      display: block;
      background-color: lightblue;
      width: 40%;
    }
  </style>
</head>
<?php
  error_reporting(0);
  $error = array();
  if (isset($_POST["submit-btn"])) {
    //var_dump($_POST);
    $uname=$_POST["usr"];
    $pw=$_POST["pw"];
    //echo "uname $uname";
    //echo "pw $pw";
    $conn=new mysqli("localhost","root","","UsersDB");
    if($conn->connect_error){
      die("connection failed : ".$conn->connect_error);
    }
    //echo "successful";
    $sql="SELECT Password from logincredentials where Name='".$uname."'";
    $result=$conn->query($sql);
```

```

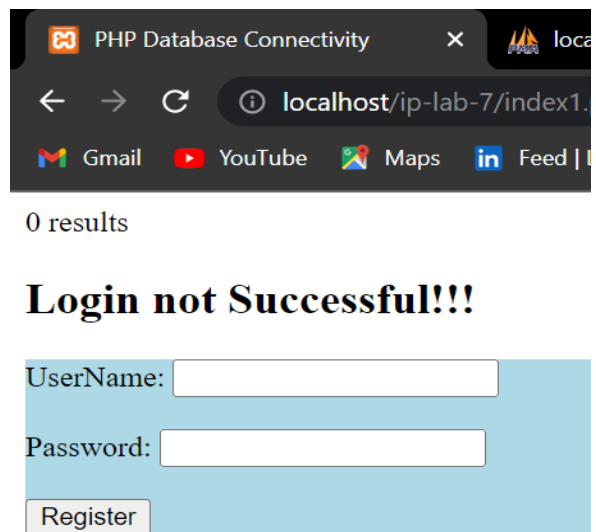
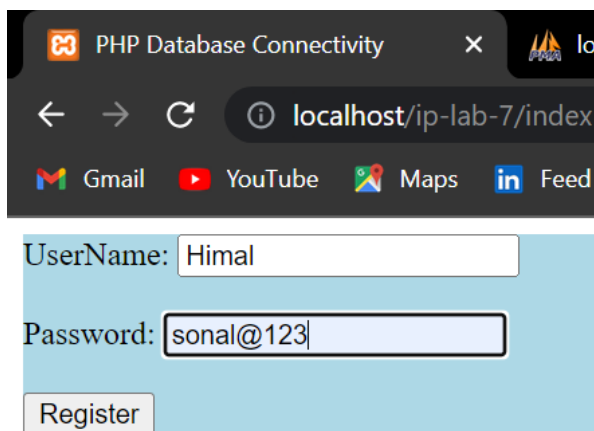
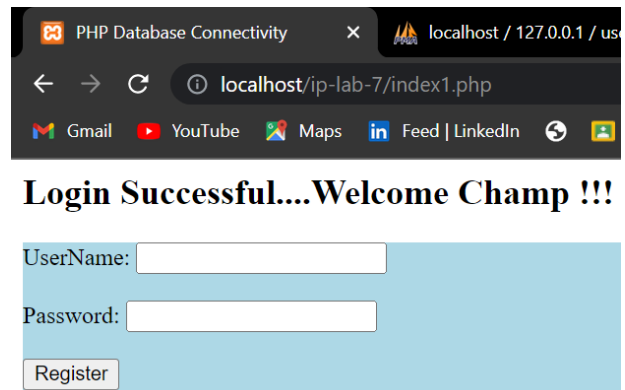
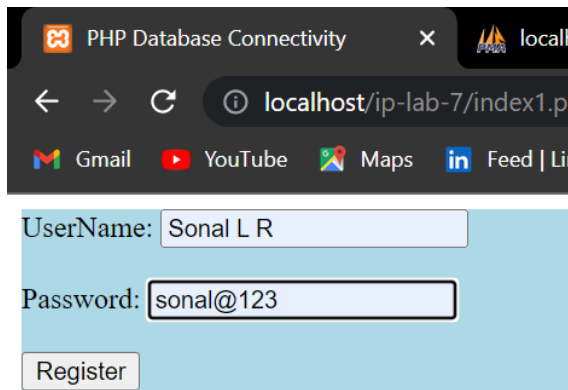
        if($result->num_rows>0){
            while($row=$result->fetch_assoc()){
                $scripw=$row["Password"];
            }
        }
        else{
            echo "0 results"; // name not matched
        }
        if(strcmp($pw,$scripw)==0){
            echo "<h2>Login Successful....Welcome Champ !!!</h2>";
        }
        else{
            echo "<h2>Login not Successful!!!</h2>";
        }
    }
?>
<div class = "des">
<body>
    <form action="index1.php" method="post">
        <label>UserName: </label>
        <input type="text" name="usr"><br><br>
        <label>Password: </label>
        <input type="text" name="pw"><br><br>
        <input type="submit" value="Register" name="submit-btn" >
    </form>
</body>
</html>

```

OUTPUT:



	Name	Password
<input type="checkbox"/> Edit Copy Delete	Aakash Chandha P A	aakash@02
<input type="checkbox"/> Edit Copy Delete	Baruni Priya T S	baruni@18
<input type="checkbox"/> Edit Copy Delete	Himal L R	himal@14
<input type="checkbox"/> Edit Copy Delete	Shibani K R	ship@mad
<input type="checkbox"/> Edit Copy Delete	Sonal L R	sonal@123
<input type="checkbox"/> Edit Copy Delete	Vasanth Ram S J	mamu@22



Observation	
Record	
Total	
Sign	

RESULT:

Thus, the login form using PHP was created and executed successfully.

Create and save an XML document at the server, which contains 10 users Information. Write a Program, which takes user Id as an input and returns the User details by taking the user information from the XML document.

AIM:

To write AJAX program for creating and saving an XML document at the server.

ALGORITHM:

1. Create an index.html file and fix the form.
2. Get roll number through the form
3. Fetch the details of student corresponding to that particular roll number.
4. Display the details below the form

PROGRAM:

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>AJAX User Information</title>
  </head>
  <body>
    <span><label>Enter Roll Number</label><span>
    <input type="text" id="roll" name="roll">
    <button onclick="fetch()">Search</button>
    <h1 id="demo"></h1>
    <script>
    function fetch(){
      var txt="";
      var xhttp=new XMLHttpRequest();
      //document.getElementById("demo").innerHTML="hello";
      xhttp.onreadystatechange=function(){
        if(this.readyState==4 && this.status==200){
          //alert("state changed");
          var xmlDoc=xhttp.responseXML;
          var x=xmlDoc.getElementsByTagName("student");
          //txt="hi,"+x[0].childNodes[2].nodeValue;
          var roll=document.getElementById("roll").value;
          for(var i=0;i<x.length;i++){

            if(roll==x[i].getElementsByTagName("Roll")[0].childNodes[0].nodeValue){
```

```

                txt+="Name :
"+x[i].getElementsByTagName("Name")[0].childNodes[0].nodeValue+ "<br>
Department : "+x[i].getElementsByTagName("Dept")[0].childNodes[0].nodeValue+
"<br> Cgpa : "+x[i].getElementsByTagName("cgpa")[0].childNodes[0].nodeValue;
            }
        }
        document.getElementById("demo").innerHTML=txt;
    }
};
xhttp.open("GET","student.xml",true);
xhttp.send();
}
</script>
</body>
</html>

```

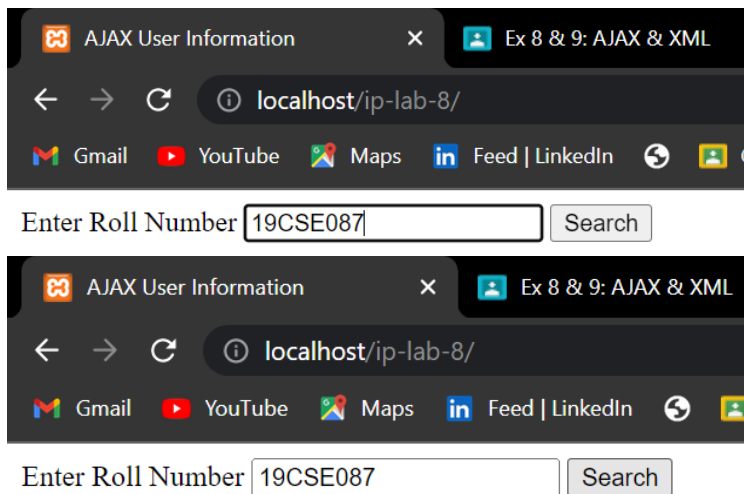
student.xml:

```

<?xml version="1.0" encoding="UTF-8" ?>
<college>
    <student>
        <Roll>19CSE087</Roll>
        <Name>Sonal</Name>
        <Dept>CSE</Dept>
        <cgpa>8.9</cgpa>
    </student>
    <student>
        <Roll>19ECE066</Roll>
        <Name>Baruni</Name>
        <Dept>ECE</Dept>
        <cgpa>9.8</cgpa>
    </student>
    <student>
        <Roll>19CSE096</Roll>
        <Name>Aakash</Name>
        <Dept>CSE</Dept>
        <cgpa>9.4</cgpa>
    </student>
    <student>
        <Roll>19IT128</Roll>
        <Name>Vasanth</Name>
        <Dept>IT</Dept>
        <cgpa>8.6</cgpa>
    </student>
    <student>
        <Roll>19EEE073</Roll>
        <Name>Shibani</Name>
        <Dept>EEE</Dept>
        <cgpa>8.2</cgpa>
    </student>
</college>

```

OUTPUT:



Enter Roll Number

Enter Roll Number

Name : Sonal
Department : CSE
Cgpa : 8.9

Observation	
Record	
Total	
Sign	

RESULT:

Thus, the AJAX program for creating and saving an XML document at the server was written and executed successfully.

Create an XML document to store the details of 10 students like name, Roll no, department and marks. Create a XSLT to display the XML document in table format. Highlight the students who belongs to CSE department.

AIM:

To write an XML and XSL Transformation program for storing students details and displaying in table format.

ALGORITHM:

1. Create an student.xml file with required data.
2. Create a studstyle.xsl file for xsl transformation.
3. Link both the files via a common tag.
4. Display the XML Data in Table format.
5. Highlight the records belonging to CSE department in pink colour.

PROGRAM:

studentstyle.xsl:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html"/>
<xsl:template match="/">
  <html>
    <head>
      <title>XML and XSL</title>
    </head>
    <body>
      <h1>Student Details</h1>
      <table border="1">
        <tr>
          <th>Name</th>
          <th>Department</th>
          <th>CGPA</th>
        </tr>
        <xsl:for-each select="college/student">
          <xsl:choose>
            <xsl:when test="Department = 'CSE'">
              <tr bgcolor="pink">
                <td>
                  <xsl:value-of select="Name"/>
                </td>
                <td>
                  <xsl:value-of select="Department"/>
                </td>
                <td>
```

```

        <xsl:value-of select="CGPA"/>
      </td>
    </tr>
  </xsl:when>
  <xsl:otherwise>
    <tr>
      <td>
        <xsl:value-of select="Name"/>
      </td>
      <td>
        <xsl:value-of select="Department"/>
      </td>
      <td>
        <xsl:value-of select="CGPA"/>
      </td>
    </tr>
  </xsl:otherwise>
</xsl:choose>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

student.xml:

```

<?xml version="1.0" encoding="UTF-8" ?>
<?xml-stylesheet type="text/xsl" href="studentstyle.xml"?>
<college>
  <student>
    <Roll>19CSE087</Roll>
    <Name>Sonal</Name>
    <Dept>ECE</Dept>
    <cgpa>8.9</cgpa>
  </student>
  <student>
    <Roll>19ECE066</Roll>
    <Name>Baruni</Name>
    <Dept>ECE</Dept>
    <cgpa>9.8</cgpa>
  </student>
  <student>
    <Roll>19CSE096</Roll>
    <Name>Aakash</Name>
    <Dept>CSE</Dept>
    <cgpa>9.4</cgpa>
  </student>
  <student>
    <Roll>19IT128</Roll>
    <Name>Vasanth</Name>
  </student>
</college>

```

```

    <Dept>CSE</Dept>
    <cgpa>8.6</cgpa>
</student>
<student>
    <Roll>19EEE073</Roll>
    <Name>Shibani</Name>
    <Dept>EEE</Dept>
    <cgpa>8.2</cgpa>
</student>
</college>

```

OUTPUT:



Student Details

Name	Department	CGPA
Sonal	CSE	8.9
Baruni	ECE	9.8
Aakash	CSE	9.4
Vasanth	IT	8.6
Shibani	EEE	8.2

Observation	
Record	
Total	
Sign	

RESULT:

Thus, an XML and XSL transformation program for storing students details and displaying in table format was written and executed successfully.