

Program 1:

write a html program to display welcome message

first.html:

```
<html>
  <head>
    <title>my first html program</title>
  </head>
  <body>
    <center>
      <p>welcome to html programming </p>
    </center>
  </body>
</html>
```

Output:



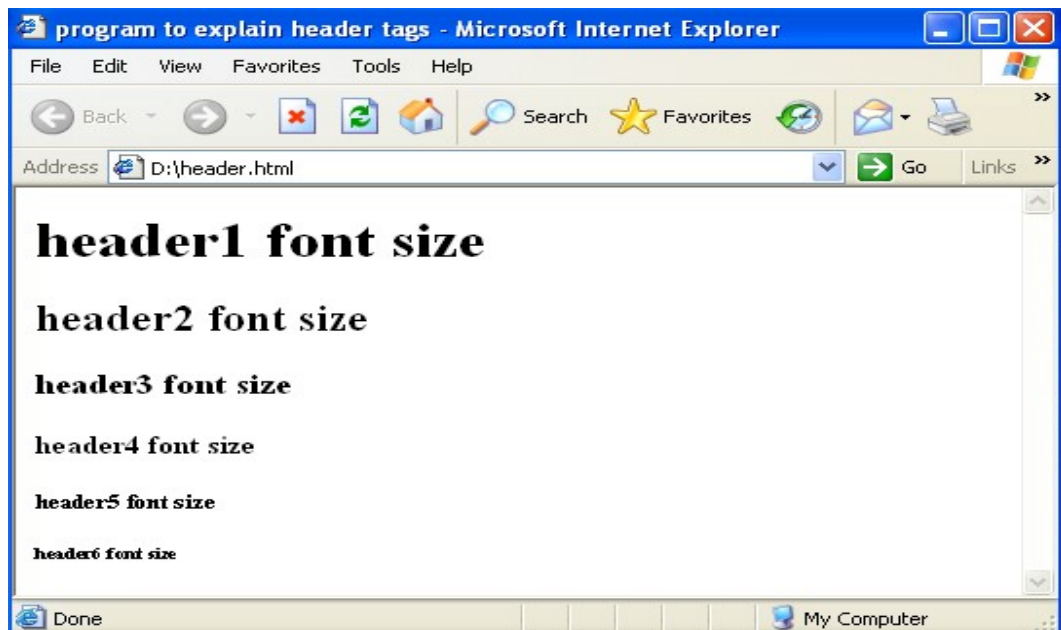
Program 2:

write a html program using header tags

header.html :

```
<html>
  <head>
    <title>program to explain header tags</title>
  </head>
  <body>
    <h1>header1 font size </h1>
    <h2>header2 font size </h2>
    <h3>header3 font size </h3>
    <h4>header4 font size </h4>
    <h5>header5 font size </h5>
    <h6>header6 font size </h6>
  </body>
</html>
```

Output:



Program 3:

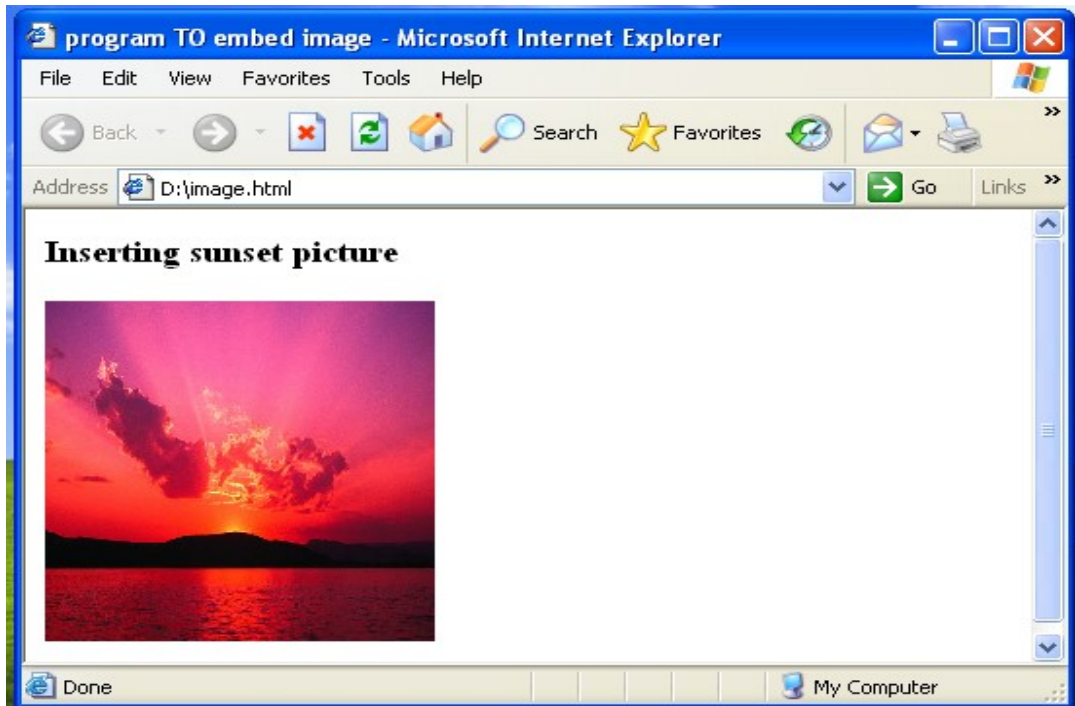
write a html program to embed an image into web document

image.html :

```
<html>
  <head>
    <title>program TO embed image</title>
  </head>
```

```
<body>
  <h3>Inserting sunset picture</h3>
  
</body>
</html>
```

Output:



Program 4:

write a html program to create hyperlinks to other documents

first.html:

```
<html>
  <head>
    <title>my first html program</title>
  </head>
  <body>
    <center>
      <p>welcome to html programming </p>
    </center>
  </body>
</html>
```

Header.html:

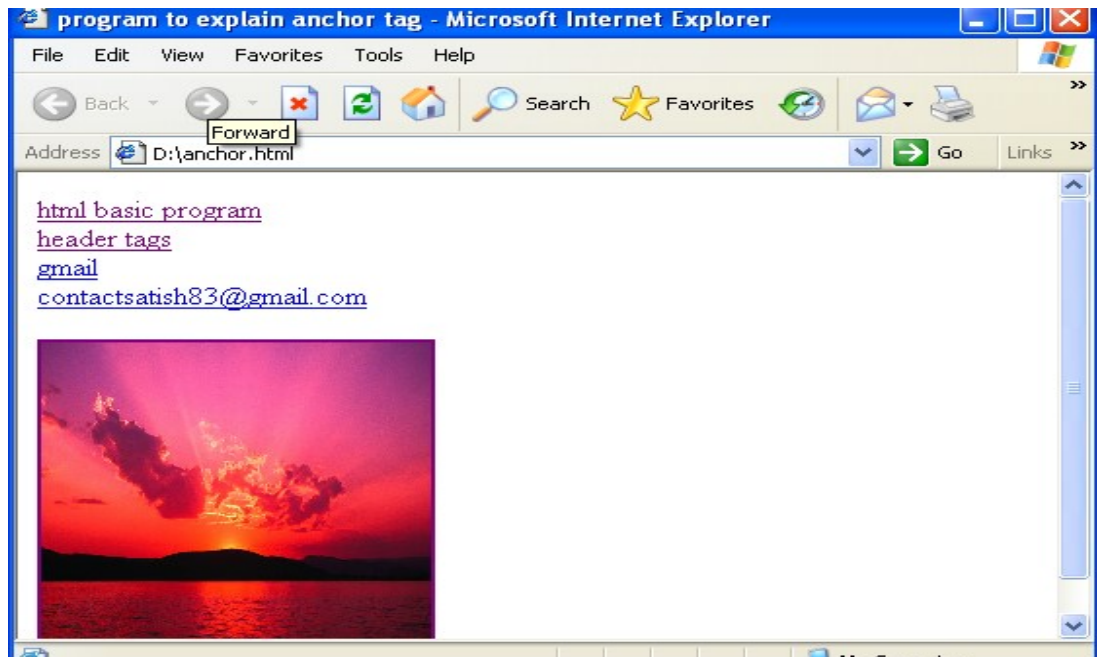
```
<html>
  <head>
    <title>program to explain header tags</title>
  </head>
  <body>
    <h1>header1 font size </h1>
    <h2>header2 font size </h2>
    <h3>header3 font size </h3>
    <h4>header4 font size </h4>
    <h5>header5 font size </h5>
    <h6>header6 font size </h6>
  </body>
</html>
```

anchor.html:

```
<html>
  <head>
    <title>program to explain anchor tag</title>
  </head>

  <body>
    <a href="first.html"> html basic program </a> <br/>
    <a href="header.html"> header tags </a> <br/>
    <a href="http://www.gmail.com"> gmail </a> <br/>
    <a href="mailto:contactsatish83@gmail.com">
      <a href="mailto:contactsatish83@gmail.com">
        <u>contactsatish83@gmail.com</u>
      </a>
    </a>
    <br/><br/>
    <a href="first.html">
      
    </a>
  </body>
</html>
```

Output:



Program 5:

write a html program to create ordered lists, unordered lists, nested lists

ordered lists:

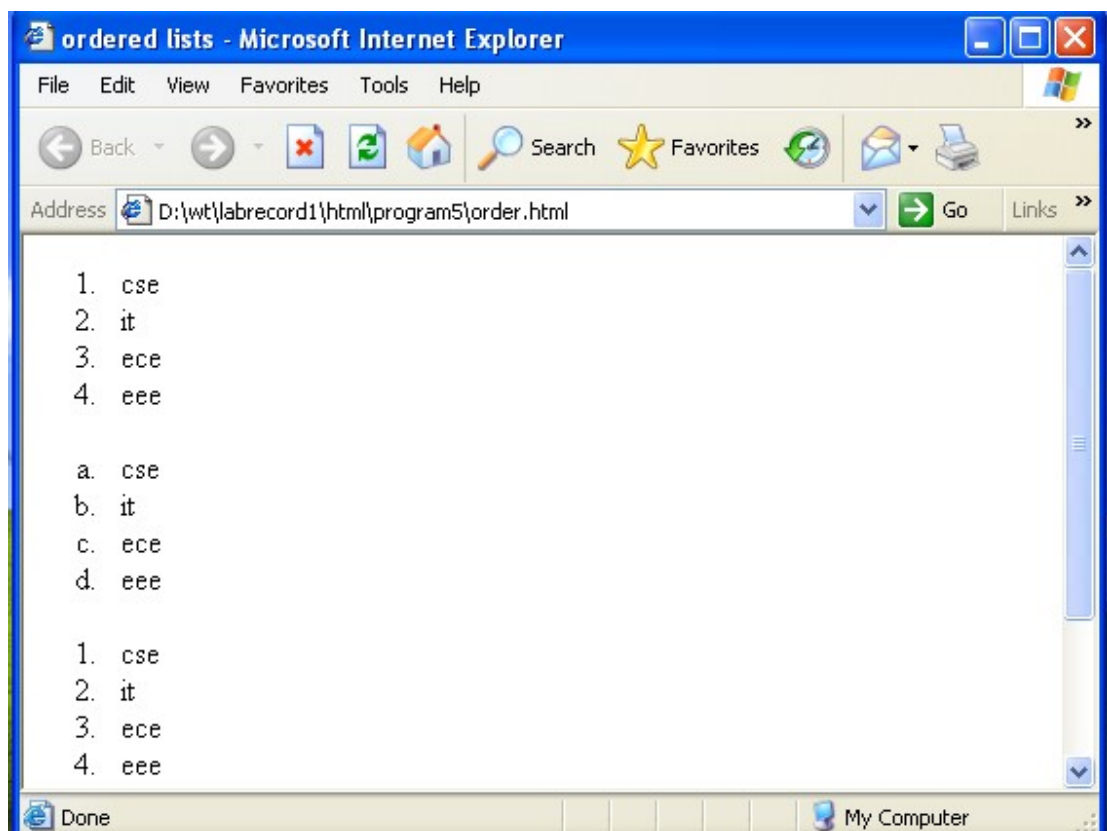
order.html:

```
<html>
  <head>
    <title>ordered lists</title>
  </head>

  <body>
    <ol>
      <li>cse</li>
      <li>it</li>
      <li>ece</li>
      <li>eee</li>
    </ol>
    <ol type="a">
```

```
        <li>cse</li>
        <li>it</li>
        <li>ece</li>
        <li>eee</li>
    </ol>
    <ol type="1">
        <li>cse</li>
        <li>it</li>
        <li>ece</li>
        <li>eee</li>
    </ol>
    <ol type="i">
        <li>cse</li>
        <li>it</li>
        <li>ece</li>
        <li>eee</li>
    </ol>
</body>
</html>
```

Output:

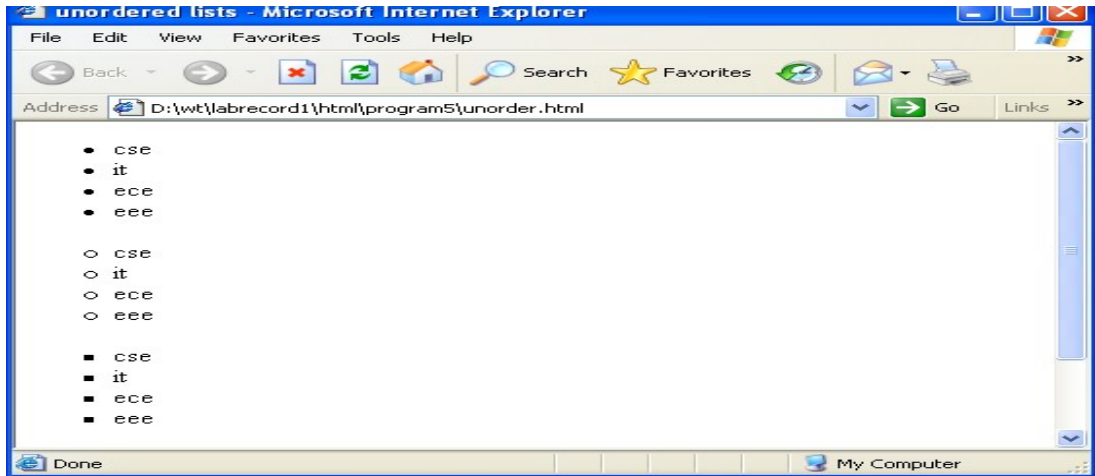


Unordered lists:

Unorder.html:

```
<html>
  <head>
    <title>unordered lists</title>
  </head>
  <body>
    <ul>
      <li>cse</li>
      <li>it</li>
      <li>ece</li>
      <li>eee</li>
    </ul>
    <ul type="circle">
      <li>cse</li>
      <li>it</li>
      <li>ece</li>
      <li>eee</li>
    </ul>
    <ul type="square">
      <li>cse</li>
      <li>it</li>
      <li>ece</li>
      <li>eee</li>
    </ul>
    <ul type="disc">
      <li>cse</li>
      <li>it</li>
      <li>ece</li>
      <li>eee</li>
    </ul>
  </body>
</html>
```

Output:



Nested lists:

Nested1.html:

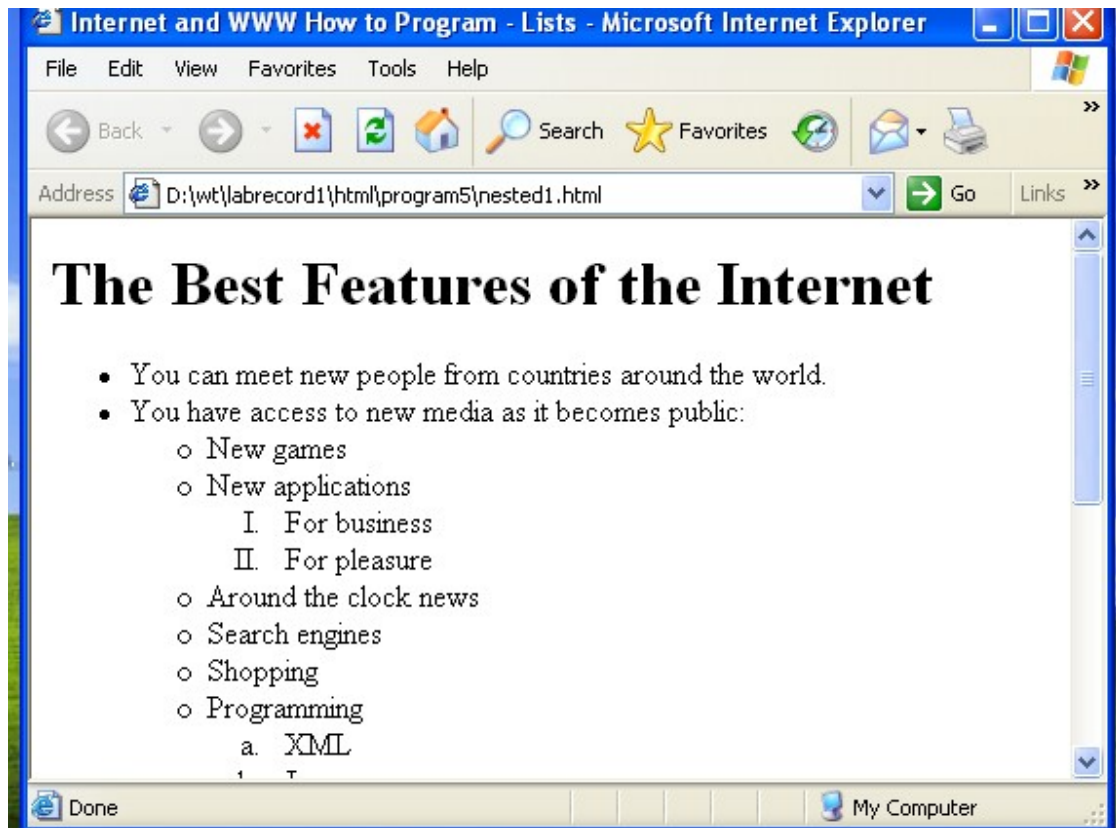
```
<html>
  <head>
    <title>Internet and WWW How to Program - Lists
    </title>
  </head>
  <body>
    <h1>The Best Features of the Internet</h1>
    <ul>
      <li>You can meet new people from countries around
        the world.</li>
      <li> You have access to new media as it becomes
        public:
        <ul>
          <li>New games</li>
          <li>
            New applications
            <ol type = "I">
              <li>For business</li>
              <li>For pleasure</li>
            </ol>
          </li>
          <li>Around the clock news</li>
          <li>Search engines</li>
          <li>Shopping</li>
          <li>
            Programming
            <ol type = "a">
              <li>XML</li>
            </ol>
          </li>
        </ul>
      </li>
    </ul>
  </body>
</html>
```



```
        <li>Java</li>
        <li>XHTML</li>
        <li>Scripts</li>
        <li>New languages</li>
    </ol>
</li>
</ul>
</li>

    <li>Links</li>
    <li>Keeping in touch with old friends</li>
    <li>It is the technology of the future!</li>
</ul>
<h1>My 3 Favorite <em>CEOs</em></h1>
<ol>
    <li>Harvey Deitel</li>
    <li>Bill Gates</li>
    <li>Michael Dell</li>
</ol>
</body>
</html>
```

Output:



Program 6:

write a html program to create Time Table using tables

table2.html:

```
<html>
  <head>
    <title>html tables</title>
  </head>
  <body>
    <table border="5" width="800">
      <caption>
        <strong>
          <h2>Time Table</h2>
        </strong>
      </caption>
      <colgroup>
        <col align="right" span="8"/>
      </colgroup>
      <thead>
```

```

<tr>
  <th>DAY/TIME</th>
  <th>09.45-10.35</th>
  <th>10.35-11.25</th>
  <th>11.25-12.15</th>
  <th>12.15-12.55</th>
  <th>1.00-1.50</th>
  <th>1.50-2.40</th>
  <th>2.40-3.30</th>
</tr>
</thead>
<tbody>
  <tr>
    <th>MONDAY</th>
    <td align="center">WT</td>
    <td align="center">ACA</td>
    <td align="center">DWDM</td>
    <td rowspan="7"
      align="center"><h3>LUNCH</h3></td>
    <td colspan="3" align="center">WT/NP
      LAB</td>
  </tr>
  <tr>
    <th>TUESDAY</th>
    <td align="center">WT</td>
    <td align="center">DWDM</td>
    <td align="center">ACA</td>
    <td align="center">SPM</td>
    <td align="center">NP</td>
    <td align="center">MT</td>
  </tr>
  <tr>
    <th>WEDNESDAY</th>
    <td colspan="3" align="center">WT/NP
      LAB</td>
    <td align="center">MT</td>
    <td align="center">SPM</td>
    <td align="center">NP</td>
  </tr>
  <tr>
    <th>THURSDAY</th>
    <td align="center">MT</td>
    <td align="center">SPM</td>
    <td align="center">NP</td>
    <td align="center">WT</td>
    <td align="center">ACA</td>

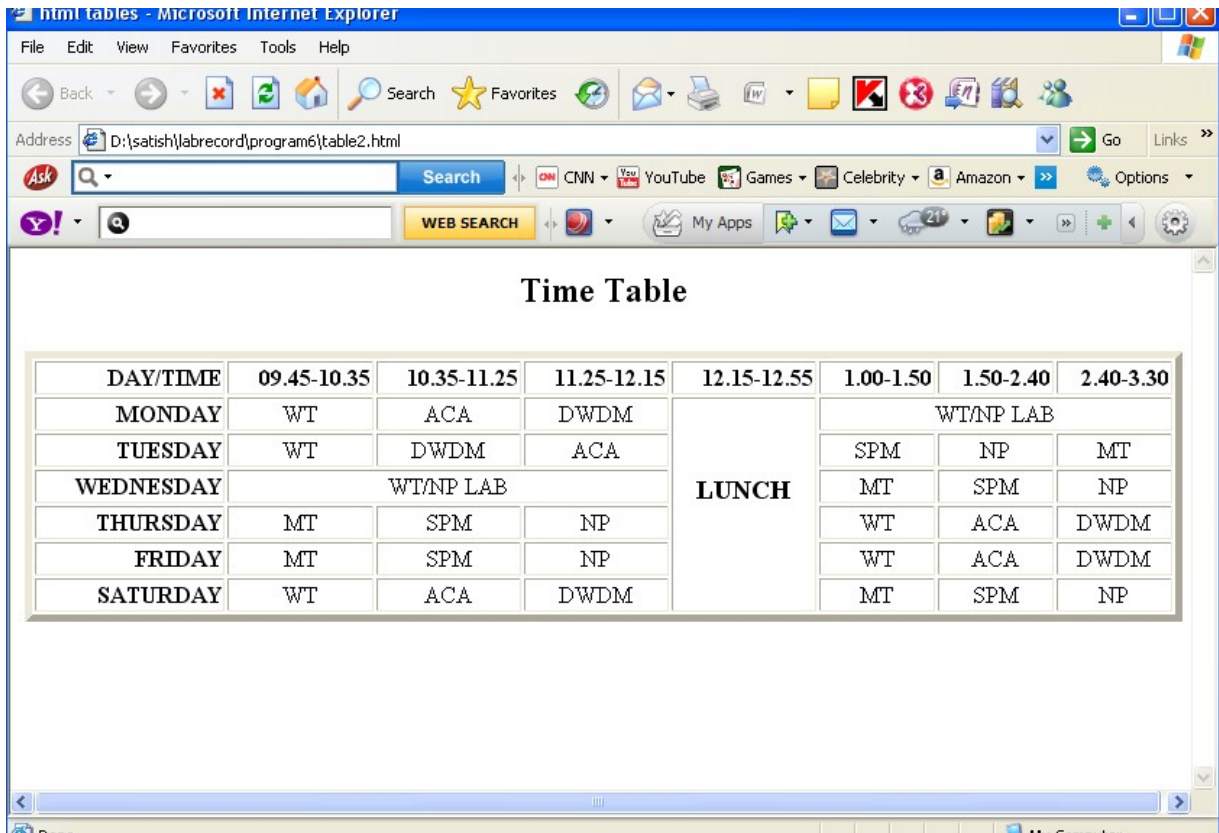
```

```

        <td align="center">DWDM</td>
</tr>
<tr>
    <th>FRIDAY</th>
    <td align="center">MT</td>
    <td align="center">SPM</td>
    <td align="center">NP</td>
    <td align="center">WT</td>
    <td align="center">ACA</td>
    <td align="center">DWDM</td>
</tr>
<tr>
    <th>SATURDAY</th>
    <td align="center">WT</td>
    <td align="center">ACA</td>
    <td align="center">DWDM</td>
    <td align="center">MT</td>
    <td align="center">SPM</td>
    <td align="center">NP</td>
</tr>
</tbody>
</table>
</body>
</html>

```

Output:



Program 7:

write a html program to create a feedback form for a website

feedback.html:

```
<html>
  <head>
    <title>Internet and WWW How to Program - Forms</title>
  </head>
  <body>
    <h1>Feedback Form</h1>
    <p>Please fill out this form to help us improve our
    site.</p>
    <form method = "post" action = "/cgi-bin/formmail">
      <p>
        <label>Name:
          <input name = "name" type = "text"
            size = "25" />
        </label>
      </p>
```

```

<p>
  <label>Comments:
    <br />
    <textarea name = "comments" rows = "4"
      cols="36">
      Enter your comments here.
    </textarea>
  </label>
</p>
<p>
  <label>E-mail Address:
    <input name = "email" type = "password"
      size = "25" />
  </label>
</p>
<p>
  <strong>Things you liked:</strong><br />
  <label>Site design
    <input name = "thingsliked" type =
      "checkbox" value = "Design"/>
  </label>
  <label>Links
    <input name = "thingsliked" type =
      "checkbox" value = "Links" />
  </label>
  <label>Ease of use
    <input name = "thingsliked" type =
      "checkbox" value = "Ease" />
  </label>
  <label>Images
    <input name = "thingsliked" type =
      "checkbox" value = "Images" />
  </label>
  <label>Source code
    <input name = "thingsliked" type =
      "checkbox" value = "Code" />
  </label>
</p>
<p>
  <strong>How did you get to oursite?:
</strong><br />
  <label>Search engine
    <input name = "howtosite" type =
      "radio" value = "search engine" checked
      = "checked" />
  </label>
  <label>Links from another site

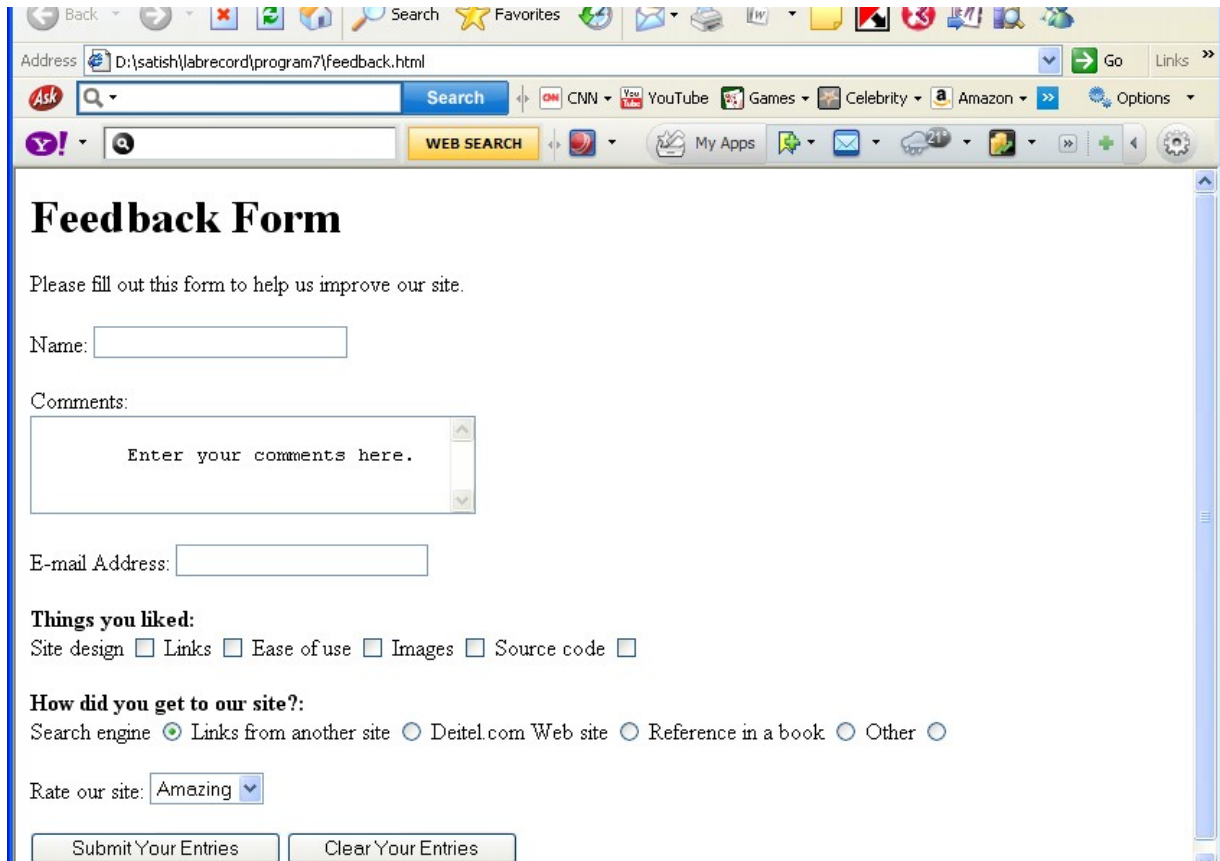
```

```

        <input name = "howtosite" type =
            "radio" value = "link" />
    </label>
    <label>Deitel.com Web site
        <input name = "howtosite" type =
            "radio" value = "deitel.com" />
    </label>
    <label>Reference in a book
        <input name = "howtosite" type =
            "radio" value = "book" />
    </label>
    <label>Other
        <input name = "howtosite" type =
            "radio" value = "other" />
    </label>
</p>
<p>
<label>Rate our site:
    <select name = "rating">
        <option selected = "selected">
            Amazing
        </option>
        <option>10</option>
        <option>9</option>
        <option>8</option>
        <option>7</option>
        <option>6</option>
        <option>5</option>
        <option>4</option>
        <option>3</option>
        <option>2</option>
        <option>1</option>
        <option>Awful</option>
    </select>
</label>
</p>
<p>
    <input type = "submit" value ="Submit Your
    Entries" />
    <input type = "reset" value = "Clear Your
    Entries" />
</p>
</form>
</html>

```

Output:



The screenshot shows a web browser window with the address bar containing "D:\satis\labrecord\program7\feedback.html". The browser's toolbar includes a search bar with "Ask" and "Search" buttons, and several navigation and utility icons. The main content area displays a "Feedback Form" with the following elements:

- Feedback Form**
- Please fill out this form to help us improve our site.
- Name:
- Comments:
- E-mail Address:
- Things you liked:**
Site design Links Ease of use Images Source code
- How did you get to our site?:**
Search engine Links from another site Deitel.com Web site Reference in a book Other
- Rate our site:
- Submit Your Entries Clear Your Entries

Program 8:

write a html program to create frames with in a web document

frame1.html:

```
<html>
  <head>
    <title>Internet and WWW How to Program -frames</title>
  </head>
  <frameset cols = "160,*">
    <frame name = "leftframe" src = "nav.html" />
    <frame name = "main" src = "first.html" />
  <noframes>
    <p>
      This page uses frames, but your browser
      does not support them.
    </p>
    <p>
      Please, <a href = "nav.html">follow this
      link to browse our site without frames</a>.
    </p>
  </noframes>
</frameset>
</html>
```

nav.html:

```
<html>
  <head>
    <title>nav.html</title>
  </head>
  <body>
    <a href="first.html" target="main">
      first program</a><br/>
    <a href="header.html" target="main">
      header program</a><br/>
  </body>
</html>
```

first.html:

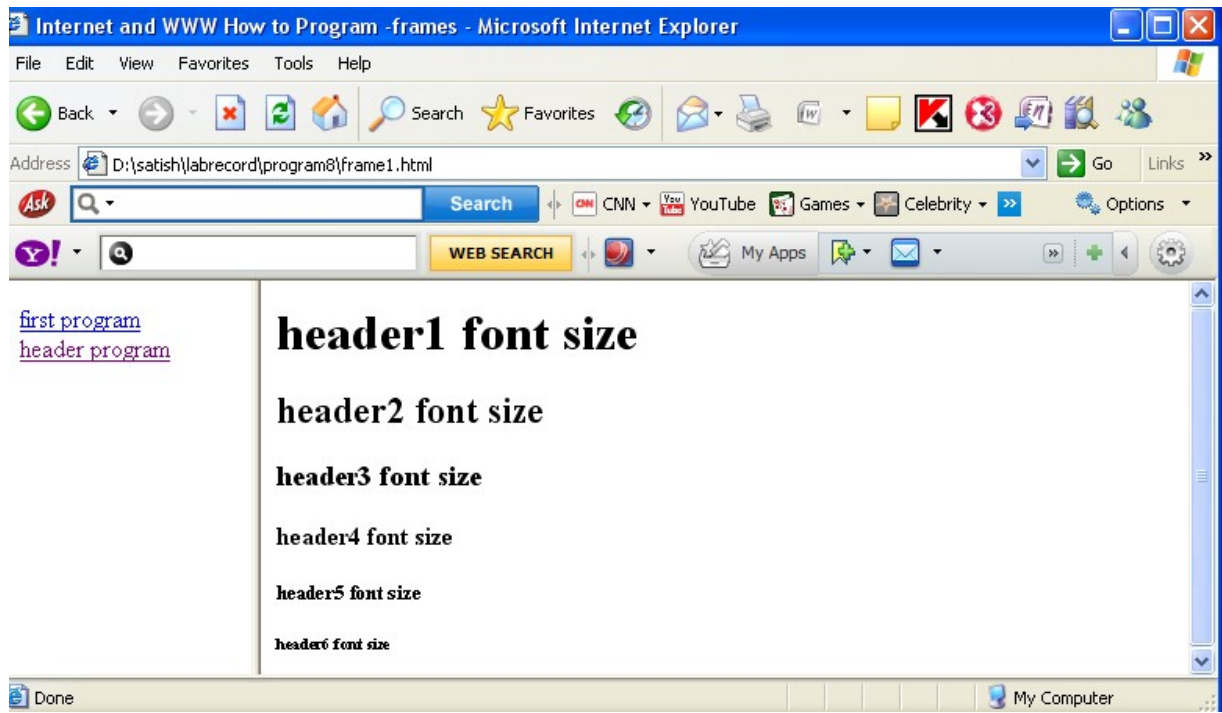
```
<html>
  <head>
    <title>my first html program</title>
```

```
</head>
<body>
  <center>
    <p>welcome to html programming </p>
  </center>
</body>
</html>
```

header.html:

```
<html>
  <head>
    <title>program to explain header tags</title>
  </head>
  <body>
    <h1>header1 font size </h1>
    <h2>header2 font size </h2>
    <h3>header3 font size </h3>
    <h4>header4 font size </h4>
    <h5>header5 font size </h5>
    <h6>header6 font size </h6>
  </body>
</html>
```

Output:



Program 9:

write a html program to create links with in the same web document(inter linking)

interlink.html:

```

<html>
  <head>
    <title>Internal linking</title>
  </head>
  <body>
    <p></p>
    <h1 id="features">The Best Features of the Internet</h1>
    <p><a href = "#ceos">Go to <em>Favorite CEOs</em></a></p>
    <ul>
      <li>You can meet people from countries around the
        world.</li>
      <li>You have access to new media as it becomes public:
        <ul>
          <li>New games</li>
          <li>New applications
            <ul>
              <li>For Business</li>
              <li>For Pleasure</li>
            </ul>
          </li>
          <li>Around the clock news</li>
          <li>Search Engines</li>
          <li>Shopping</li>
          <li>Programming
            <ul>
              <li>XHTML</li>
              <li>Java</li>
              <li>Dynamic HTML</li>
              <li>Scripts</li>
              <li>New languages</li>
            </ul>
          </li>
        </ul>
      </li>
      <li>Links</li>
      <li>Keeping in touch with old friends</li>
      <li>It is the technology of the future!</li>
    </ul><br/>
    <h1 id="ceos">My 3 Favorite <em>CEOs</em></h1>
    <p>
      <a href = "#features">Go to <em>Favorite Features</em>
    </a></p>
    <ol>
      <li>Bill Gates</li>
      <li>Steve Jobs</li>
      <li>Michael Dell</li>

```

```
</ol>
</body>
</html>
```

Output:



Program 10:

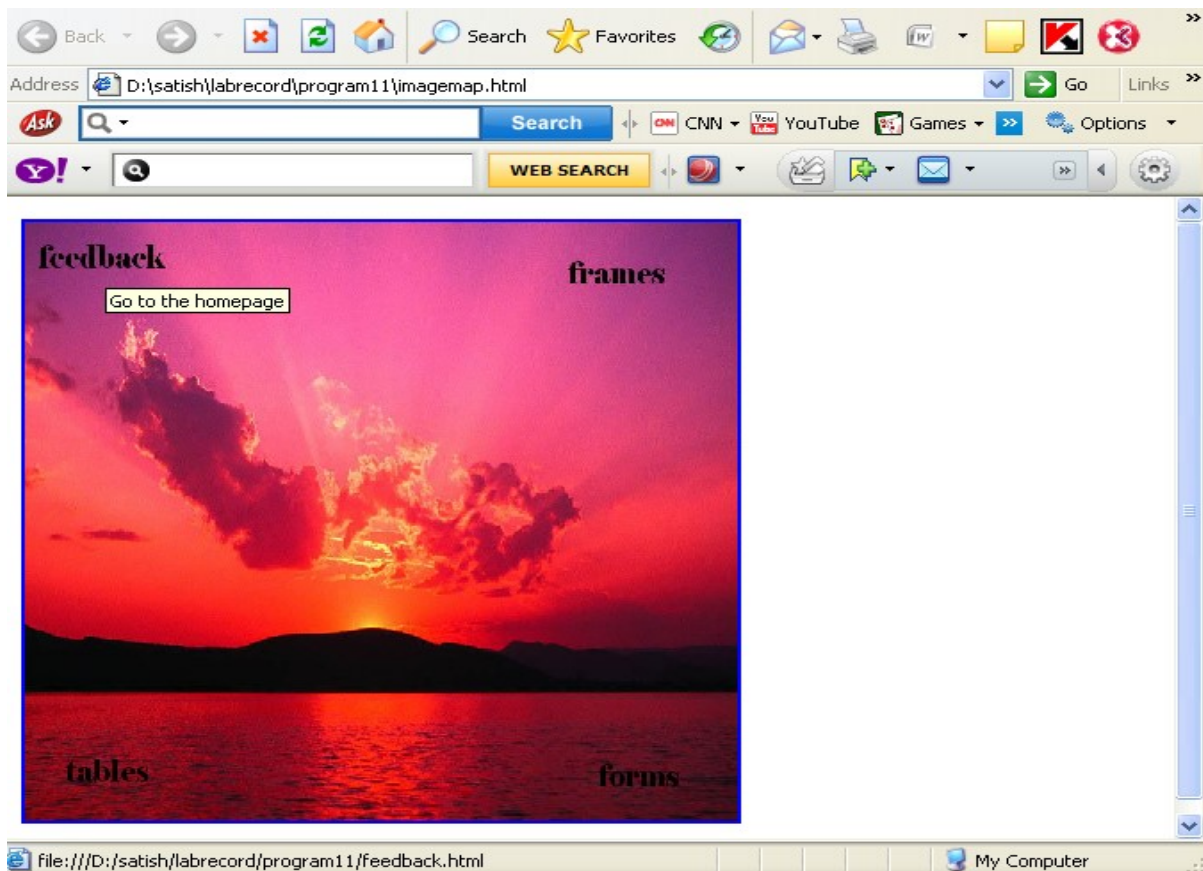
write a html program to create image maps

imagemap.html:

```
<html>
  <head>
    <title>
      Internet and WWW How to Program - Image Map
    </title>
  </head>
  <body>
    <p>
      <map id = "picture">
        <area href = "feedback.html" shape = "rect"
```

```
        coords = "3,7,72,25" alt = "Go to the
        homepage" />
<area href = "frame1.html" shape = "rect"
        coords = "300,18,356,40"
        alt = "Go to the links page" />
<area href = "table2.html" shape = "rect"
        coords = "3,360,72,370"
        alt = "Go to the feedback form" />
<area href = "form1.html" shape = "rect"
        coords = "330,365,365,380"
        alt = "Go to the contact page" />
</map>
<img src = "sunset1.jpg" width = "400" height = "400"
        alt = "Deitel logo" usemap = "#picture" />
</p>
</body>
</html>
```

output:



Program 11:

write a html program to create inline stylesheets, embedded stylesheets and external stylesheets

inline stylesheets:

inline.html:

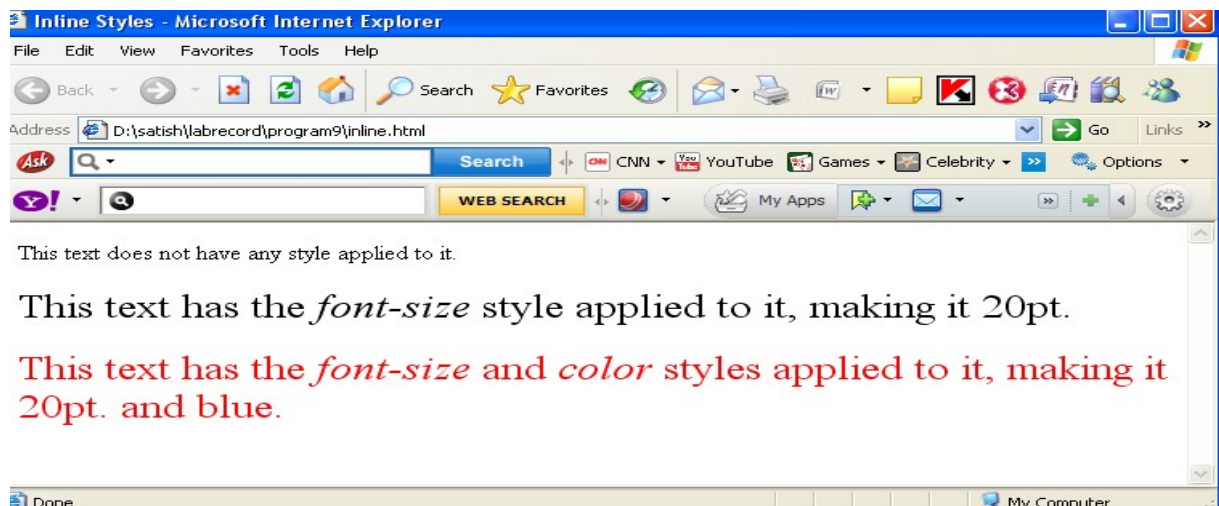
```
<html>
```

```

<head>
  <title>Inline Styles</title>
</head>
<body>
  <p>This text does not have any style applied to it.</p>
  <p style = "font-size: 20pt">This text has the
  <em>font-size</em> style applied to it, making it 20pt.
</p>
  <p style = "font-size: 20pt; color: #ff0000">
    This text has the <em>font-size</em> and
    <em>color</em> styles applied to it, making it
    20pt. and blue.</p>
</body>
</html>

```

Output:



Embedded stylesheets:

embedded2.html:

```

<html>
  <head>
    <title>embedded Styles</title>
    <style type = "text/css">
      a.nodect { text-decoration: none }
      a:hover { text-decoration: underline;
                color: red;
                background-color: #ccffcc }
      li em { color: red;
              font-weight: bold }
      ul { margin-left: 75px }
      ul ul { text-decoration: underline;

```

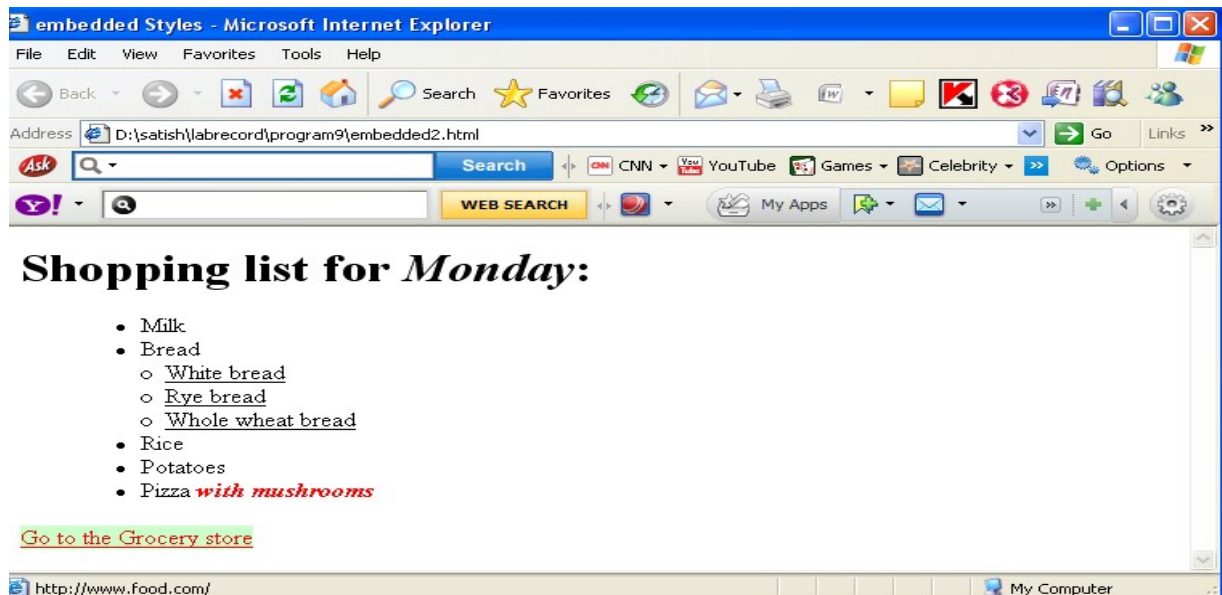


```

        margin-left: 15px }
    </style>
</head>
<body>
    <h1>Shopping list for <em>Monday</em>:</h1>
    <ul>
        <li>Milk</li>
        <li>Bread
            <ul>
                <li>White bread</li>
                <li>Rye bread</li>
                <li>Whole wheat bread</li>
            </ul>
        </li>
        <li>Rice</li>
        <li>Potatoes</li>
        <li>Pizza <em>with mushrooms</em></li>
    </ul>
    <p><a class = "nodec" href = "http://www.food.com">
    Go to the Grocery store</a></p>
</body>
</html>

```

Output:



External stylesheets:

styles.css:

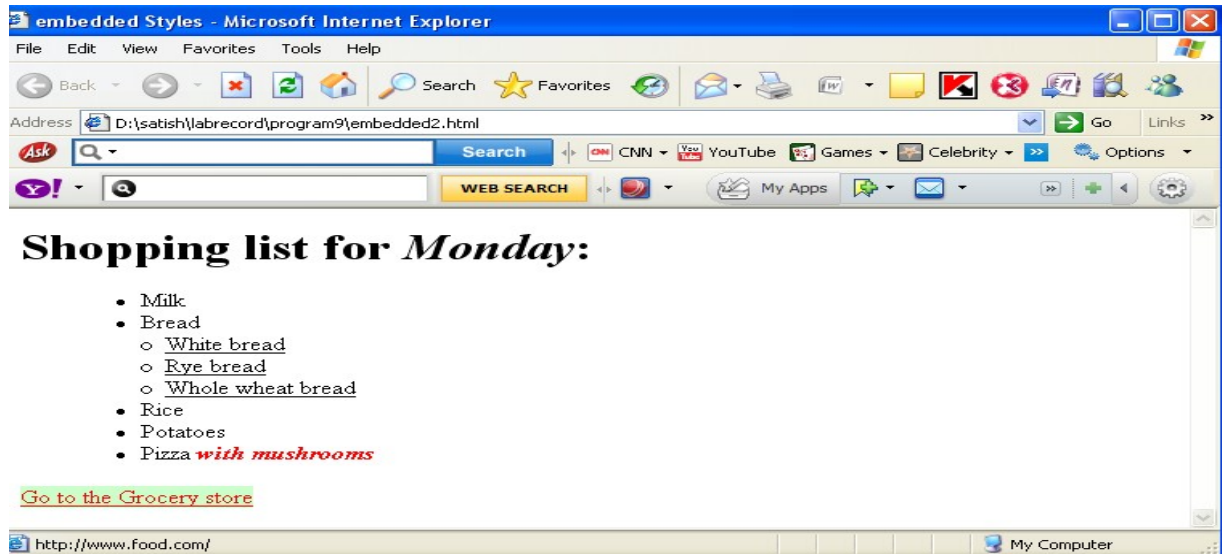
```
a { text-decoration: none }
a:hover { text-decoration: underline;
          color: red;
          background-color: #ccffcc }
li em { color: red;
        font-weight: bold;
        background-color: #ffffff }
ul { margin-left: 2cm }
ul ul { text-decoration: underline;
        margin-left: 15px }
```

external.html:

```
<html>
  <head>
    <title>More Styles</title>

    <link rel = "stylesheet" type = "text/css"
          href = "styles.css" />
  </head>
  <body>
    <h1>Shopping list for <em>Monday</em>:</h1>
    <ul>
      <li>Milk</li>
      <li>Bread
        <ul>
          <li>White bread</li>
          <li>Rye bread</li>
          <li>Whole wheat bread</li>
        </ul>
      </li>
      <li>Rice</li>
      <li>Potatoes</li>
      <li>Pizza <em>with mushrooms</em></li>
    </ul>
    <p><a class = "nodec" href = "http://www.food.com">
      Go to the Grocery store</a></p>
  </body>
</html>
```

Output:



Program 12:

Develop static pages (using Only HTML) of an online Book store. The pages should resemble: www.amazon.com. The website should consist of the following pages.

- Home page
- Registration
- User Login
- Books catalog

main.html:

```
<html>
  <head>
    <title>home page</title>
  </head>
  <body>
    <center>
      <b><h1>welcome to amazon.com</h1></b><br><br>
      <form method="post"action="login.html">
        <input type="submit" value="click"/>
        registration user login here
    </center>
```

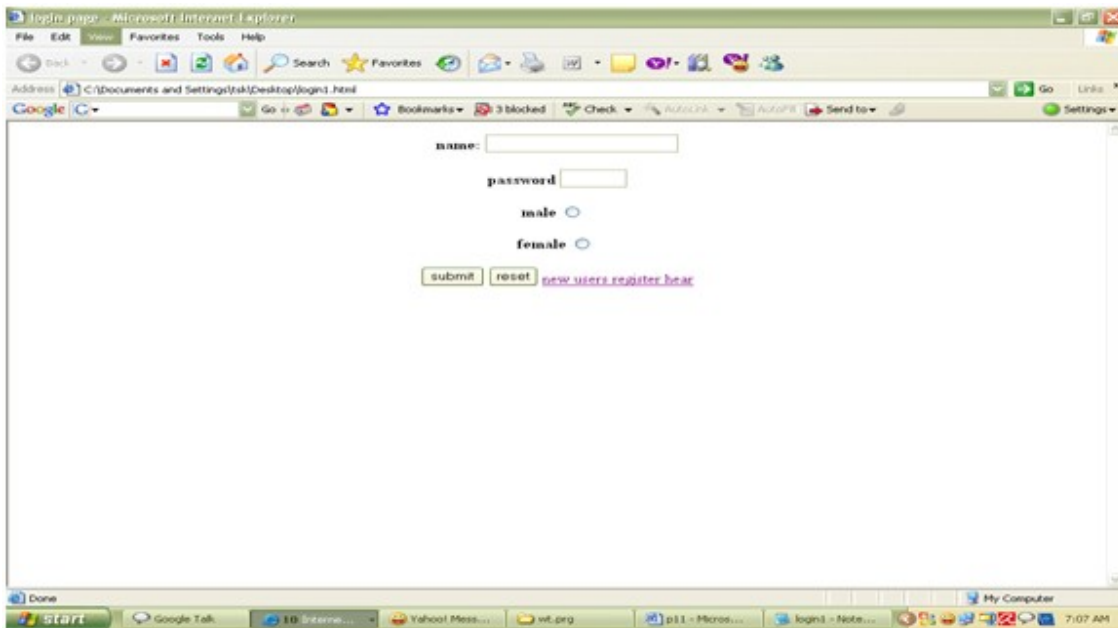


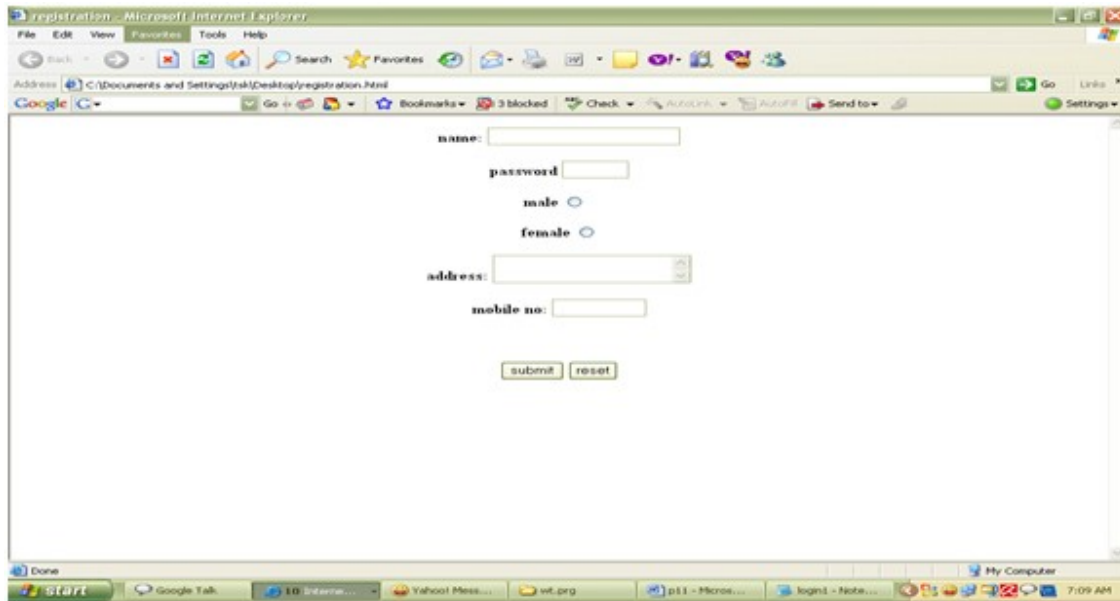
```
        </center>
    </body>
</html>
```

bookcatalog.html:

```
<html>
  <head>
    <title>books catalog</title>
  </head>
  <body>
    <center>
      <h1><p>welcome to books catalog</p></h1>
      <table border="1"width="25%"height="50%">
        <tr>
          <th>computers</th>
          <th>electronics</th>
          <th>biotech</th>
          <th>mechanical</th>
        </tr>
      </center>
    </body>
</html>
```

Output:





Program 13:

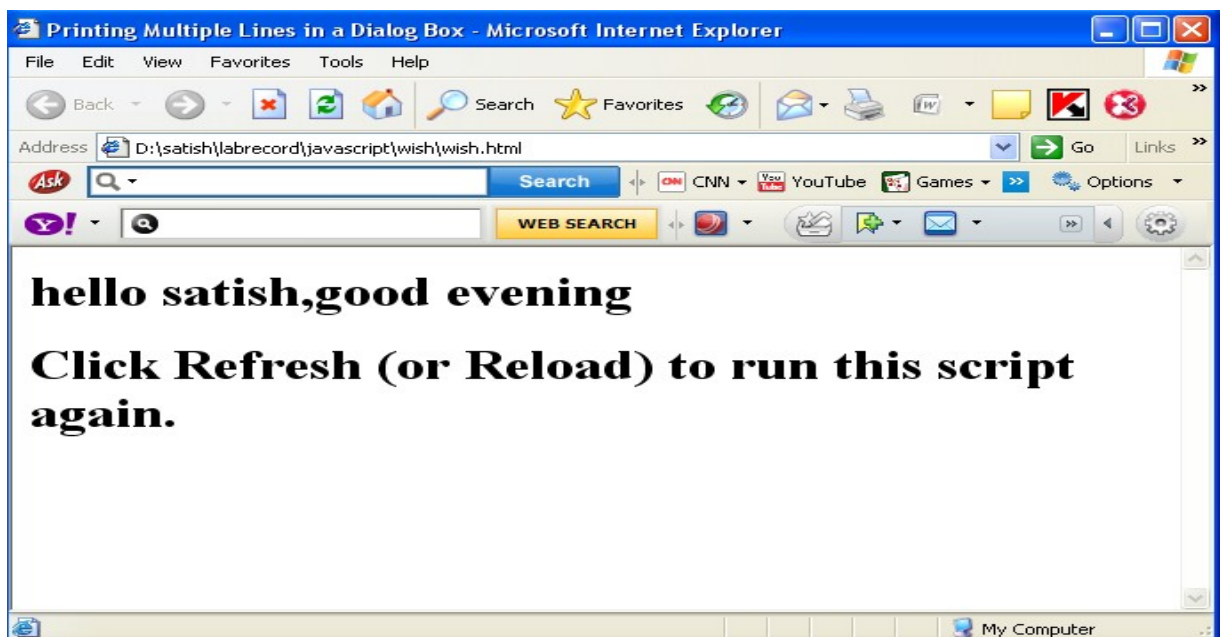
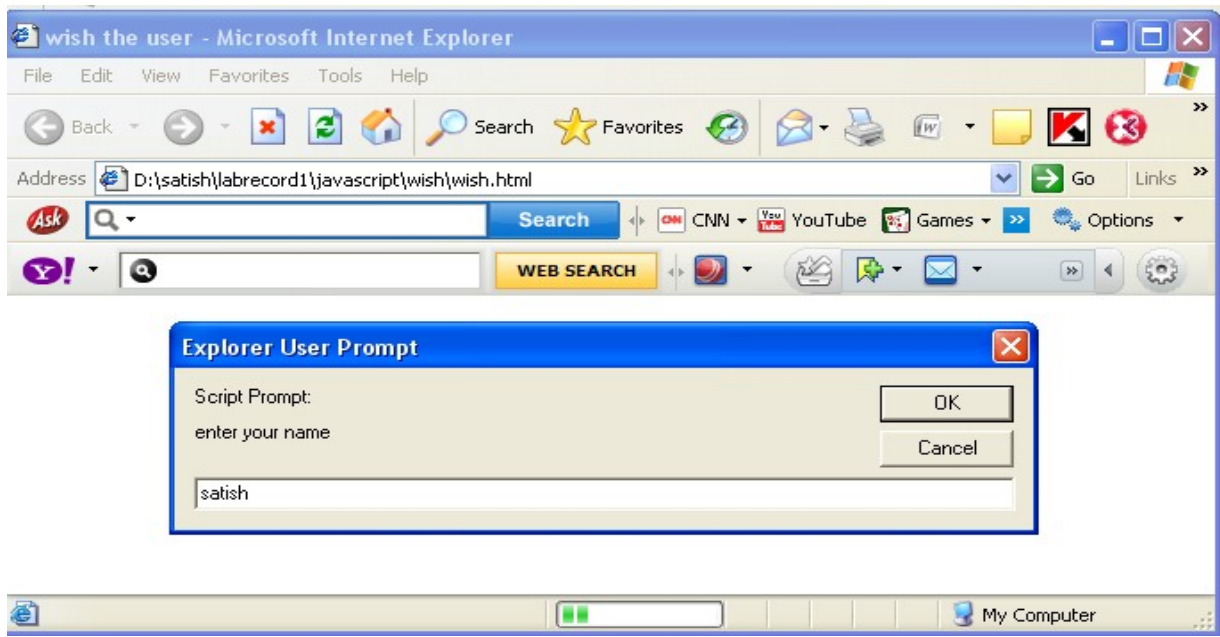
write a javascript program to wish the user according to time

wish.html:


```
<html>
  <head>
    <title>wish the user</title>
    <script type = "text/javascript">
      <!--
        var d1, hours, name;
        d1 = new Date();
        hours = d1.getHours();
        name = window.prompt("enter your name", "sat");
        if( hours < 12 )
          document.writeln("<h1>hello "+ name + ", good
            morning");
        if( hours >= 12)
        {
          hours = hours-12;
          if(hours < 6 )
            document.writeln("<h1>hello "+ name + ", good
              afternoon");
          if( hours >= 6 )
            document.writeln("<h1>hello "+ name + ", good
              evening");
        }
      // -->
    </script>
  </head>

  <body>
    <p>Click Refresh (or Reload) to run this script again.</p>
  </body>
</html>
```

Output:



Program 14:

write a javascript program to display list of items according to users choice

- 1 unorder list
- 2 order list

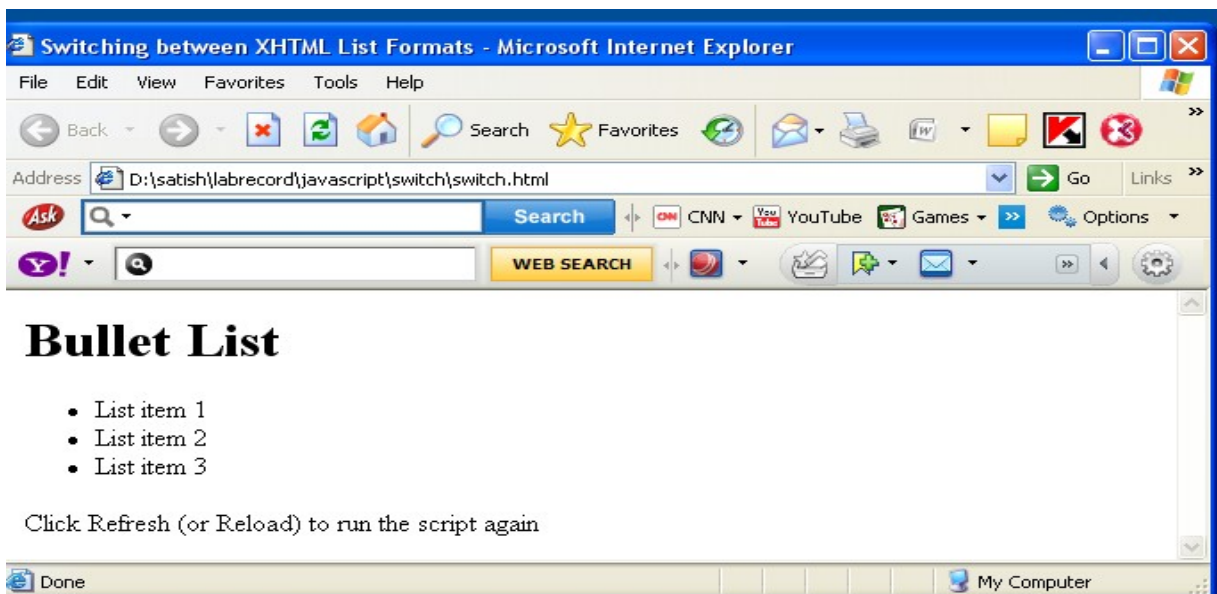
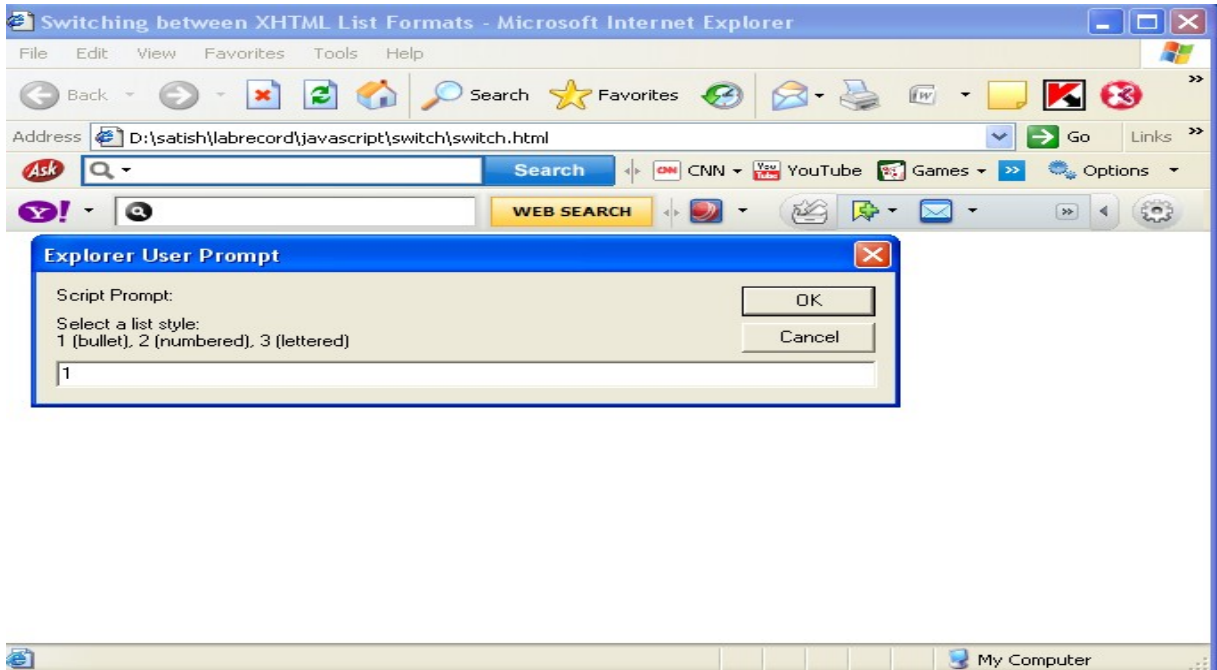
3 ordered list using alphabets

switch.html:

```
<html>
<head>
  <title>Switching between XHTML List Formats</title>
  <script type = "text/javascript">
    <!--
      var choice,startTag,endTag,validInput = true,listType;
      choice = window.prompt( "Select a list style:\n" +
        "1 (bullet), 2 (numbered), 3 (lettered)", "1" );
      switch ( choice )
      {
        case "1":
          startTag = "<ul>";
          endTag = "</ul>";
          listType = "<h1>Bullet List</h1>";
          break;
        case "2":
          startTag = "<ol>";
          endTag = "</ol>";
          listType = "<h1>Ordered List: Numbered</h1>";
          break;
        case "3":
          startTag = "<ol type = \"A\">";
          endTag = "</ol>";
          listType = "<h1>Ordered List: Lettered</h1>";
          break;
        default:
          validInput = false;
      }
      if ( validInput == true )
      {
        document.writeln( listType + startTag );
        for ( var i = 1; i <= 3; ++i )
          document.writeln( "<li>List item " + i + "</li>" );
        document.writeln( endTag );
      }
      else
        document.writeln( "Invalid choice: " + choice );
    // -->
  </script>
</head>
<body>
```

```
<p>Click Refresh (or Reload) to run the script again</p>
</body>
</html>
```

Output:



Program 15:

write a javascript program to develop craps game, the rules of the game are

- 1) In the first throw of the 2 dice if the sum of two numbers on the top of the 2 dice is equals to either 7 or 11 the user wins the game
- 2) In the first throw of the 2 dice if the sum of two numbers on the top of the 2 dice is equals to either 2 or 3 or 12 the user losses the game
- 3) In the first throw of the 2 dice if the sum of two numbers on the top of the 2 dice is equals to either 4 or 5 or 6 or 8 or 9 or 10 then the user gets that sum as his/her point and he has to continue rolling dice until he/she gets next sum equals to point which means he/she wins the game ,mean while if he/she gets sum as 7 indicates loss of game

Crapsgame.html :

```
<html>
  <head>
    <title>Program that Simulates the Game of Craps</title>
    <script type = "text/javascript">
    <!--
```

```

var WON = 0, LOST = 1, CONTINUE_ROLLING = 2;
var firstRoll=true,sumOfDice=0,myPoint=0;
var gameStatus= CONTINUE_ROLLING;
function play()
{
  if ( firstRoll )
  {
    sumOfDice = rollDice();
    switch ( sumOfDice )
    {
      case 7: case 11:
        gameStatus = WON;
        document.craps.point.value = "";
        break;
      case 2: case 3: case 12:
        gameStatus = LOST;
        document.craps.point.value = "";
        break;
      default:
        gameStatus = CONTINUE_ROLLING;
        myPoint = sumOfDice;
        document.craps.point.value = myPoint;
        firstRoll = false;
    }
  }
  else
  {
    sumOfDice = rollDice();
    if ( sumOfDice == myPoint )
      gameStatus = WON;
    else
      if ( sumOfDice == 7 )
        gameStatus = LOST;
  }
  if ( gameStatus == CONTINUE_ROLLING )
    window.status = "Roll again";
  else
  {
    if ( gameStatus == WON )
      window.status = "Player wins. " +
        "Click Roll Dice to play again.";
    else
      window.status = "Player loses. " +
        "Click Roll Dice to play again.";
    firstRoll = true;
  }
}

```

```

function rollDice()
{
    var die1, die2, workSum;
    die1 = Math.floor( 1 + Math.random() * 6 );
    die2 = Math.floor( 1 + Math.random() * 6 );
    workSum = die1 + die2;
    document.craps.firstDie.value = die1;
    document.craps.secondDie.value = die2;
    document.craps.sum.value = workSum;
    return workSum;
}
// -->
</script>
</head>

<body>

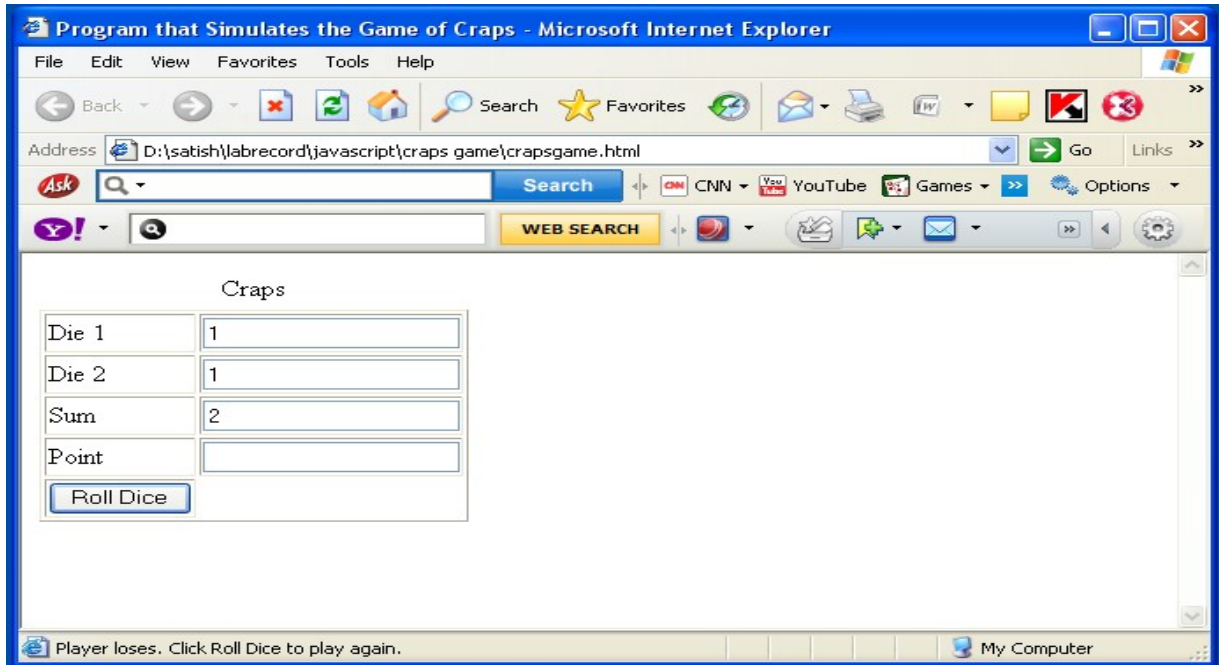
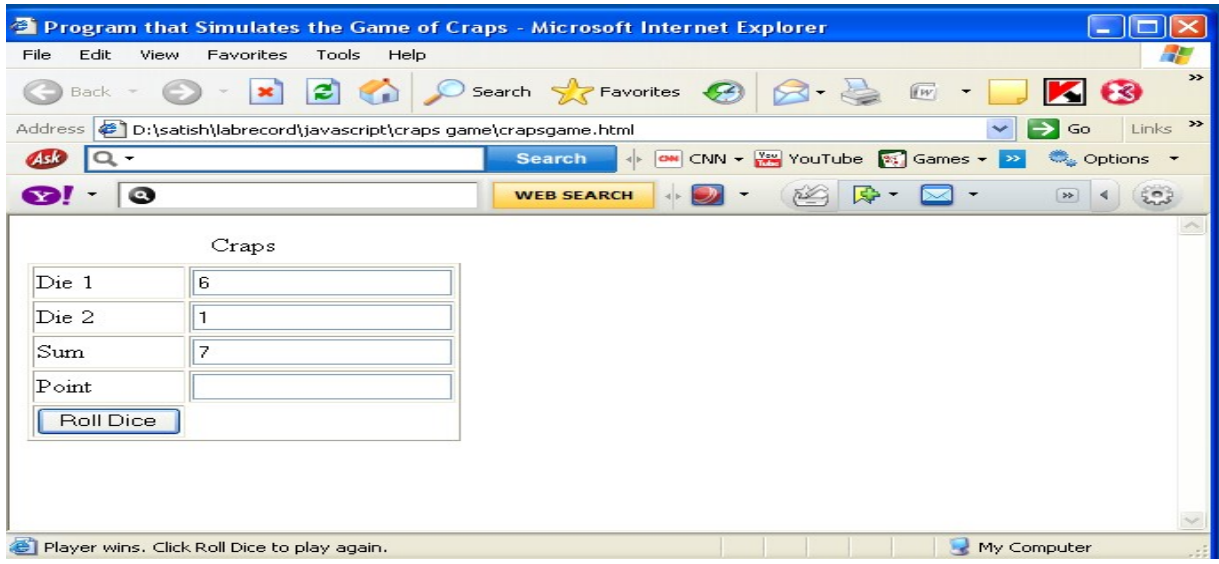
    <form name = "craps" action = "">
        <table border = "1">
            <caption>Craps</caption>
            <tr><td>Die 1</td>
            <td><input name = "firstDie" type = "text" />
            </td></tr>
            <tr><td>Die 2</td>
            <td><input name = "secondDie" type = "text" />
            </td></tr>
            <tr><td>Sum</td>
            <td><input name = "sum" type = "text" />
            </td></tr>
            <tr><td>Point</td>
            <td><input name = "point" type = "text" />
            </td></tr>
            <tr><td><input type = "button" value = "Roll Dice"
                onclick = "play()" /></td></tr>
        </table>
    </form>

</body>

</html>

```

Output:



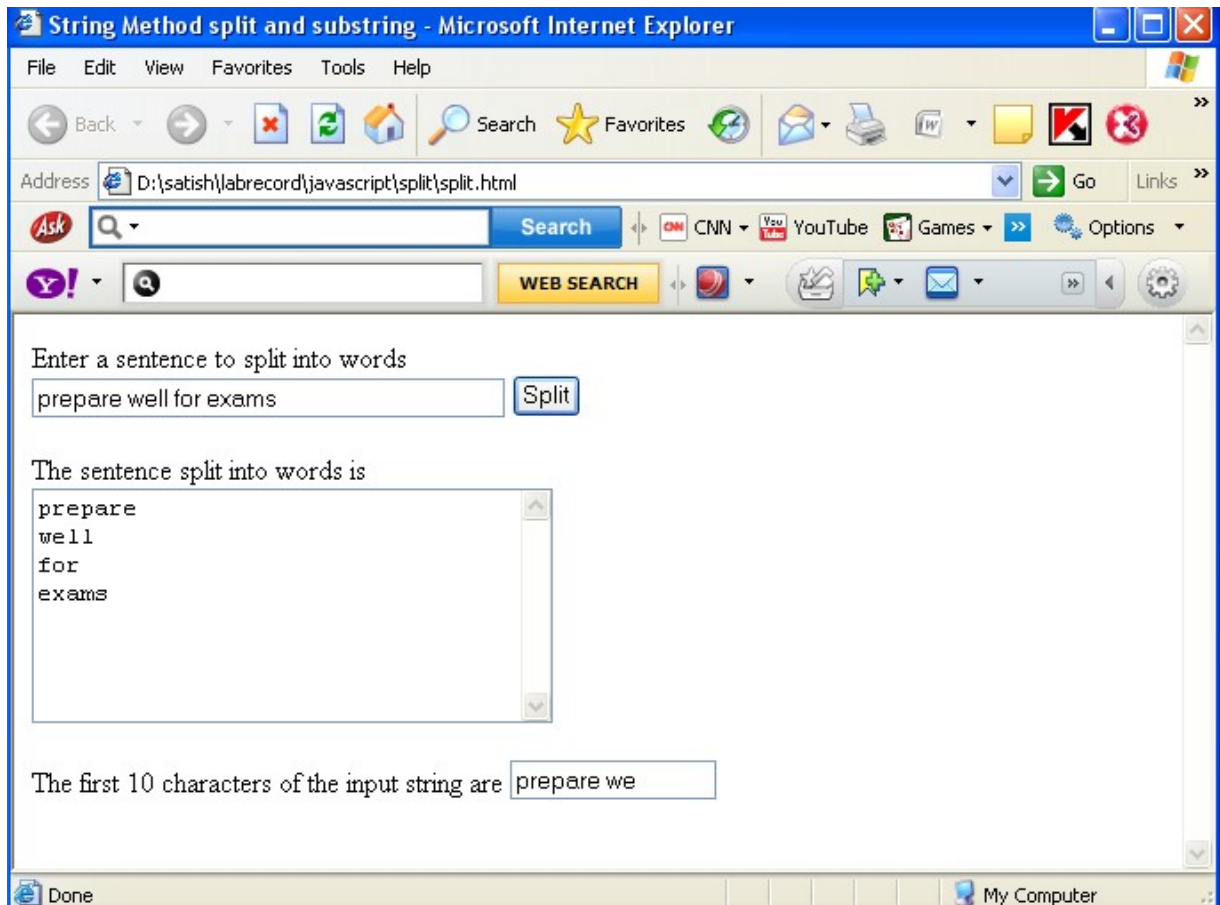
Program 16:

write a javascript program to split a given sentence into set of tokens or words by using space character as delimiter

split.html:

```
<html>
  <head>
    <title>String Method split and substring</title>
    <script type = "text/javascript">
      <!--
      function splitButtonPressed()
      {
        var strings=myForm.inputVal.value.split(" " );
        myForm.output.value = strings.join( "\n" );
        myForm.outputSubstring.value =
          myForm.inputVal.value.substring( 0, 10 );
      }
      // -->
    </script>
  </head>
  <body>
    <form name = "myForm" action = "">
      <p>Enter a sentence to split into words<br />
      <input name="inputVal" type="text" size="40" />
      <input name="splitButton" type="button" value =
        "Split" onclick = "splitButtonPressed()" />
      </p>
      <p>The sentence split into words is<br />
      <textarea name = "output" rows = "8" cols = "34">
      </textarea>
      </p>
      <p>The first 10 characters of the input string are
      <input name = "outputSubstring" type = "text"
        size = "15" /></p>
    </form>
  </body>
</html>
```

Output:



Program 17:

write a javascript program to explain about onload event

onload.html:

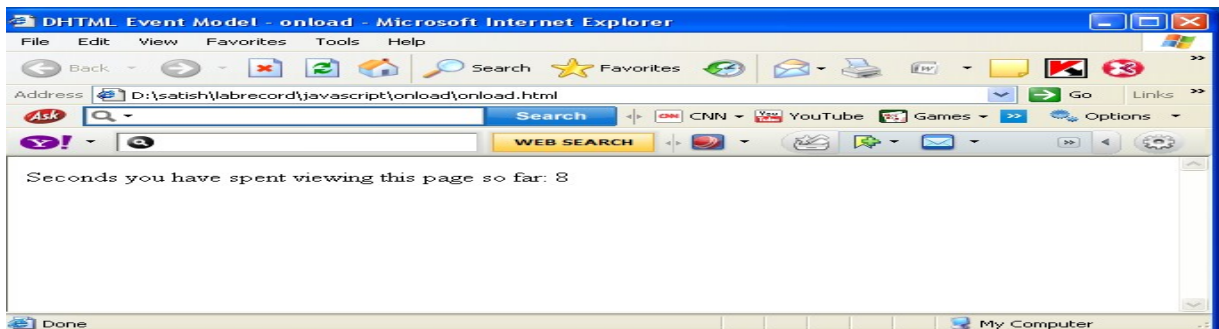
```
<html>
  <head>
    <title>DHTML Event Model - onload</title>
```

```

<script type = "text/javascript">
<!--
var seconds = 0;
function startTimer()
{
    window.setInterval( "updateTime()", 1000 );
}
function updateTime()
{
    seconds++;
    soFar.innerHTML = seconds;
}
// -->
</script>
</head>
<body onload = "startTimer()">
    <p>Seconds you have spent viewing this page so far:
    <a id = "soFar"><strong>0</strong></a></p>
</body>
</html>

```

Output:



Program 18:

write a javascript program to explain about onmousemove event

onmousemove.html:

```

<html>
<head>
    <title>DHTML Event Model - onmousemove event</title>
    <script type = "text/javascript">
    <!--
    function updateMouseCoordinates()
    {
        coordinates.innerHTML = event.srcElement.tagName +
        " (" + event.offsetX + ", " + event.offsetY + ")";
    }
    </script>
</head>
<body>
    <div id = "coordinates">
    </div>
</body>
</html>

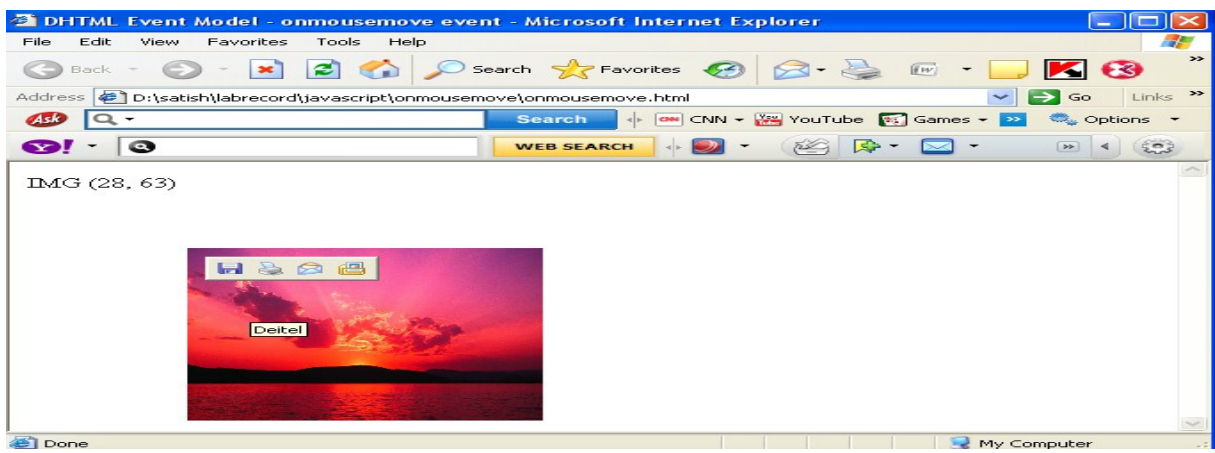
```

```

    }
    // -->
  </script>
</head>
<body style = "back-groundcolor: wheat"
  onmousemove = "updateMouseCoordinates()">
  <span id = "coordinates">(0, 0)</span><br />
  <img src = "sunset.jpg" style = "position: absolute;
  top:100;left:100" alt = "Deitel" width="200" height="200"/>
</body>
</html>

```

Output:



Program 19:

Write a javascript program to explain about onfocus,onblur ,onsubmit ,onreset events

Onfocusonblur.html:

```

<html>
  <head>
    <title>
      DHTML Event Model - onfocus onblur onsubmit onreset
    </title>
    <script type = "text/javascript">
      <!--
      var helpArray =
        [ "Enter your name in this input box.",
          "Enter your email address in this input box, " +
          "in the format user@domain.",
          "Check this box if you liked our site.",

```

```

        "In this box, enter any comments you would " +
        "like us to read.",
        "This button submits the form to the " +
        "server-side script",
        "This button clears the form",
        "This textarea provides context-sensitive " +
        "help. Click on any input field or use the TAB" +
        "key to get more information about the " +
        "input field." ];
function helpText( messageNum )
{
    myForm.helpBox.value = helpArray[ messageNum ];
}
function formSubmit()
{
    window.event.returnValue = false;
    if ( confirm ( "Are you sure you want to submit?" ) )
        window.event.returnValue = true;
}
function formReset()
{
    window.event.returnValue = false;
    if ( confirm( "Are you sure you want to reset?" ) )
        window.event.returnValue = true;
}
// -->
</script>
</head>
<body>
<form id="myForm" method="post" action=""
    onsubmit="formSubmit()" onreset="formReset()">

    Name: <input type = "text" name = "name"
        onfocus = "helpText(0)" onblur = "helpText(6)"/><br />

    Email: <input type = "text" name = "email"
        onfocus = "helpText(1)" onblur = "helpText(6)"/><br />

    Click here if you like this site
    <input type = "checkbox" name = "like" onfocus =
        "helpText(2)" onblur = "helpText(6)" /><br /><hr />
    Any comments?<br />
    <textarea name = "comments" rows = "5" cols = "45"
        onfocus = "helpText(3)" onblur = "helpText(6)">
    </textarea><br />

    <input type = "submit" value = "Submit" onfocus =

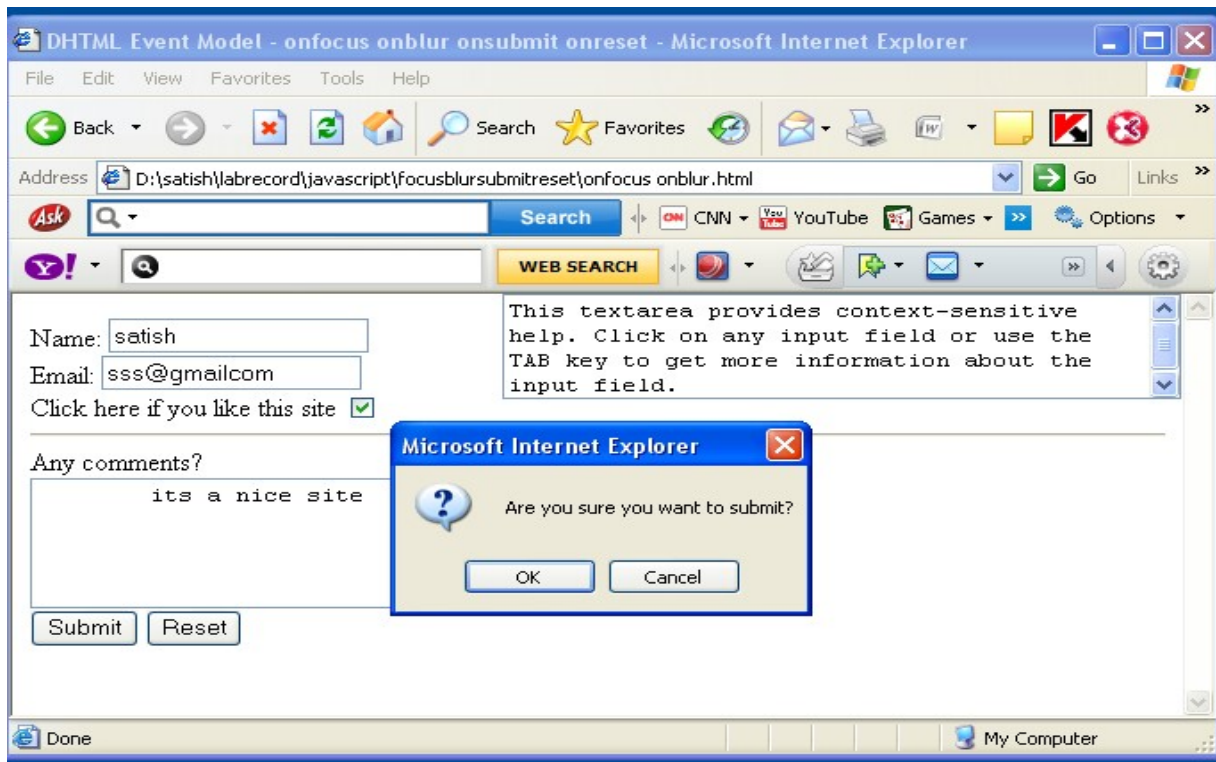
```

```

        "helpText(4)" onblur = "helpText(6)" />
<input type = "reset" value = "Reset" onfocus =
    "helpText(5)" onblur = "helpText(6)" />
<textarea name = "helpBox" style = "position: absolute;
    right: 0; top: 0" rows = "4" cols = "45">
    This textarea provides context-sensitive help. Click
    on any input field or use the Tab key to get more
    information about the input field.
</textarea>
</form>
</body>
</html>

```

Output:



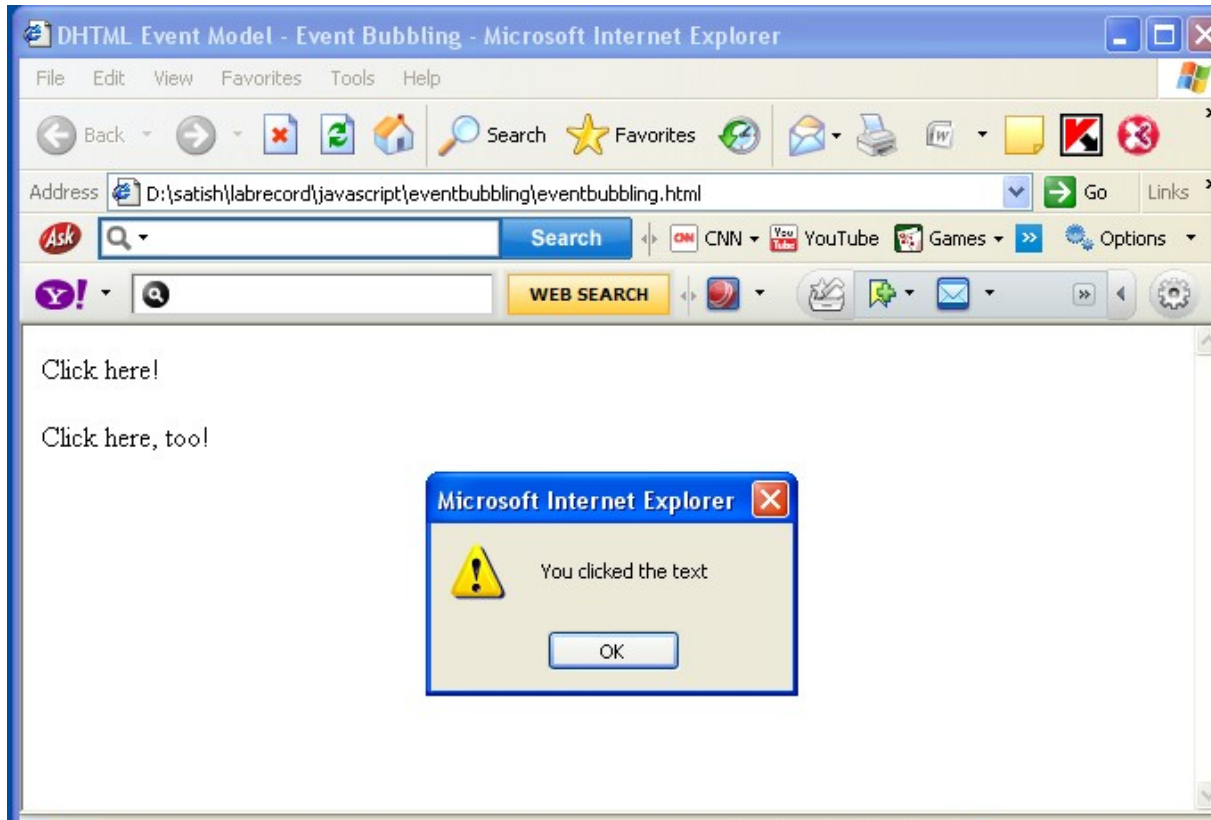
Program 20:

Write a javascript program to explain about event bubbling

Eventbubbling.html:

```
<html>
  <head>
    <title>DHTML Event Model - Event Bubbling</title>
    <script type = "text/javascript">
      <!--
        function documentClick()
        {
          alert( "You clicked in the document" );
        }
        function bodyClick()
        {
          alert( "You clicked in the body" );
        }
        function paragraphClick( value )
        {
          alert( "You clicked the text" );
          if ( value )
            event.cancelBubble = true;
        }
        document.onclick = documentClick;
      // -->
    </script>
  </head>
  <body onclick="bodyClick()">
    <p onclick = "paragraphClick( false )" >Click here!</p>
    <p onclick = "paragraphClick( true )" >Click here, too!</p>
  </body>
</html>
```

Output:



Program 21:

Write a javascript program which displays the different element/tag names of a web document using "all" collection object

all1.html:

```
<html>
  <head>
    <title>Object Model</title>
    <script type = "text/javascript">
      <!--
      var elements = "";
      function start()
      {
        for ( var loop = 0; loop < document.all.length; ++loop )
          elements += "<br />" + document.all[ loop ].tagName;
        pText.innerHTML += elements;
        alert( elements );
      }
      // -->
    </script>
  </head>
  <body onload = "start()">
    <p id = "pText">
      Elements on this Web page:
    </p>
  </body>
</html>
```

Output:



Elements on this Web page:

HTML
HEAD
TITLE
SCRIPT
BODY
P



Program 22:

Write a javascript program which displays the different element/tag names of a web document using "children" collection object

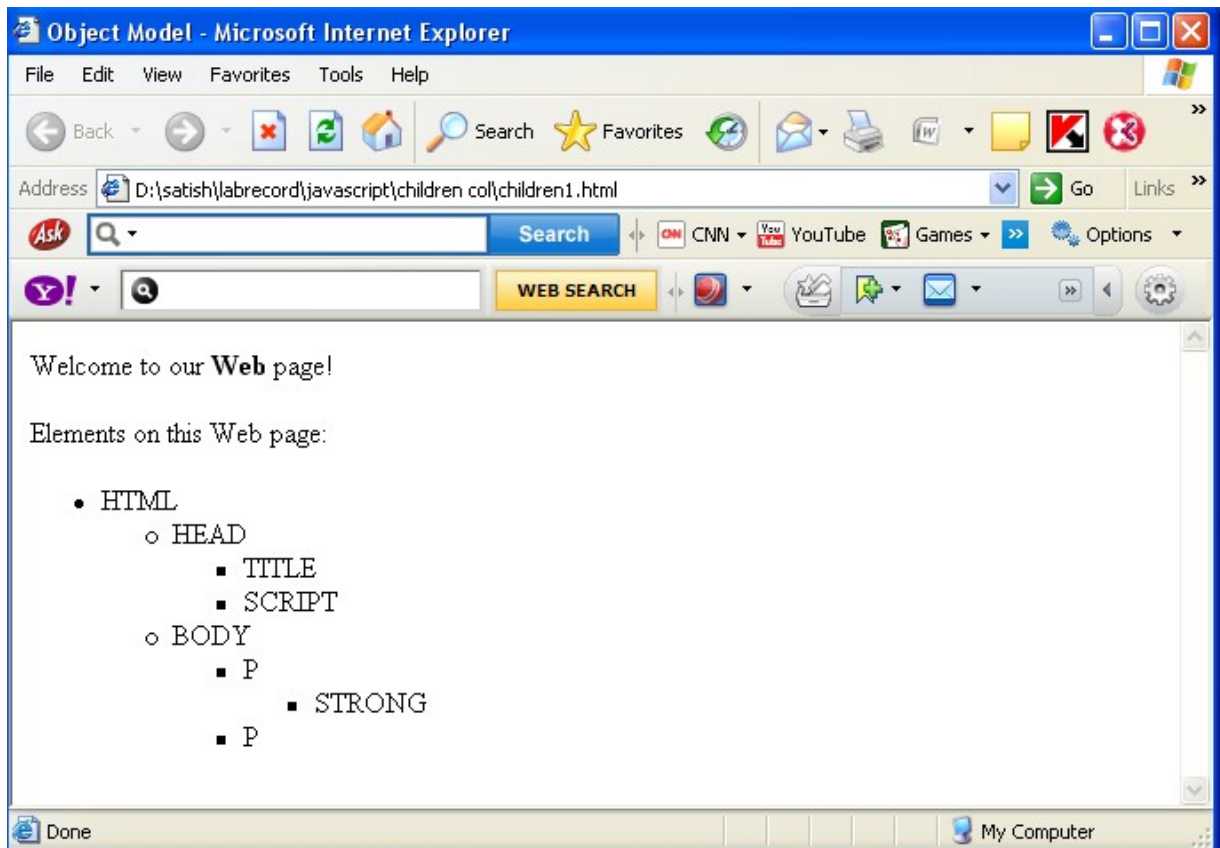
Children1.html:

```

<html>
<head>
  <title>Object Model</title>
  <script type = "text/javascript">
  <!--
    var elements = "<ul>";
    function child( object )
    {
      var loop = 0;
      elements += "<li>" + object.tagName + "<ul>";
      for ( loop = 0; loop < object.children.length; loop++ )
      {
        if ( object.children[ loop ].children.length )
          child( object.children[ loop ] );
        else
          elements += "<li>" + object.children[ loop ].tagName
            + "</li>";
      }
      elements += " </ul> ";
    }
  // -->
  </script>
</head>
<body onload = "child( document.all[ 0 ] );
      myDisplay.outerHTML += elements;">
  <p>Welcome to our <strong>Web</strong> page!</p>
  <p id = "myDisplay">
    Elements on this Web page:
  </p>
</body>
</html>

```

Output:



program 23:

Write a javascript program to explain about navigator object

navigator.html:

```
<html>
```

```

<head>
  <title>The navigator Object</title>
  <script type = "text/javascript">
    <!--
      function start()
      {
        if (navigator.appName=="Microsoft Internet Explorer")
        {
          if ( navigator.appVersion.substring( 1, 0 ) >= "4" )
            document.location = "newIEversion.html";
          else
            document.location = "oldIEversion.html";
        }
        else
          document.location = "NSversion.html";
      }
    // -->
  </script>
</head>
<body onload = "start()">
  <p>Redirecting your browser to the appropriate page,
  please wait...</p>
</body>
</html>

```

newIEversion.html:

```

<html>
  <body>
    <h3>this content is for internet explorer
  whose version is 4 or greater than 4</h3>
  </body>
</html>

```

oldIEversion.html:

```

<html>
  <body>
    <h3>this content is for internet explorer whose
  version is less than 4</h3>
  </body>
</html>

```

NSversion.html:

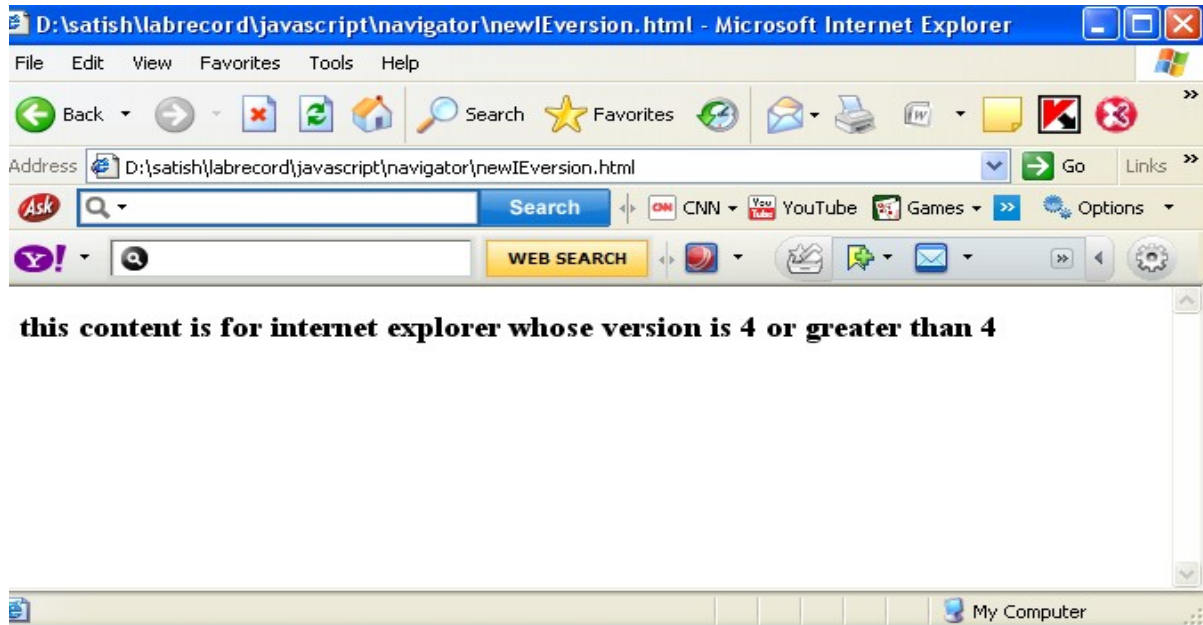
```

<html>
  <body>
    <h3>this content is for netscape navigator</h3>

```

```
</body>
</html>
```

Output:



Program 24:

Write a javascript program to explain about flip filter

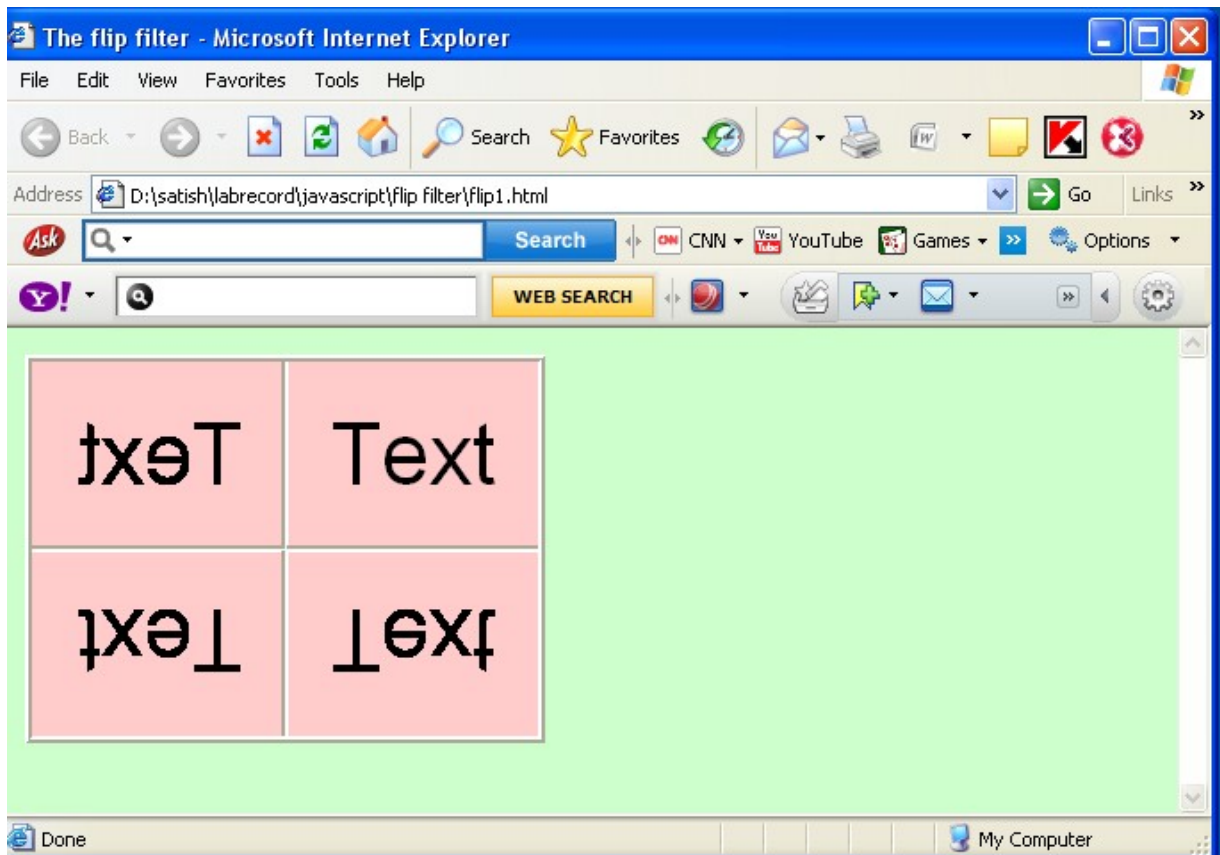
Flip1.html:

```
<html>
  <head>
    <title>The flip filter</title>
    <style type = "text/css">
      body { background-color: #CCFFCC }
      table { font-size: 3em;
              font-family: Arial, sans-serif;
              background-color: #FFCCCC;
              border-style: ridge ;
            }
      td   { border-style: groove;
              padding: 1ex
            }
    </style>
  </head>
  <table border="1">
    <tr>
      <td>
        <div style="background-color: red; width: 100px; height: 100px; margin: 10px 0; border: 1px solid black; border-style: ridge; border-radius: 50%; text-align: center; color: white; font-weight: bold; font-size: 2em; padding: 5px 20px; display: inline-block; transform: rotate(45deg);">
          Filter
        </div>
      </td>
    </tr>
  </table>
</html>
```

```
</style>
</head>

<body>
  <table>
    <tr>
      <td style = "filter: fliph">Text</td>
      <td>Text</td>
    </tr>
    <tr>
      <td style = "filter: flipv fliph">Text</td>
      <td style = "filter: flipv">Text</td>
    </tr>
  </table>
</body>
</html>
```

Output:



Program 25:

Develop static pages (using only HTML) of an online Book store. The pages should resemble :www.amazon.com. The website should consist the following pages.

- Home page
- Registration and user Login
- User profile page
- Books catalog
- Shopping cart
- Payment by credit card
- Order Conformation

Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

Main.html:

```
<frameset rows="25%, 75 %">
  <frame src="top.html" name="top">
  <frameset cols="25%,75%">
    <frame src="left.html" name="left">
    <frame src="right.html" name="right">
  </frameset>
</frameset>
```

Top.html:

```
<html>
  <body bgcolor="pink">
    <br><br>
    <marquee><h1 align="center"><b><u>ONLINE BOOK
    STORAGE</u></b></h1></marquee>
  </body>
</html>
```

Right.html:

```
<html>
  <body bgcolor="pink">
  <br><br><br><br><br>
  <h2 align="center">
    <b><p> welcome to online book storage. Press login if you
    are having id otherwise press registration.
  </p></b></h2></body>
</html>
```

Left.html :

```
<html>
  <body bgcolor="pink">
    <h3>
    <ul>
    <li><a href="login.html" target="right"><font color="black">
    LOGIN</font></a></li><br><br>
    <li><a href="reg.html" target="right"><font color="black">
    REGISTRATION</font></a></li><br><br>
    <li><a href="profile.html" target="right"><fontcolor="black">
    USER PROFILE</font></a></li><br><br>
    <li><a href="catalog.html" target="right"><fontcolor="black">
    BOOKS CATALOG</font></a></li><br><br>
    <li><a href="scart.html" target="right"><font color="black">
    SHOPPINGCART</font></a></li><br><br>
    <li><a href="payment.html" target="right"><fontcolor="black">
    PAYMENT</font></a></li><br><br>
    <li><a href="order.html" target="right"><font color="black">
    ORDER CONFIRMATION</font></a></li><br><br>
    </ul>
  </body>
</html>
```

login.html :

```
<html>
  <body bgcolor="pink"><br><br><br>
  <script language="javascript">
    function validate()
    {
      var flag=1;
      if(document.myform.id.value==" "||document.myform.pwd.
      value==" ")
      {
        flag=0;
      }
      if(flag==1)
      {
        alert("VALID INPUT");
      }
    }
  </script>

```



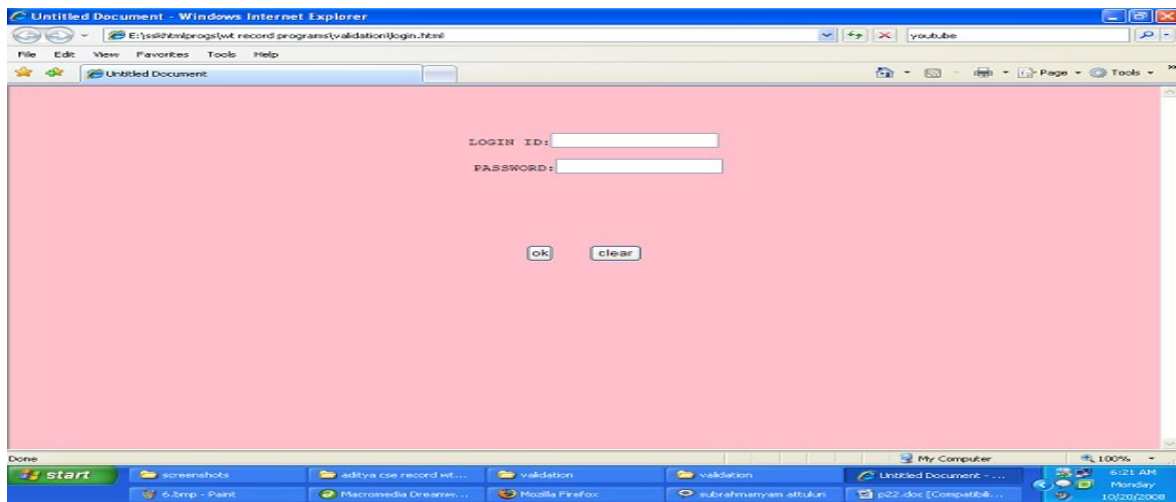
```
<body bgcolor="pink"><br><br><br>
<script language="javascript">
function validate()
{

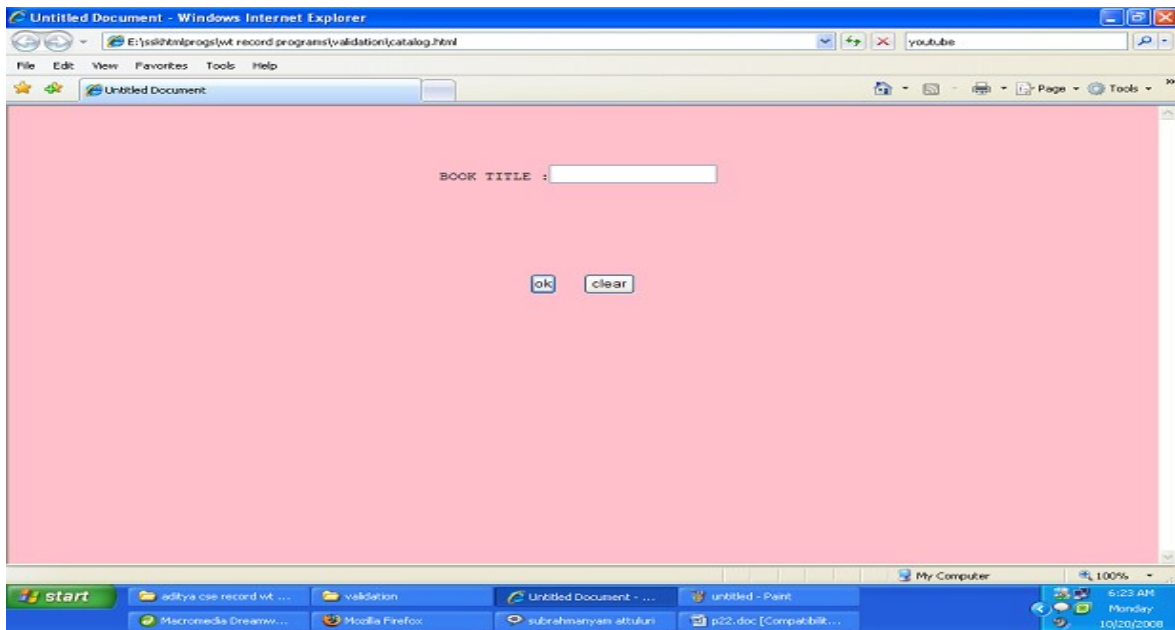
var flag=1;
if(document.myform.id.value=="||
document.myform.pwd.value=="||
document.myform.amount.value=="||
document.myform.num.value=="")
{
  flag=0;
}
var str=document.myform.amount.value;
var x;
for(var i=0;i<str.length;i++)
{
x=str.substr(i,1);
if(!(x<=9))
{
flag=0;
break;
}
}
str=document.myform.num.value;
for(var i=0;i<str.lenght;i++)
{
x=str.substr(i,1);
if(!(x<=9))
{
flag=0;
break;
}
}
if(flag==1)
{
alert("VALID INPUT");
}
else
{
```

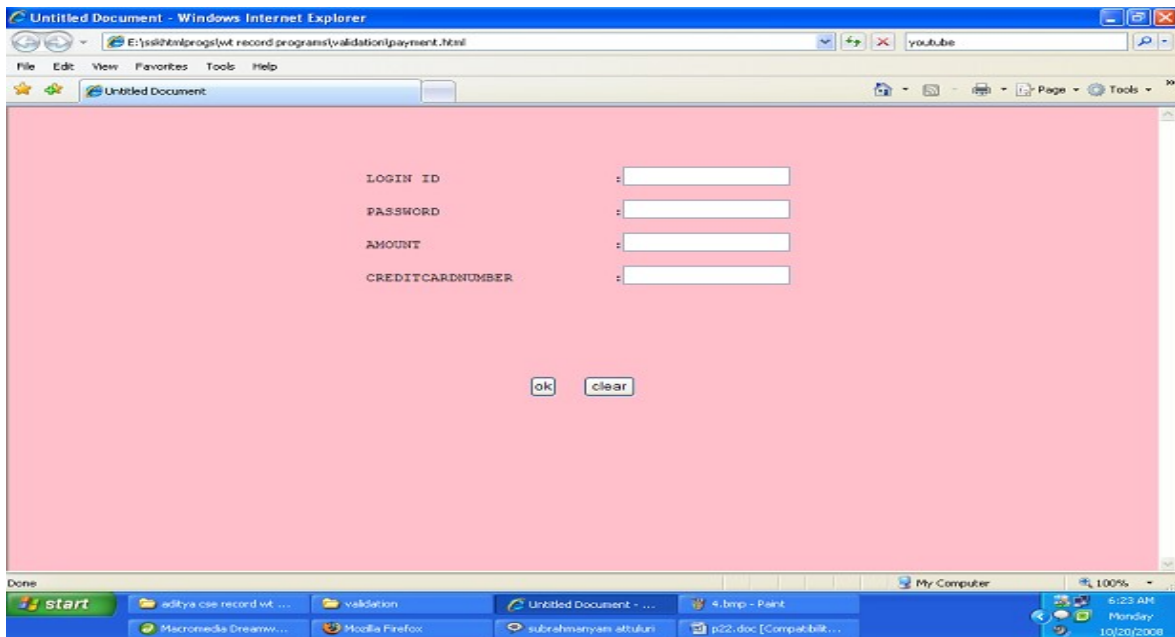
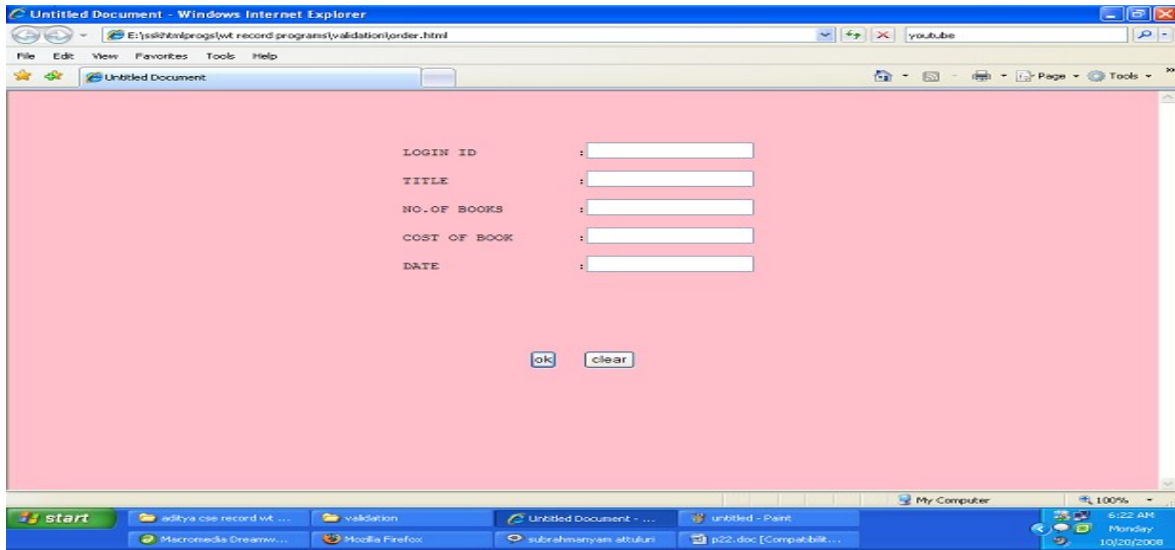
ONLINE BOOK STORAGE

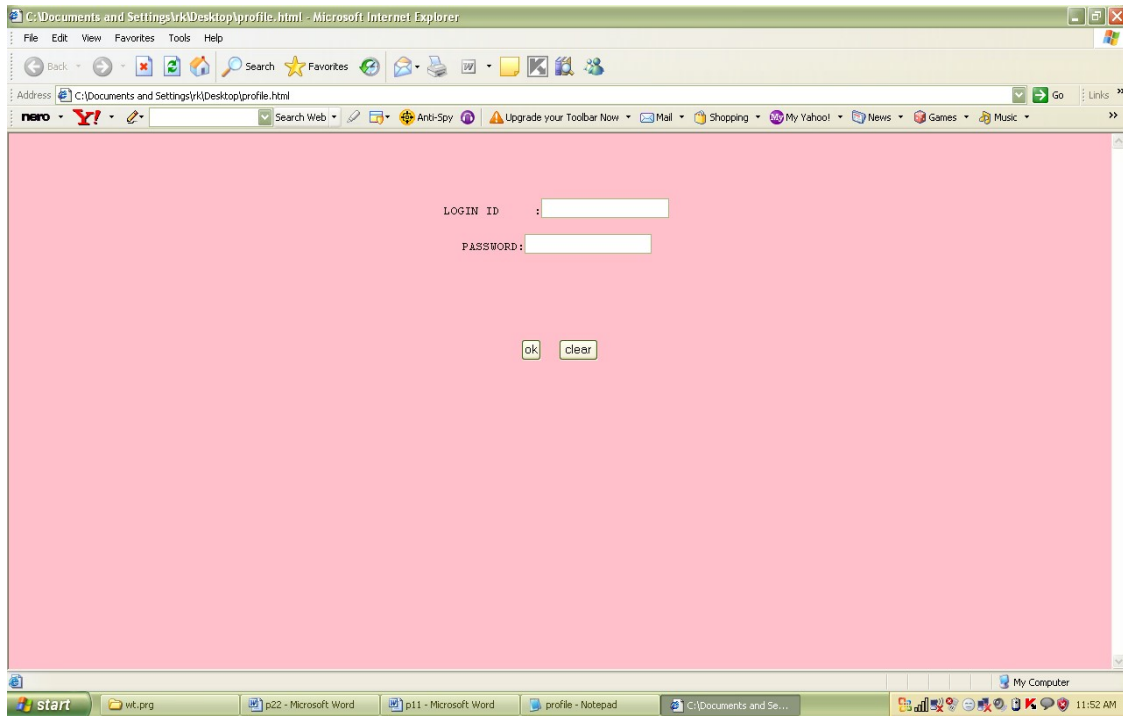
- [LOGIN](#)
- [REGISTRATION](#)
- [USER PROFILE](#)
- [BOOKS CATALOG](#)
- [SHOPPINGCART](#)
- [PAYMENT](#)
- [ORDER CONFIRMATION](#)

welcome to online book storage. Press login if you are having id otherwise press registration.









Program 26:

Write a Document type definition (DTD) document for representing the structure of employee information

Emps1.dtd:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT emps (emp+,dept)>
```

```

<!ELEMENT emp (name,salary,desig)>
<!ELEMENT name (firstname,lastname)>
<!ELEMENT firstname (#PCDATA)>
<!ELEMENT lastname (#PCDATA)>
<!ELEMENT salary (#PCDATA)>
<!ELEMENT desig (#PCDATA)>
<!ELEMENT dept (deptname,desc)>
<!ELEMENT deptname (#PCDATA)>
<!ELEMENT desc (#PCDATA)>
<!ATTLIST emp empno ID #REQUIRED>
<!ATTLIST emp mgrid IDREF #IMPLIED>
<!ATTLIST emp working (yes | no) 'yes'>
<!ATTLIST dept deptno ID #REQUIRED>

```

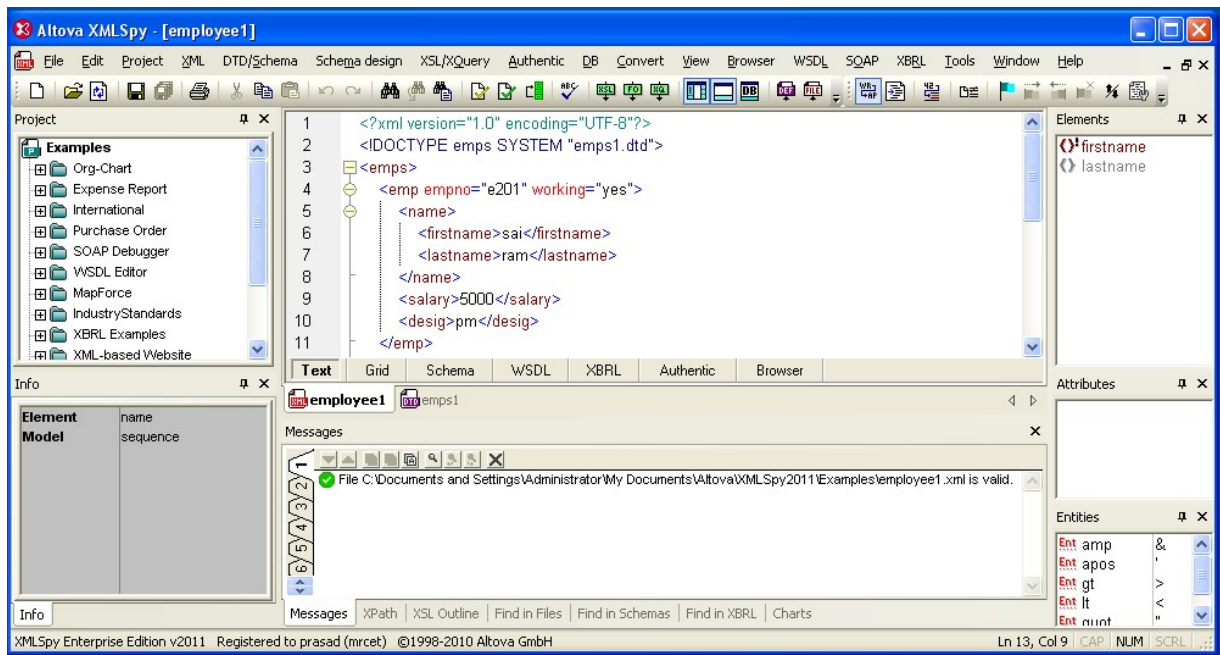
Employee1.xml :

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE emps SYSTEM "emps1.dtd">
<emps>
  <emp empno="e201" working="yes">
    <name>
      <firstname>sai</firstname>
      <lastname>ram</lastname>
    </name>
    <salary>5000</salary>
    <desig>pm</desig>
  </emp>
  <emp empno="e202" mgrid="e201" working="yes">
    <name>
      <firstname>satish</firstname>
      <lastname>kumar</lastname>
    </name>
    <salary>2000</salary>
    <desig>programmer</desig>
  </emp>
  <dept deptno="d111">
    <deptname>developing team</deptname>
    <desc>develops programs</desc>
  </dept>
</emps>

```

Output:



Program 27:

Write a xml schema document (xsd) document for representing the structure of employee information

emps2.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs=http://www.w3.org/2001/XMLSchema
elementFormDefault="qualified"
attributeFormDefault="unqualified">
<xs:element name="emps" type="empstype">
  <xs:annotation>
    <xs:documentation>
      this element stores about various employees
    </xs:documentation>
  </xs:annotation>
</xs:element>
```

```

<xs:simpleType name="salarytype">
  <xs:restriction base="xs:double">
    <xs:minInclusive value="5000"></xs:minInclusive>
    <xs:maxInclusive value="35000"></xs:maxInclusive>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="desigtype">
  <xs:restriction base="xs:string">
    <xs:enumeration value="projectmanager"></xs:enumeration>
    <xs:enumeration value="teamleader"></xs:enumeration>
    <xs:enumeration value="developer"></xs:enumeration>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="deptnametype">
  <xs:restriction base="xs:string">
    <xs:enumeration value="testing"></xs:enumeration>
    <xs:enumeration value="design"></xs:enumeration>
    <xs:enumeration value="programming"></xs:enumeration>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="workingtype">
  <xs:restriction base="xs:string">
    <xs:enumeration value="yes"></xs:enumeration>
    <xs:enumeration value="no"></xs:enumeration>
  </xs:restriction>
</xs:simpleType>
<xs:attributeGroup name="empattributes">
  <xs:attribute name="empno" type="xs:ID" use="required">
</xs:attribute>
  <xs:attribute name="mgrid" type="xs:IDREF" use="optional">
</xs:attribute>
  <xs:attribute name="working" default="yes" use="optional"
type="workingtype">
</xs:attribute>
</xs:attributeGroup>
<xs:complexType name="empnametype">
  <xs:sequence>
    <xs:element name="firstname" type="xs:string">
</xs:element>
    <xs:element name="lastname" type="xs:string">
</xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="emptytype">
  <xs:sequence>
    <xs:element name="name" type="empnametype">

```

```

        </xs:element>
        <xs:element name="salary" type="salarytype">
        </xs:element>
        <xs:element name="desig" type="desigtype">
        </xs:element>
    </xs:sequence>
    <xs:attributeGroup ref="empattributes"></xs:attributeGroup>
</xs:complexType>
<xs:complexType name="depttype">
    <xs:sequence>
        <xs:element name="name" type="deptnametype">
        </xs:element>
        <xs:element name="desc" type="xs:string"></xs:element>
    </xs:sequence>
    <xs:attribute name="deptno" use="required"></xs:attribute>
</xs:complexType>
<xs:complexType name="empstype">
    <xs:sequence>
        <xs:element name="emp" type="emptytype"
            minOccurs="1" maxOccurs="20"></xs:element>
        <xs:element name="dept" type="depttype"
            minOccurs="1"></xs:element>
    </xs:sequence>
</xs:complexType>
</xs:schema>

```

Employee2.xml :

```

<?xml version="1.0"?>
<emps xmlns="http://www.w3schools.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="emps2.xsd" >
    <emp empno="e201" working="yes">
        <name>
            <firstname> raj </firstname>
            <lastname> kumar </lastname>
        </name>
        <salary>10000</salary>
        <desig>projectmanager</desig>
    </emp>
    <emp empno="e102" mgrid="e201" working="yes">
        <name>
            <firstname> durga </firstname>
            <lastname> rao </lastname>
        </name>

```



```
<salary>16500</salary>
<desig> teamlead </desig>
</emp>
<dept deptno="d101">
  <name>testing</name>
  <desc> tests the projects </desc>
</dept>
</emps>
```

Output:

The screenshot displays the Altova XMLSpy interface. The main window shows an XML document with the following content:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <emps xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:noNamespaceSchemaLocation="emps2.xsd">
4   <emp empno="e201" working="yes">
5     <name>
6       <firstname>sai</firstname>
7       <lastname>ram</lastname>
8     </name>
9     <salary>25000</salary>
10    <desig>projectmanager</desig>
11  </emp>
```

The interface includes a Project pane on the left with a tree view of examples, an Info pane at the bottom left, and a Messages pane at the bottom center. The Messages pane shows a green checkmark and the text: "File C:\Documents and Settings\Administrator\My Documents\Altova\XMLSpy2011\Examples\employee2.xml is valid." The status bar at the bottom indicates "Ln 1, Col 1 | CAP NUM SCRL ...".

Program 28:

Write an xml stylesheet language document (xsl) for converting Employee markup language document to html document

Empxsl.xsl:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <html>
  <body>
  <h2>Employee details</h2>
  <table border="1">
    <tr bgcolor="#9acd32">
      <th>empno</th>
      <th>firstname</th>
      <th>lastname</th>
      <th>salary</th>
      <th>designation</th>
    </tr>
    <xsl:for-each select="emps/emp">
      <tr>
        <td><xsl:value-of select="@empno"/></td>
        <td><xsl:value-of select="./name/first-name"/></td>
        <td><xsl:value-of select="./name/last-name"/></td>
        <td><xsl:value-of select="sal"/></td>
        <td><xsl:value-of select="desig"/></td>
      </tr>
    </xsl:for-each>
  </table>
  </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

Employee.xml:

```
<?xml version="1.0"?>
<?xml-stylesheet href="empxsl.xsl" type="text/xsl" ?>
```

```

<emps>
  <emp empno="e201" working="yes">
    <name>
      <first-name> raj </first-name>
      <last-name> kumar </last-name>
    </name>
    <sal>4400.0</sal>
    <desig> engineer </desig>
  </emp>
  <emp empno="e102" managerid="e201" working="yes">
    <name>
      <first-name> durga </first-name>
      <last-name> rao </last-name>
    </name>
    <sal>16500.0</sal>
    <desig> manager </desig>
  </emp>
<dept deptno="d101">
  <dept-name> SCIENCE </dept-name>
  <desc> Developes science projects </desc>
</dept>
</emps>

```

Output:



Program 29:

Write an web application to explain the servlet lifecycle

/servletlifecycle/hello.html:

```
<html>
```

```

    <body>
        <center><u><h3>Program to demonstrate lifecycle of
        servlet</h3></u></center>
        <form method="get" action="sayhello">
            <input type="submit" value="HELLO"/>
        </form>
    </body>
</html>

```

/servletlifecycle/WEB-INF/web.xml:

```

<web-app>
    <servlet>
        <servlet-name>HS</servlet-name>
        <servlet-class>com.nit.hello.HelloServlet
        </servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>HS</servlet-name>
        <url-pattern>/sayhello</url-pattern>
    </servlet-mapping>
</web-app>

```

/servletlifecycle/WEB-INF/classes/HelloServlet.java:

```

package com.nit.hello;
import javax.servlet.*;
import java.io.*;
public class HelloServlet implements Servlet
{
    ServletConfig sc;
    public void init(ServletConfig sc) throws ServletException
    {
        System.out.println("in init method ");
        this.sc=sc;
    }
    public void service(ServletRequest req, ServletResponse res)
        throws ServletException, IOException
    {
        System.out.println("in service method");
        PrintWriter out=res.getWriter();
        out.println("<center><h3><u>");
        out.println("Program to demonstrate lifecycle of
        servlet </u></h3></center>");
        out.println("<b> Hello from first servlet </b> ");
    }
}

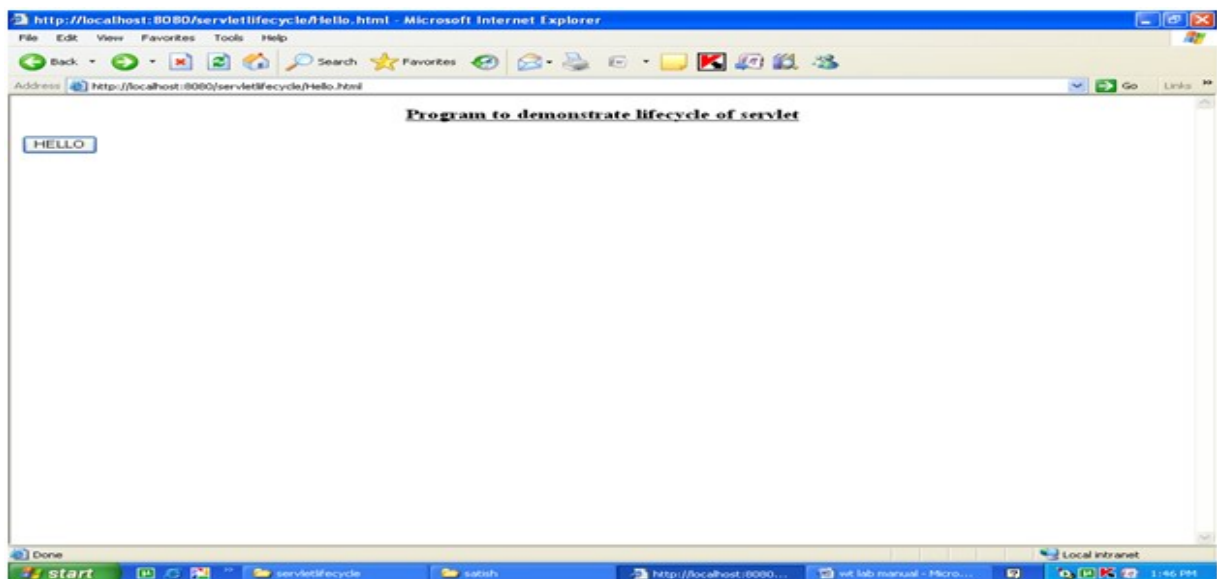
```

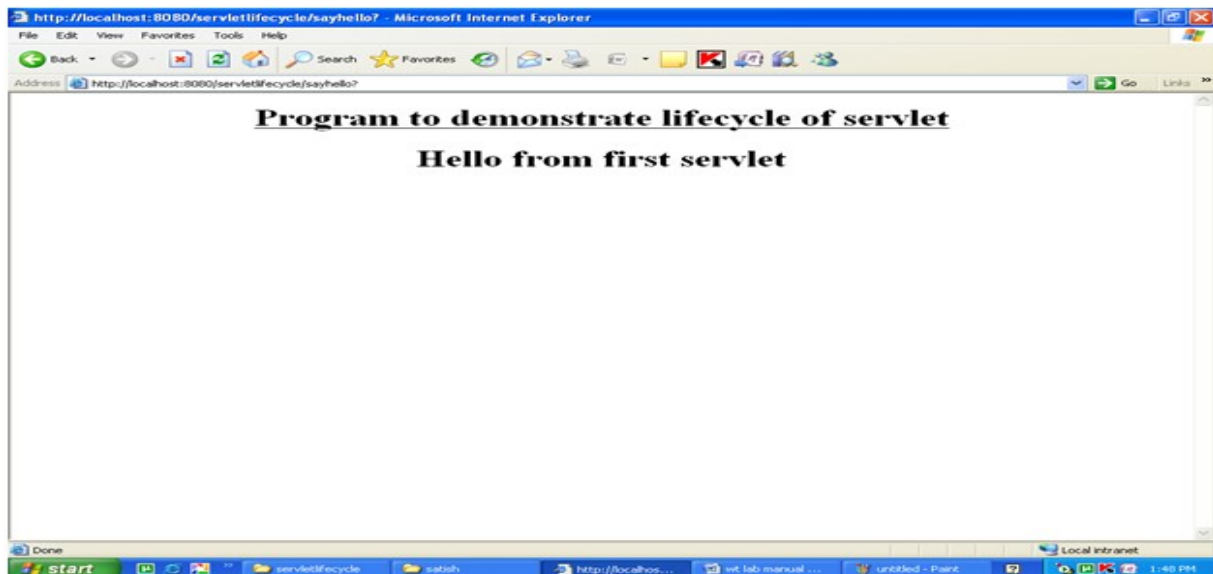
```
public void destroy()
{
    System.out.println("in destroy method");
}
public ServletConfig getServletConfig()
{
    return sc;
}
public String getServletInfo()
{
    return "helloservlet";
}
}
```

Compilation:

```
Javac -d . HelloServlet.java
```

Output:





Program 30:

Write an web application to explain about servlet initial parameters and context or application level parameters

/initialparameters/first.html:

```
<html>
  <body>
    <center>
      <h3>Program explaining initial parameters and
      context level parameters </h3><br/><br/>
      <a href="./wish">click here to view initial and
      context level parameters </a>
    </center>
  </body>
</html>
```

/initialparameters/WEB-INF/web.xml:

```
<web-app >
  <context-param>
    <param-name>emailid</param-name>
    <param-value>mrec@gmail.com</param-value>
  </context-param>
```

```

<context-param>
  <param-name>drivername</param-name>
  <param-value>sun.jdbc.odbc.JdbcOdbcDriver</param-value>
</context-param>
<servlet>
  <servlet-name>firstexample</servlet-name>
  <servlet-class>MyServlet</servlet-class>
  <init-param>
    <param-name>user1</param-name>
    <param-value>cseA</param-value>
  </init-param>
  <init-param>
    <param-name>user2</param-name>
    <param-value>cseB</param-value>

  </init-param>
</servlet>
<servlet-mapping>
  <servlet-name>firstexample</servlet-name>
  <url-pattern>/wish</url-pattern>
</servlet-mapping>
</web-app>

```

/initialparameters/WEB-INF/MyServlet.java:

```

import javax.servlet.*;
import java.io.*;
import java.util.*;
public class MyServlet extends GenericServlet
{
  public void init()throws ServletException
  {
    System.out.println("in init() method");
  }
  public void service(ServletRequest req,ServletResponse res)
    throws ServletException,IOException
  {
    System.out.println("in
Service(servletrequest,servletresponse) method");
    res.setContentType("text/html");
    PrintWriter out=res.getWriter();
    ServletConfig scg=getServletConfig();
    ServletContext sct=scg.getServletContext();
    out.println("<html><body>");
    out.println("<h3>The Context level parameters configured
for application are:</h3><br/>");
  }
}

```

```

out.println("<table width=\"200\"
border=\"3\"><tr><th>ParamName</th><th>ParamValue</th></tr>
");
Enumeration en=sct.getInitParameterNames();
while(en.hasMoreElements())
{
    String name=(String)en.nextElement();
    String value=sct.getInitParameter(name);

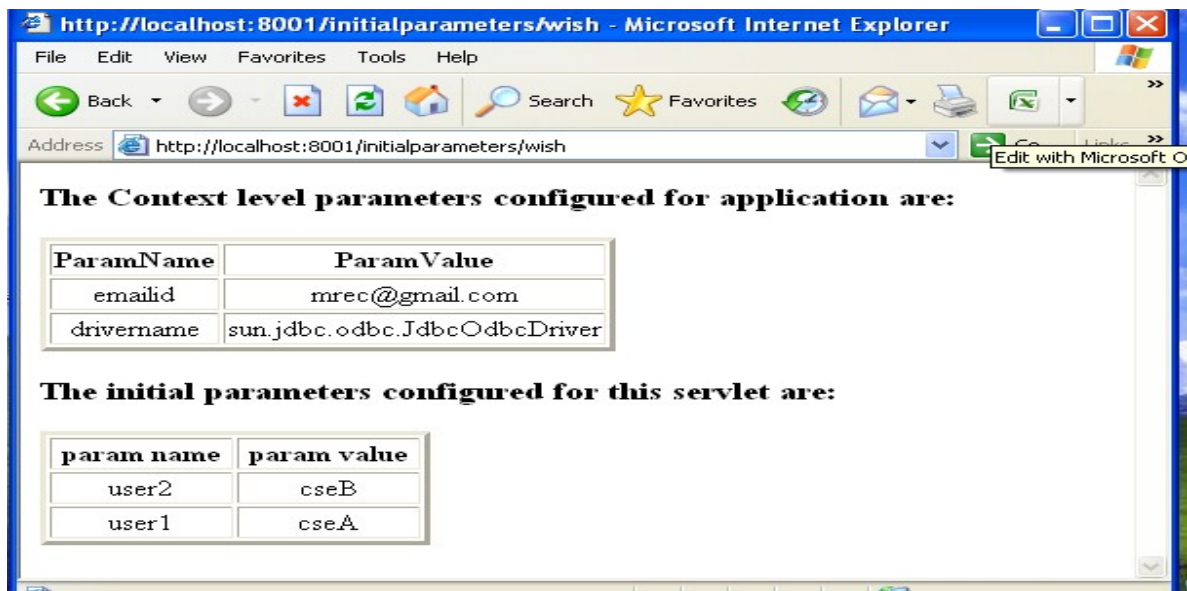
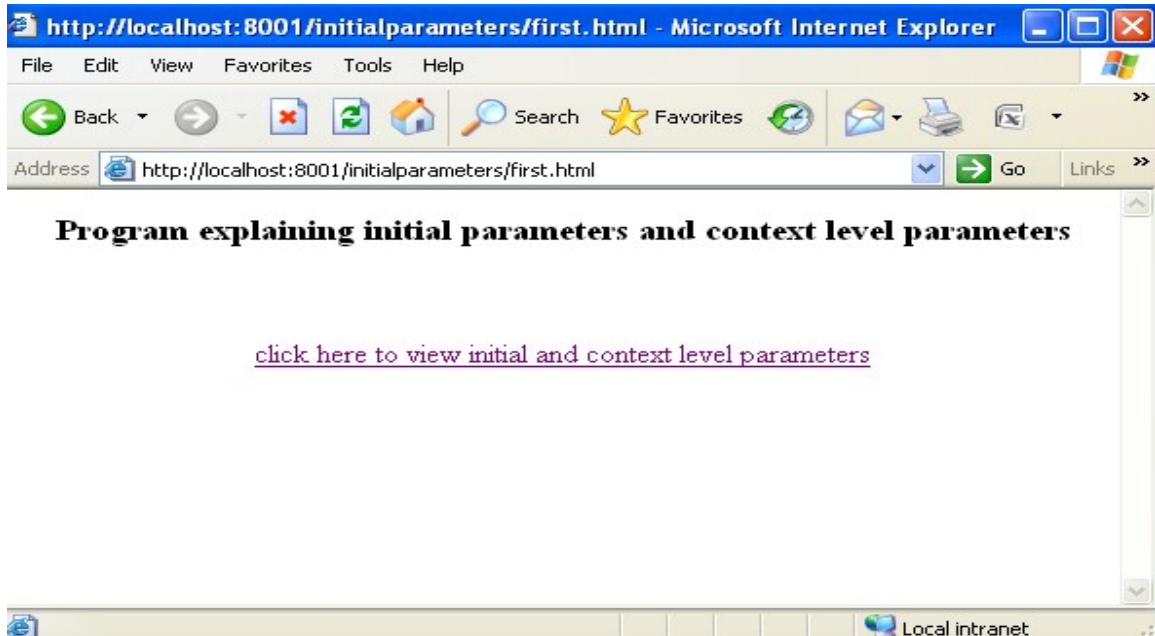
    out.println("<tr><td align=\"center\">"+ name
    + "</td><td align=\"center\">" + value + "</td></tr>");
}
out.println("</table>");
out.println("<h3>The initial parameters configured for
this servlet are:</h3><br/>");
out.println("<table width=\"200\"
border=\"3\"><tr><th>param name</th><th>param
value</th></tr>");
Enumeration en1=scg.getInitParameterNames();
while(en1.hasMoreElements())
{
    String name=(String)en1.nextElement();
    String value=scg.getInitParameter(name);
    out.println("<tr><td align=\"center\">"+ name
    + "</td><td align=\"center\">" + value + "</td></tr>");
}
out.println("</table>");
out.println("</body></html>");
}
}

```

Compilation:

Javac MyServlet.java

Output:



Program 31:

Write an web application for retrieving request parameters

/requestparameters/form.html:

```
<html>
  <body>
    <center><u><h3>Program to get the request data from
the client</h3></u></center>
    <form method="get" action="s1">
      First-Name:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input type="text"
name="firstname"/><br/><br/>
      Last-Name:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input
type="text" name="lastname"/><br/><br/>
      Father-Name:&nbsp;&nbsp;&nbsp;<input type="text"
name="fathername"/><br/><br/>
      <input type="submit" value="submit"/>
    </form>
  </body>
</html>
```

/requestparameters/WEB-INF/web.xml:

```
<web-app>
  <servlet>
    <servlet-name>HS</servlet-name>
    <servlet-class>com.nit.reqdata.RequestData
    </servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>HS</servlet-name>
    <url-pattern>/s1</url-pattern>
  </servlet-mapping>
</web-app>
```

/requestparameters/WEB-INF/classes/RequestData.java:

```
package com.nit.reqdata;
import java.io.*;
import javax.servlet.*;
public class RequestData extends GenericServlet
{
```

```

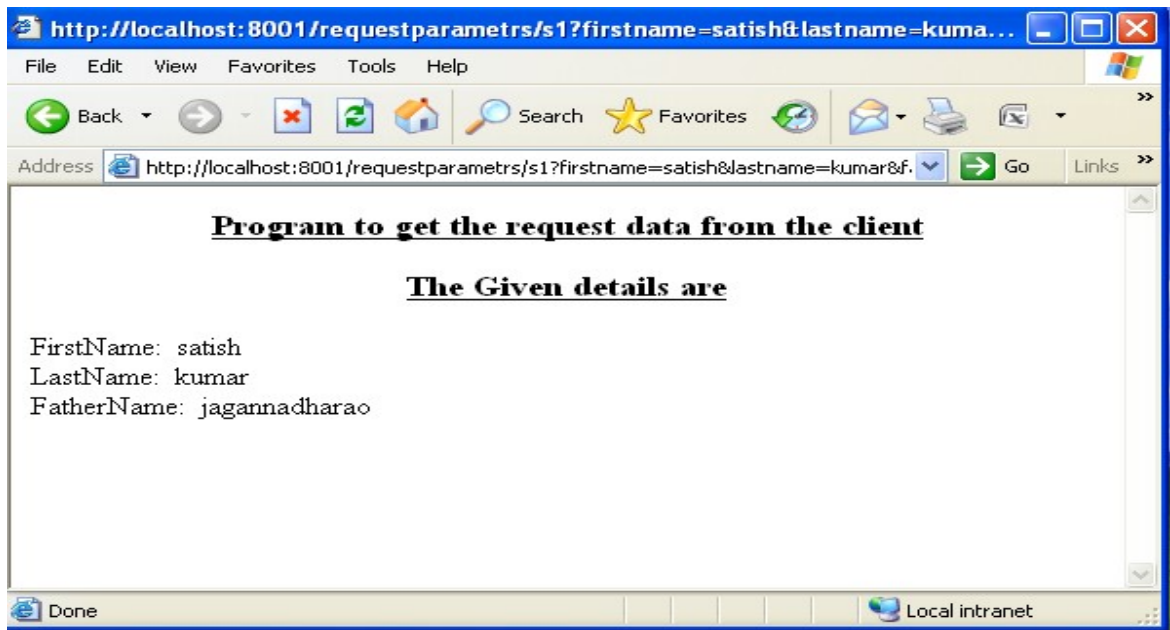
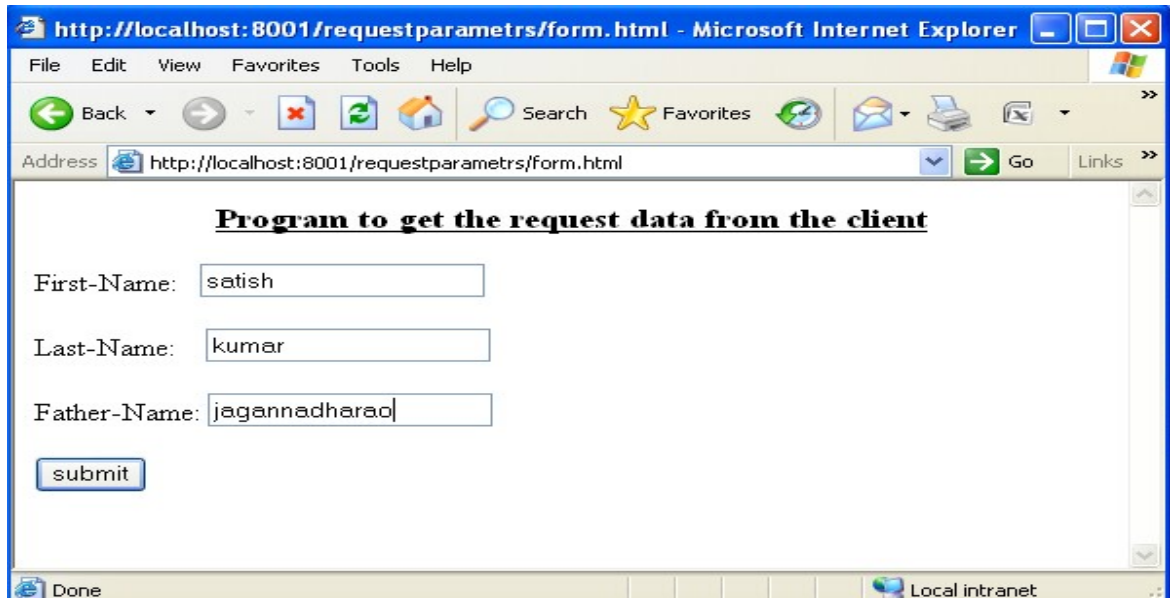
public void service(ServletRequest req,ServletResponse
                  res)throws ServletException,IOException
{
    String fn=req.getParameter("firstname");
    String ln=req.getParameter("lastname");
    String fathername=req.getParameter("fathername");
    res.setContentType("text/html");
    PrintWriter pw=res.getWriter();
    pw.println("<html><body>");
    pw.println("<center><u><h3>Program to get the request
              data from the client</h3></u></center>");
    pw.println("<center><u><h3>The Given details
              are</h3></u></center>");
    pw.println("FirstName:&nbsp;&nbsp;&nbsp;"+fn+"<br/>");
    pw.println("LastName:&nbsp;&nbsp;&nbsp;"+ln+"<br/>");
    pw.println("FatherName:&nbsp;&nbsp;&nbsp;"+fathername+"<br/>
              ");
    pw.println("</html></body>");
}
}

```

Compilation:

```
Javac -d . RequestData.java
```

Output:



Program 32:

Write an web application to create and retrieve cookies

/cookiesex/firstform.html:

```
<html>
```



```

</servlet>
<servlet>
    <servlet-name>Servlet2</servlet-name>
    <servlet-class>SecondServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>Servlet1</servlet-name>
    <url-pattern>/ser1</url-pattern>
</servlet-mapping>

<servlet-mapping>
    <servlet-name>Servlet2</servlet-name>
    <url-pattern>/ser2</url-pattern>
</servlet-mapping>
</web-app>

```

/cookiesex/WEB-INF/classes/FirstServlet.java:

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class FirstServlet extends HttpServlet
{
    public void doGet(HttpServletRequest req, HttpServletResponse
                        res) throws ServletException, IOException
    {
        res.setContentType("text/html");
        PrintWriter pw=res.getWriter();
        String cookieonename=req.getParameter("cookieonename");
        String cookietwoname=req.getParameter("cookietwoname");
        String cookieonevalue=req.getParameter("cookieonevalue");
        String cookietwovalue=req.getParameter("cookietwovalue");
        Cookie c1=new Cookie(cookieonename,cookieonevalue);
        Cookie c2=new Cookie(cookietwoname,cookietwovalue);
        res.addCookie(c1);
        res.addCookie(c2);
        RequestDispatcher rd=
            req.getRequestDispatcher("/secondform.html");
        rd.forward(req,res);
    }
}

```

/cookiesex/WEB-INF/classes/SecondServlet.java:

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

```

```

public class SecondServlet extends HttpServlet
{
    public void doGet(HttpServletRequest req,HttpServletResponse
                        res)throws ServletException,IOException
    {
        res.setContentType("text/html");
        PrintWriter pw=res.getWriter();
        pw.println("<html><body>");
        pw.println("<center><u><h3>Program to demonstrate Use of
                    Cookies</h3></u></center>");
        pw.println("<center><u><h3>The Added Cookies
                    are</h3></u></center>");
        Cookie c[]=req.getCookies();
        for(int i=0;i<c.length;i++)
        {
            pw.println("CookieName:&nbsp;&nbsp; "+c[i].getName()+
                       "&nbsp;&nbsp;&nbsp;");
            pw.println("cookievalue:&nbsp;&nbsp;&nbsp;"+c[i].getValue());
            pw.println("<br/><br/>");
        }
        pw.println("<br/><br/><br/><br/>");
        pw.println("<center><h3> <u><a
                    href=\"./firstform.html\">Click here to add
                    more cookies</a> </u></h3></center>");
        pw.println("</html></body>");
    }
}

```

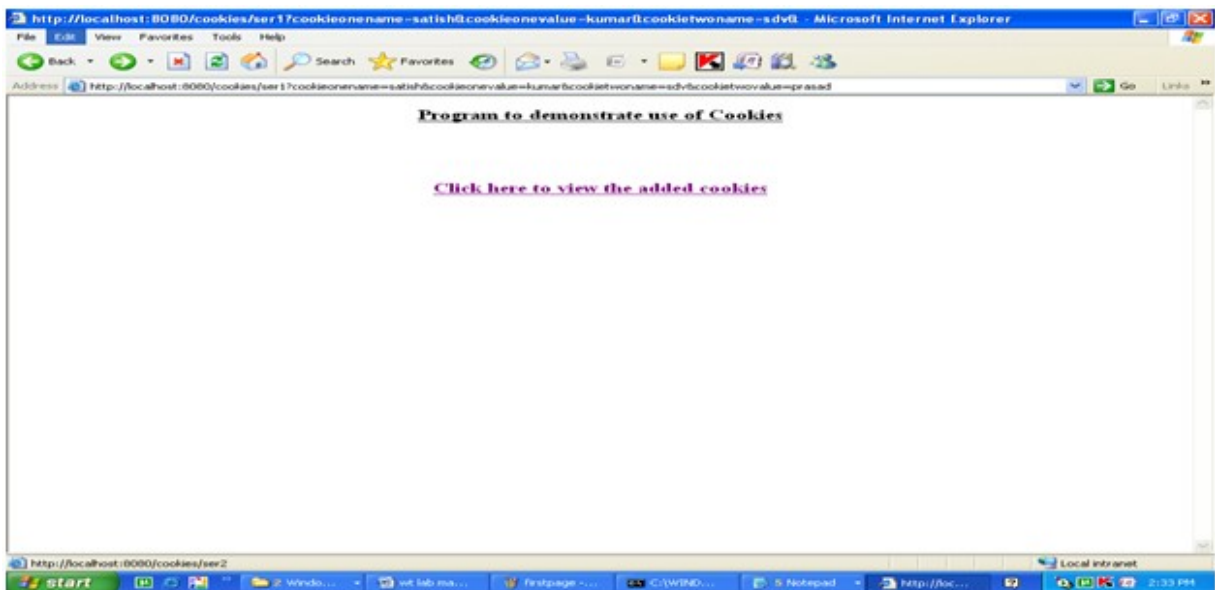
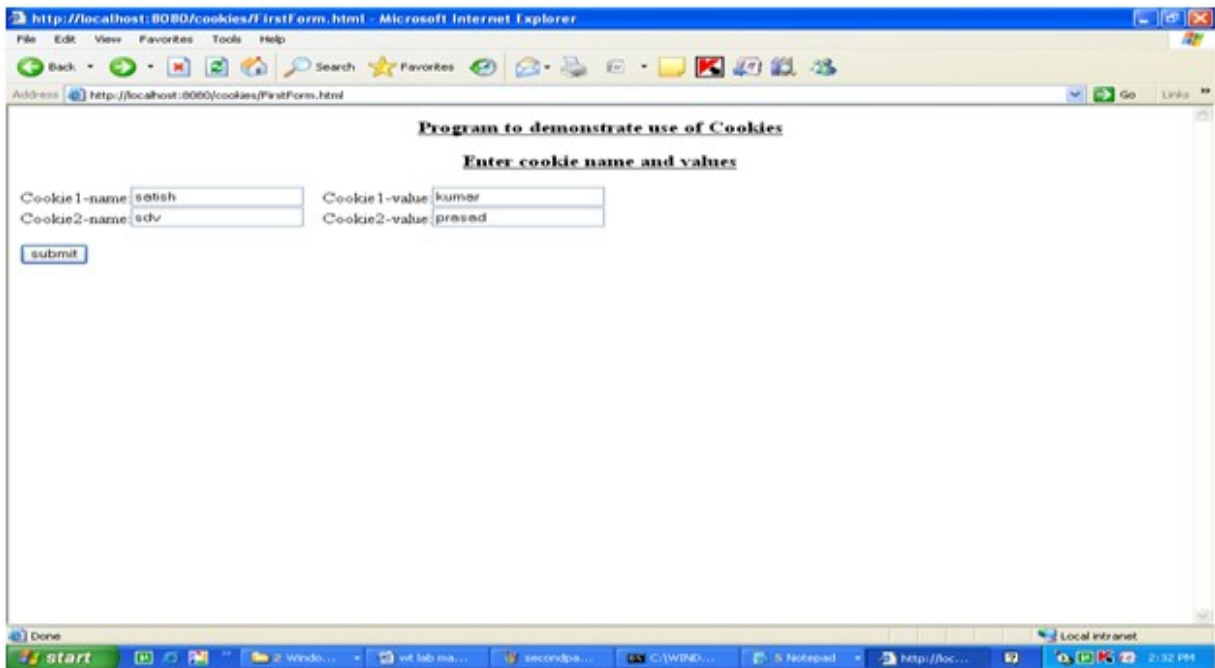
Compilation:

```

javac FirstServlet.java
javac SecondServlet.java

```

output:




```

public void doGet(HttpServletRequest req, HttpServletResponse
                    res) throws ServletException, IOException
{
    res.setContentType("text/html");
    PrintWriter pw=res.getWriter();
    HttpSession hs=req.getSession();
    String firstname=req.getParameter("firstname");
    String lastname=req.getParameter("lastname");
    String fathename=req.getParameter("fathename");
    hs.setAttribute("firstname",firstname);
    hs.setAttribute("lastname",lastname);

    hs.setAttribute("fathename",fathename);
    RequestDispatcher rd=
        req.getRequestDispatcher("/secondform.html");
    rd.forward(req,res);
}
}

```

/session/WEB-INF/classes/SecondServlet.java:

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class SecondServlet extends HttpServlet
{
    public void doGet(HttpServletRequest req,HttpServletResponse
                    res) throws ServletException, IOException
    {
        res.setContentType("text/html");
        PrintWriter pw=res.getWriter();
        HttpSession hs=req.getSession();
        String dob=req.getParameter("dob");
        String address=req.getParameter("address");
        String firstname=(String)hs.getAttribute("firstname");
        String lastname=(String)hs.getAttribute("lastname");
        String fathename=(String)hs.getAttribute("fathename");
        pw.println("<html><body>");
        pw.println("<center><u><h3>Program to demonstrate Session
                    Tracking</h3></u></center>");
        pw.println("<center><h3>The Details Given by you
                    are</h3></center>");
        pw.println("First Name is:<h3>"+firstname+"</h3><br/>");
        pw.println("Last Name is:<h3>"+lastname+"</h3><br/>");
        pw.println("Father name is:<h3>"+fathename+"</h3><br/>");
        pw.println("Date of Birth:<h3>"+dob+"</h3><br/>");
        pw.println("address is:<h3>"+address+"</h3><br/>");
    }
}

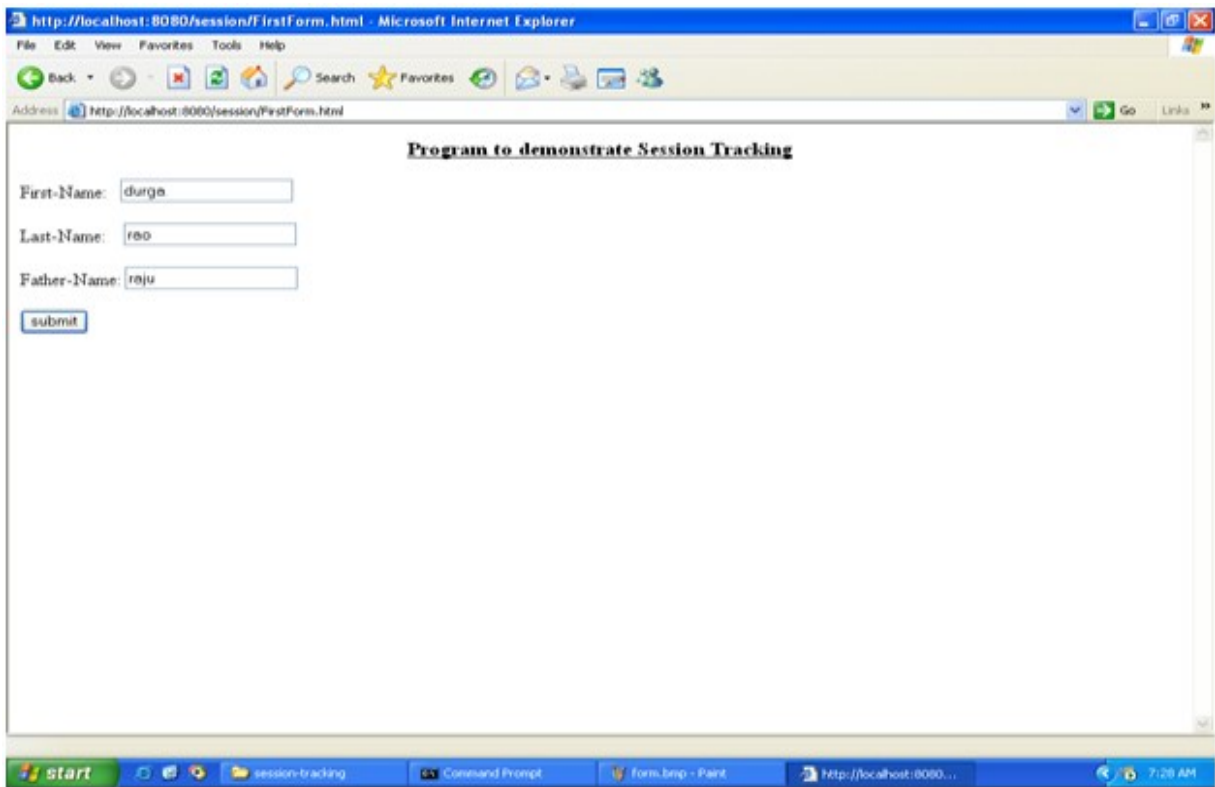
```

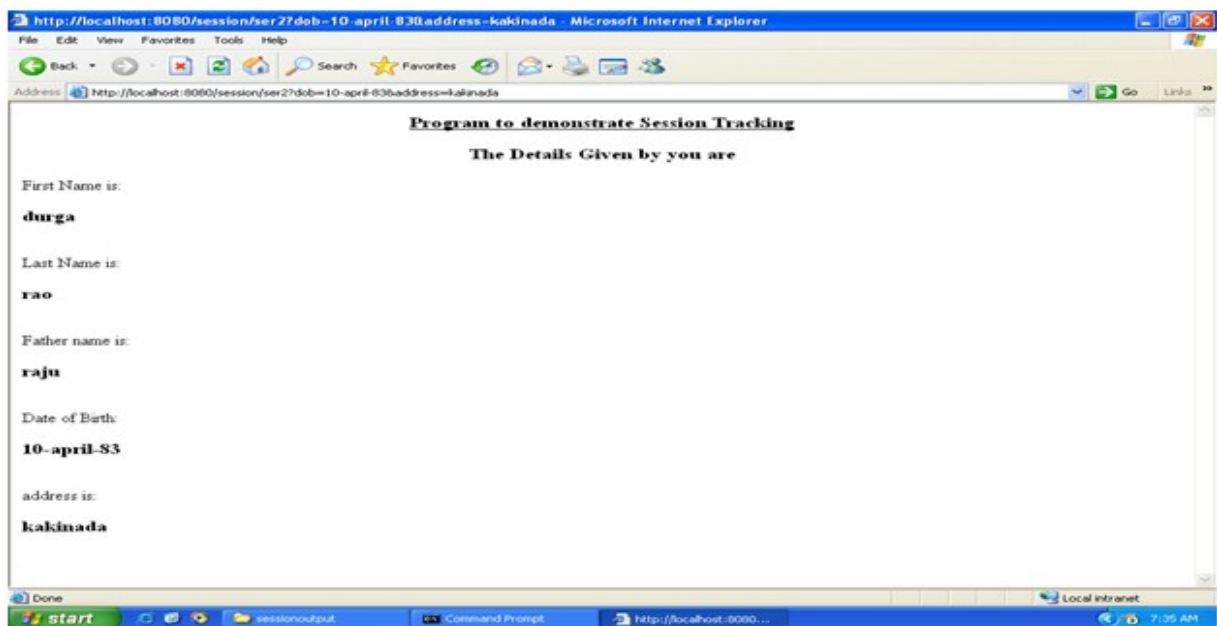
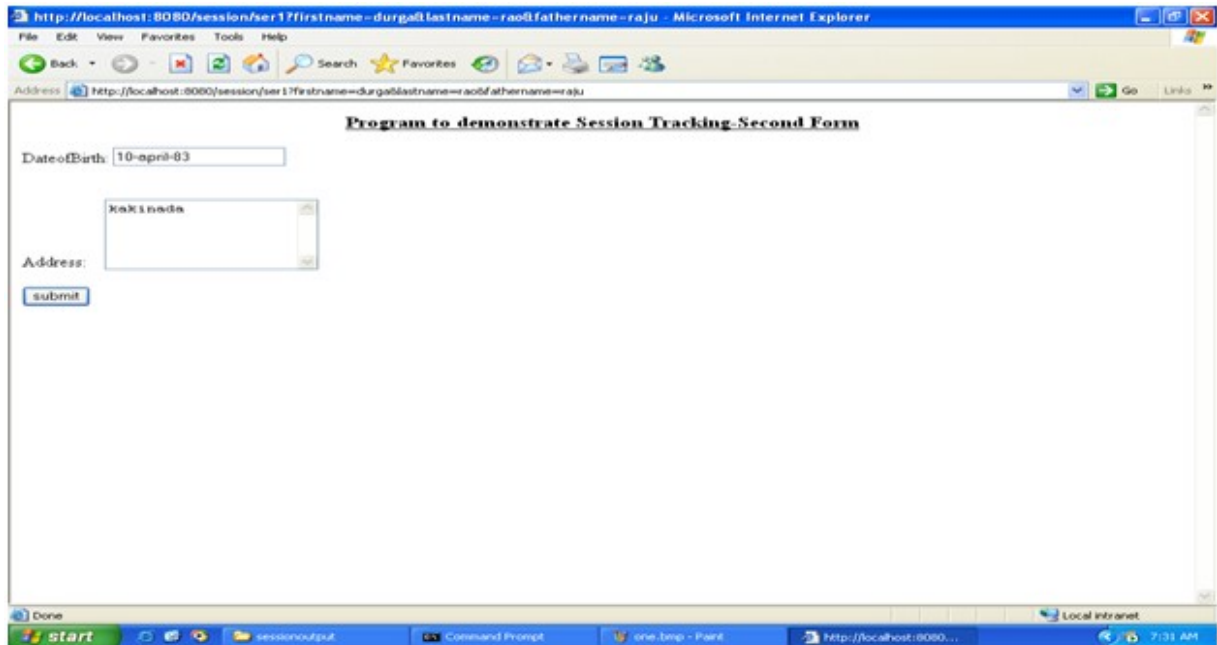
```
}  
}
```

Compilation:

```
Javac FirstServlet.java  
Javac SecondServlet.java
```

Output:





Program 34:

Install TOMCAT web server. Convert the static webpages of assignments 2 into dynamic webpages using servlets and cookies. Hint: Users information (user id, password, credit card number)

would be stored in web.xml. Each user should have a separate Shopping Cart.

PROCEDURE :

First install the tomcat into the system.
Then make a subdirectory(eg., tr) in the \tomcat\webapps.
Under tr create WEB-INF directory and also place the html files in this tr directory only.
Next under WEB-INF create two subclasses lib,classes and web.xml
Next place all the class files under the classes and jar files(servlet-api.jar,classes12.jar etc..) under lib subdirectories.
After this start tomcat by giving the following command at the instll_dir>tomcat>bin
Catalina.bat run
At the I.E(web browser) give the url as
http://localhost:8080//tr/htmlfile or servlet url pattern
Portno 8080 is assigned for the tomcat.

Web.xml :

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
  <display-name>Servlet 2.4 Examples</display-name>
  <description>
    Servlet 2.4 Examples.
  </description>
  <servlet>
    <servlet-name>reg</servlet-name>
    <servlet-class>reg</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>login</servlet-name>
    <servlet-class>login</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>profile</servlet-name>
    <servlet-class>profile</servlet-class>
  </servlet>
</web-app>
```

```

<servlet>
    <servlet-name>catalog</servlet-name>
    <servlet-class>catalog</servlet-class>
<servlet-mapping>
    <servlet-name>order</servlet-name>
    <url-p</servlet>
<servlet>
    <servlet-name>order</servlet-name>
    <servlet-class>order</servlet-class>
</servlet>
attern>order</url-pattern>
</servlet-mapping>
<servlet-mapping>
    <servlet-name>catalog</servlet-name>
    <url-pattern>catalog</url-pattern>
</servlet-mapping>
<servlet-mapping>
    <servlet-name>profile</servlet-name>
    <url-pattern>profile</url-pattern>
</servlet-mapping>
<servlet-mapping>
    <servlet-name>login</servlet-name>
    <url-pattern>login</url-pattern>
</servlet-mapping>
<servlet-mapping>
    <servlet-name>reg</servlet-name>
    <url-pattern>reg</url-pattern>
</servlet-mapping>
</web-app>

```

Main.html:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<body bgcolor="pink">
<br /><br /><br /><br /><br />
<h1 align="center"><U>ONLINE BOOK STORAGE</U></h1><br /><br
/><br />
<h2 align="center"><pre>
<b>Welcome to online book storage.
Press LOGIN if you are having id

```

```

otherwise press REGISTRATION
</b></pre></h2>
<br /><br /><pre>
<div align="center"><a href="/tr/login.html">LOGIN</a> <a
href="/tr/reg.html">
REGISTRATION</a></div></pre></body></html>

```

Login.html :

```

<html>
<body bgcolor="pink"><br /><br /><br />
<form name="myform" method="post" action="/tr/login">
<div align="center"><pre>
LOGIN ID :<input type="text" name="id" /><br />
PASSWORD :<input type="password" name="pwd" /></pre><br /><br />
</div>
<br /><br />
<div align="center">
<input type="submit" value="ok" onclick="validate()" />
<input type="reset" value="clear" />
</div>
</form>
</body>
</html>

```

Reg.html :

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<body bgcolor="pink"><br /><br />
<form name="myform" method="post" action="/tr/reg">
<div align="center"><pre>
NAME:<input type="text" name="name" /><br />
ADDRESS :<input type="text" name="addr" /><br />
CONTACT NUMBER :<input type="text" name="phno" /><br />
LOGINID :<input type="text" name="id" /><br />
PASSWORD :<input type="password" name="pwd" /></pre><br /><br />
</div>
<br /><br />
<div align="center">

```



```
<input type="submit" value="ok" onclick="validate()" />
                <input type="reset"
value="clear" />
</div>
</form>
</body>
</html>
```

Profile.html:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<body bgcolor="pink"><br /><br /><br />
<form name="myform" method="post" action="/tr/profile">
<div align="center"><pre>
LOGIN ID :<input type="text" name="id" /><br />
</pre><br /><br />
</div>
<br /><br />
<div align="center">
<input type="submit" value="ok" onclick="validate()" />
                <input type="reset"
value="clear" />
</div>
</form>
</body>
</html>
```

Catalog.html:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<body bgcolor="pink"><br /><br /><br />
<form method="post" action="/tr/catalog">
<div align="center"><pre>
BOOK TITLE :<input type="text" name="title" /><br />
</pre><br /><br />
</div>
<br /><br />
```



```

import javax.servlet.http.*;
public class login extends HttpServlet
{

public void service(HttpServletRequest req,HttpServletResponse
                    resp)throws ServletException,IOException
{
    PrintWriter pw=resp.getWriter();
    pw.println("<html><body bgcolor=\"pink\"");
    String id=req.getParameter("id");
    String pwd=req.getParameter("pwd");
    try
    {
        Driver d=new oracle.jdbc.driver.OracleDriver();
        DriverManager.registerDriver(d);
        Connection con=DriverManager.getConnection("
        jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger"
            );

        Statement stmt=con.createStatement();
        String sqlstmt="select id,password from login";
        ResultSet rs=stmt.executeQuery(sqlstmt);
        int flag=0;
        while(rs.next())
        {
            if(id.equals(rs.getString(1))&&pwd.equals(
                rs.getString(2)))
            {
                flag=1;
            }
        }
        if(flag==0)
        {
            pw.println("SORRY INVALID ID TRY AGAIN ID<br><br>");
            pw.println("<a href=\"/tr/login.html\">press LOGIN
                to RETRY</a>");
        }
        else
        {
            pw.println("VALID LOGIN ID<br><br>");

        pw.println("<h3><ul>");
    }
}
}

```

```

        pw.println("<li><a href=\"profile.html\"><font color=\"black\">USER PROFILE</font> </a></li><br><br>");
        pw.println("<li><a href=\"catalog.html\"><font color=\"black\">BOOKS CATALOG</font></a></li><br><br>");
        pw.println("<li><a href=\"order.html\"><font color=\"black\">ORDER CONFIRMATION</font>
                </a></li><br><br>");
    }
    pw.println("</body></html>");
}
catch (Exception e)
{
    resp.sendError(500, e.toString());
}
}
}

```

Reg.java:

```

import java.sql.*;
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class login extends HttpServlet
{
    public void service(HttpServletRequest req, HttpServletResponse
        resp) throws ServletException, IOException
    {
        PrintWriter pw=resp.getWriter();
        pw.println("<html><body bgcolor=\"pink\">");
        String name=req.getParameter("name");
        String addr=req.getParameter("addr");
        String phno=req.getParameter("phno");

        String id=req.getParameter("id");
        String pwd=req.getParameter("pwd");
        int no=Integer.parseInt(phno);
        try
        {
            Driver d=new oracle.jdbc.driver.OracleDriver();

```

```

DriverManager.registerDriver(d);
Connection con=DriverManager.getConnection("
jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger"
);

Statement stmt=con.createStatement();
String sqlstmt="select id,password from login";
ResultSet rs=stmt.executeQuery(sqlstmt);
int flag=0;
while(rs.next())
{
    if(id.equal(rs.getString(1))&&pwd.equals(
        rs.getString(2)))
    {
        flag=1;
    }
}
if(flag==1)
{
    pw.println("SORRY INVALID ID ALREADY EXISTS TRY
        AGAIN WITH NEW ID<br><br>");
    pw.println("<a href=\"/tr/reg.html\">press
        REGISTER to RETRY</a>");
}
else
{
    Statement stmt1=con.createStatement();
    stmt1.executeUpdate("insert into login
        values("+names+","+addr+","+no+","+id+","+pwd
        +")");

    pw.println("YOUR DETAILS ARE ENTERED<br><br>");
    pw.println("<a href=\"/tr/login.html\">press
        LOGIN to login</a>");
}
pw.println("</body></html>");
}
catch(Exception e)
{
    resp.sendError(500,e.toString());
}
}

```

```
}
```

Catlog.java:

```
import java.sql.*;
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class login extends HttpServlet
{
    public void service(HttpServletRequest req,
                        HttpServletResponse resp)
                        throws ServletException, IOException
    {
        PrintWriter pw=resp.getWriter();
        pw.println("<html><body bgcolor=\"pink\"");
        String title=req.getParameter("title");
        try
        {
            Driver d=new oracle.jdbc.driver.OracleDriver();
            DriverManager.registerDriver(d);

            Connection con=DriverManager.getConnection("
                jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger");
            Statement stmt=con.createStatement();
            String sqlstmt="select id,password from login";
            ResultSet rs=stmt.executeQuery(sqlstmt);
            int flag=0;
            while(rs.next())
            {
                pw.println(",div align=\"center\">");
                pw.println("TITLE:"+rs.getString(1)+"<br>");
                pw.println("AUTHOR  :"+rs.getString(2)+"<br>");
                pw.println("VERSION :"+rs.getString(3)+"<br>");
                pw.println("PUBLISHER   :"+rs.getString(4)+"<br>");
                pw.println("COST       :"+rs.getString(5)+"<br>");
                pw.println("</div>");
                flag=1;
            }
        }
    }
}
```

```

    }
    if(flag==0)
    {
        pw.println("SORRY INVALID TITLE TRY AGAIN <br><br>");
        pw.println("<a href=\"/tr/catalog.html\">press HERE to
                    RETRY</a>");
    }
    pw.println("</body></html>");
}
catch(Exception e)
{
    resp.sendError(500,e.toString());
}
}
}

```

Profile.java:

```

import java.sql.*;
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class login extends HttpServlet
{
    public void service(HttpServletRequest req,
                        HttpServletResponse resp) throws
                        ServletException, IOException
    {
        PrintWriter pw=resp.getWriter();
        pw.println("<html><body bgcolor=\"pink\">");
        String id=req.getParameter("id");
        try
        {
            Driver d=new oracle.jdbc.driver.OracleDriver();
            DriverManager.registerDriver(d);

```

```

Connection con=DriverManager.getConnection("
    jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger");
Statement stmt=con.createStatement();
String sqlstmt="select * from login where id="+id+"";
ResultSet rs=stmt.executeQuery(sqlstmt);
int flag=0;
pw.println("<br><br><br>");
while(rs.next())
{
    pw.println("<div align=\"center\">");
    pw.println("NAME      :"+rs.getString(1)+"<br>");
    pw.println("ADDRESS  :"+rs.getString(2)+"<br>");
    pw.println("PHONE NO   :"+rs.getString(3)+"<br>");
    pw.println("</div>");
    flag=1;

}
if(flag==0)
{
    pw.println("SORRY INVALID ID TRY AGAIN ID<br><br>");
    pw.println("<a href=\"/tr/profile.html\">press HERE
                to RETRY</a>");
}
pw.println("</body></html>");
}
catch(Exception e)
{
    resp.sendError(500,e.toString());
}
}
}

```

Order.java:

```

import java.sql.*;
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class login extends HttpServlet
{

```



```

public void service(HttpServletRequest req, HttpServletResponse
                    resp) throws ServletException, IOException
{
    PrintWriter pw=resp.getWriter();
    pw.println("<html><body bgcolor=\"pink\"");
    String id=req.getParameter("id");
    String pwd=req.getParameter("pwd");
    String title=req.getParameter("title");
    String count1=req.getParameter("no");
    String date=req.getParameter("date");

    String cno=req.getParameter("cno");
    int count=Integer.parseInt(count1);
    try
    {
        Driver d=new oracle.jdbc.driver.OracleDriver();
        DriverManager.registerDriver(d);
        Connection con=DriverManager.getConnection("
            jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger"
                                                    );

        Statement stmt=con.createStatement();
        String sqlstmt="select id,password from login";
        ResultSet rs=stmt.executeQuery(sqlstmt);
        int flag=0, amount, x;
        while(rs.next())
        {
            if(id.equals(rs.getString(1)) && pwd.equals(
                rs.getString(2)))
            {
                flag=1;
            }
        }
        if(flag==0)
        {
            pw.println("SORRY INVALID ID TRY AGAIN D<br><br>");
            pw.println("<a href=\"\"/tr/order.html\">press HERE
                to RETRY</a>");
        }
        else
        {
            Statement stmt2=con.createStatement();

```

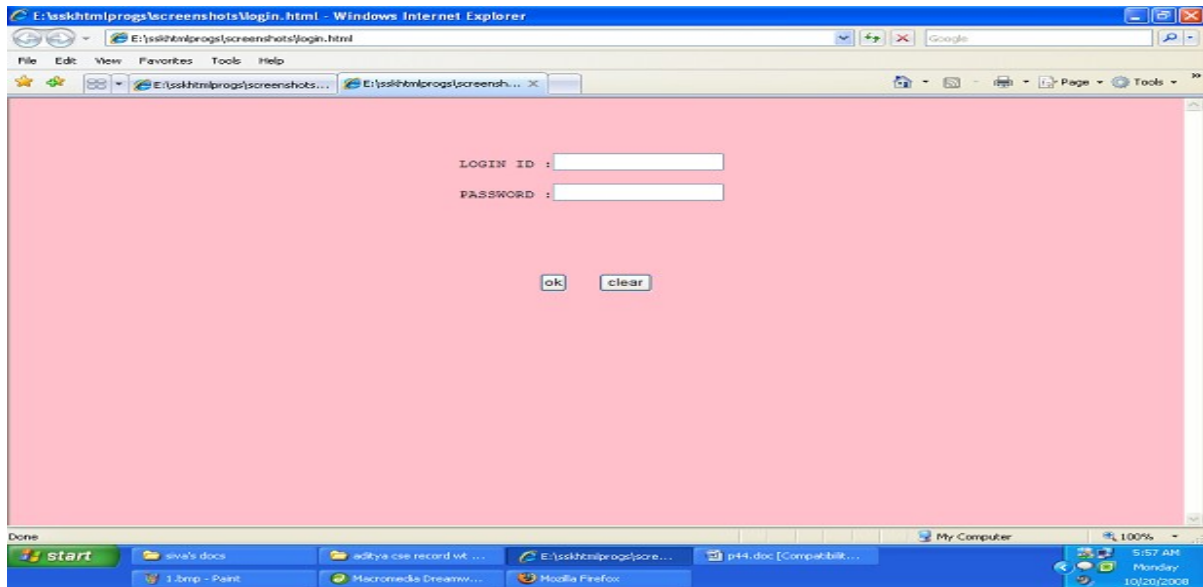
```

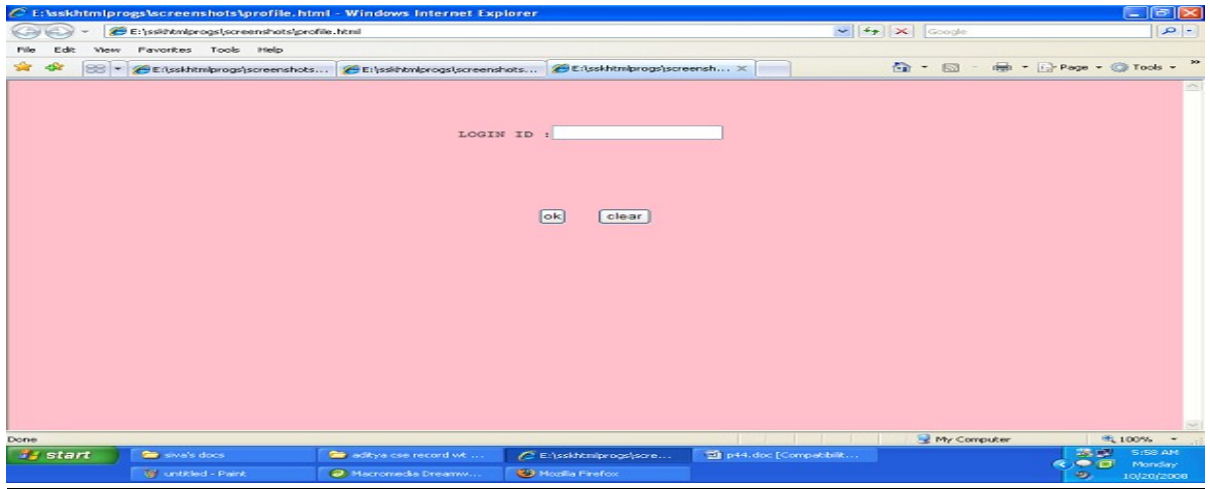
String s="select cost from book where
                                                title="+title+"";
ResultSet rs1=stmt2.executeQuery(s);
int flag1=0;
while(rs1.next())

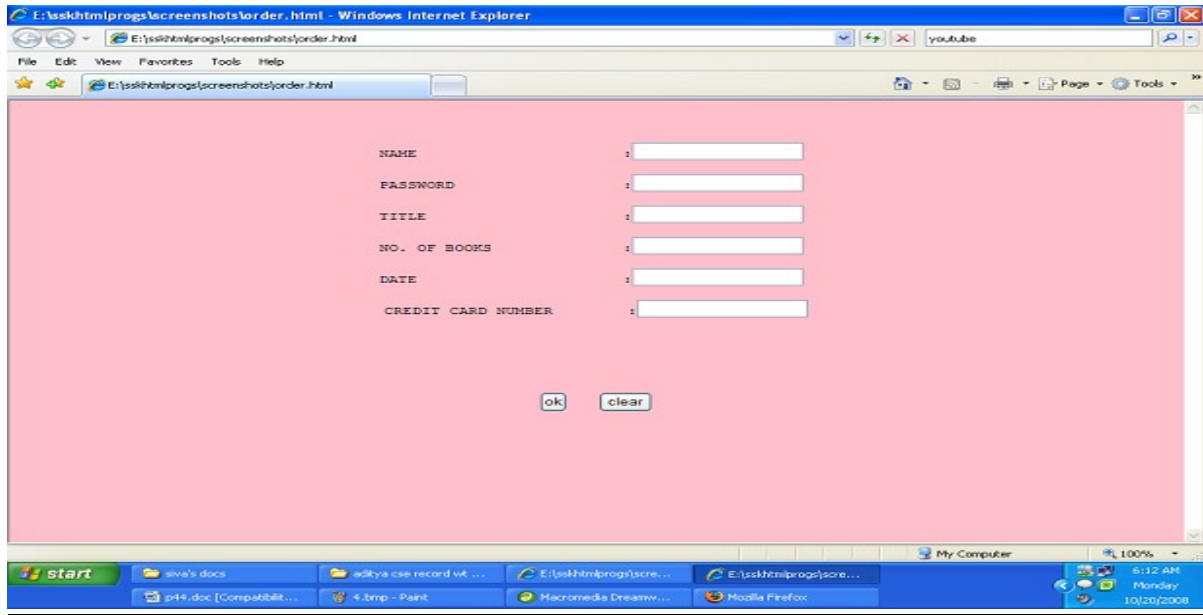
{
    flag1=1;
    x=Integer.parseInt(rs1.getString(1));
    amount=count*x;
    pw.println("AMOUNT:"+amount+"<br><br><br>");
    Statement stmt1=con.createStatement();
    stmt1.executeUpdate("insert into details
values('"+id+", '"+title+"'"+amount+"', '"+cno+"')");
    pw.println("YOUR ORDER has taken<br>");
}
if(flag1==0)
{
    pw.println("SORRY INVALID ID TRY AGAIN ID<br><br>");
    pw.println("<a href=\\\"/tr/order.html\\\">press HERE
to RETRY</a>");
}
}
pw.println("</body></html>");
con.close();
}
catch(Exception e)
{
    resp.sendError(500,e.toString());
}
}

```

Output:







Program 35:

Write a Program to Explain Accessing Java Beans From Java Server Pages

/javabeans/firstpage.html:

```
<html>
```

```

<body>
  <center><u><h3>
    Program to demonstrate accessing javabeans from jsp
  </h3></u></center>
  <form action="First.jsp">
    First-name:<input type="text" name="firstName"/>
    <br/><br/>
    Last-Name:<input type="text" name="lastName"/>
    <br/></br>
    Dateofbirth:<input type="text" name="dob"/><br/><br/>
    Address:<textarea name="address" rows="5" cols"10"/>
    </textarea><br/><br/>
    <input type="submit" value="submit"/>
  </form>
</body>
</html>

```

/javabeans/First.jsp:

```

<html>
  <body>
    <center><u><h3>
      Program to demonstrate accessing javabeans from jsp
    </h3></u></center><br/><br/>
    <%@page language="java" import="com.nit.beans.FirstBean" %>
    <jsp:useBean id="bean" scope="session"
      class="com.nit.beans.FirstBean">
      <jsp:setProperty name="bean" property="*" />
    </jsp:useBean>

    <center><a href="Second.jsp">
      <h2>click here to view Details</h2></a>
    </center>
  </body>
</html>

```

/javabeans/Second.jsp:

```

<html>
  <body>
    <center><u><h3>
      Program to demonstrate accessing javabeans from jsp
    </h3></u></center><br/><br/>
    <%@ page import="com.nit.beans.FirstBean" %>
    <jsp:useBean id="bean" scope="session"

```

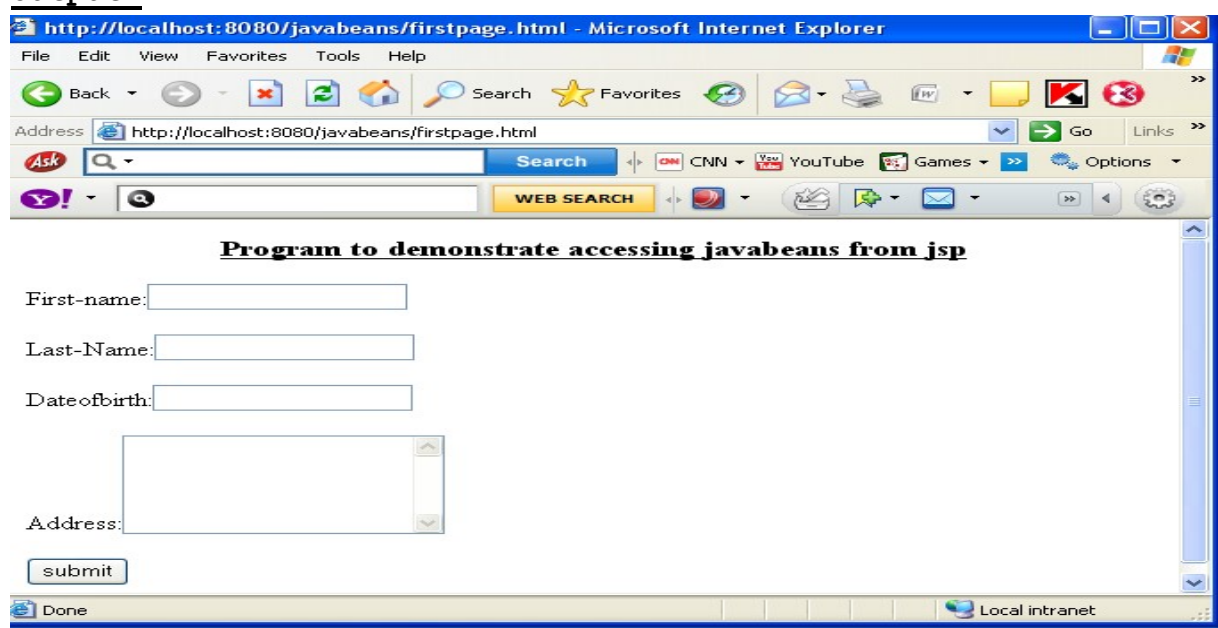


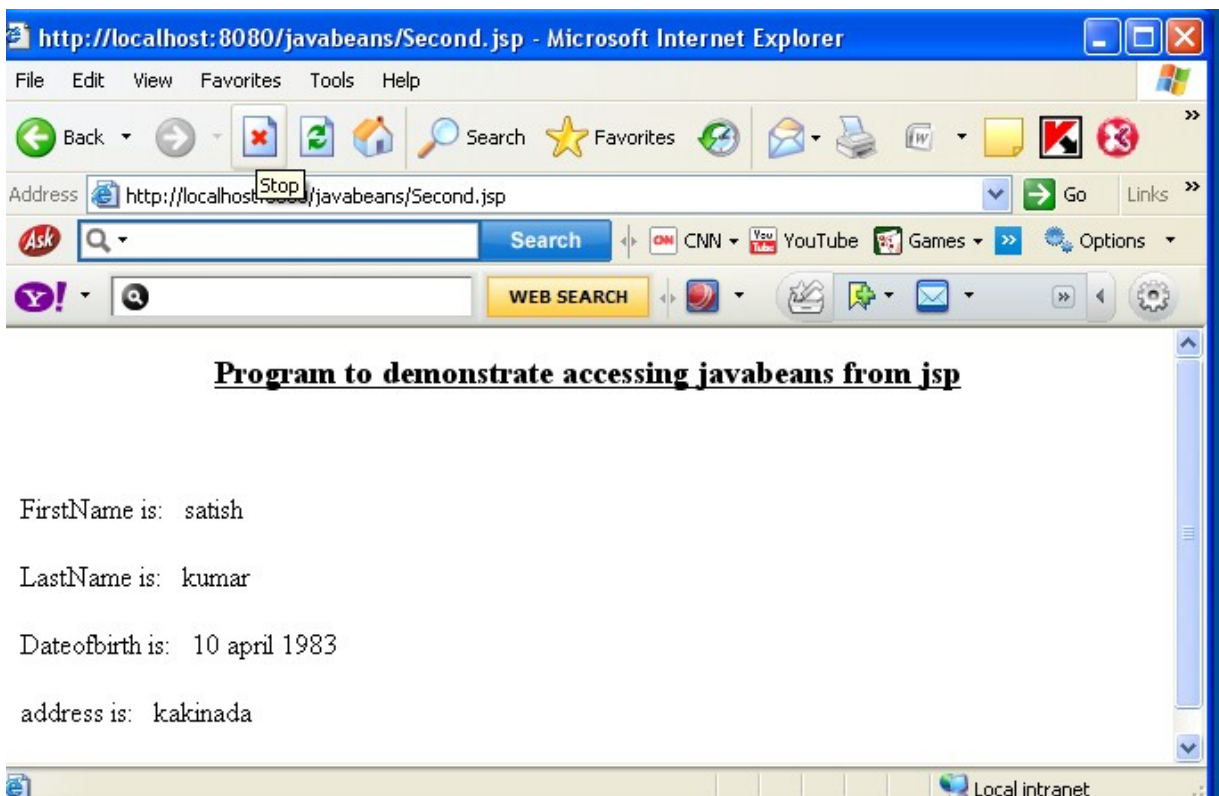
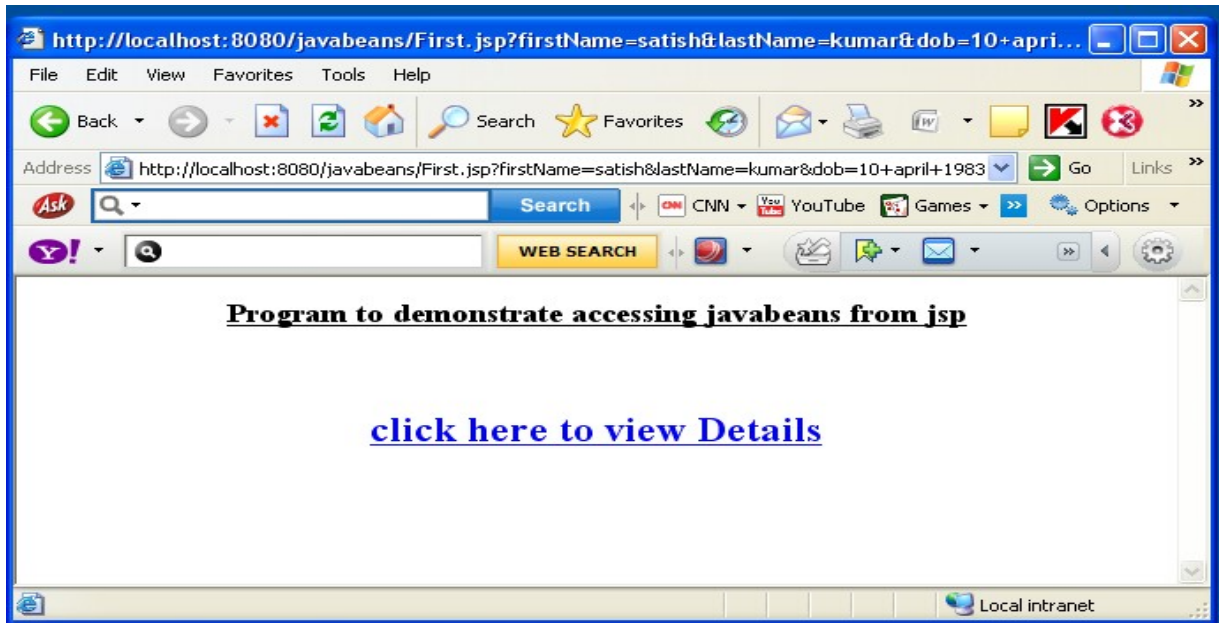
```
{
    return address;
}
public void setDob(String d)
{
    dob=d;
}
public String getDob()
{
    return dob;
}
}
```

Compilation:

Javac -d . FirstBean.java

Output:





Program 36:

Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

PROCEDURE:

- 1) Create your own directory under tomcat/webapps (e.g. tr1)
- 2) Copy the html files in tr1
- 3) Copy the jsp files also into tr1
- 4) Start tomcat give the following command
Catalina.bat run
At install-dir/bin
- 4) at I.E give url as <http://localhost:8081/tr1/main.html>

Main.html:

```
<html>
<body bgcolor="pink">
<br><br><br><br><br><br>
<h1 align="center"><u>ONLINE BOOK STORAGE</u></h1><br><br><br>
<h2 align="center"><PRE>
<b> Welcome to online book storage.
        Press LOGIN if you are having id
        Otherwise press REGISTRATION
</b></PRE></h2>
```



```

                                press LOGIN to RETRY</a>");
    }
    else
    {
        out.println("VALID LOGIN ID<br><br>");
        out.println("<h3><ul>");
        out.println("<li><a href=\"profile.html\">
                    <fontcolor=\"black\">USER
                    PROFILE</font></a></li><br><br>");
        out.println("<li><a href=\"catalog.html\">
                    <fontcolor=\"black\">BOOKS
                    CATALOG</font></a></li><br><br>");

        out.println("<li><a href=\"order.html\"><fontcolor=\"black\">ORDER CONFIRMATION</font></a></li><br><br>");
        out.println("</ul>");
    }
    out.println("<body></html>");
%>

```

Reg.jsp:

```

<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<%
    out.println("<html><body bgcolor=\"pink\">");
    String name=request.getParameter("name");
    String addr=request.getParameter("addr");
    String phno=request.getParameter("phno");
    String id=request.getParameter("id");
    String pwd=request.getParameter("pwd");
    int no=Integer.parseInt(phno);

    Driver d=new oracle.jdbc.driver.OracleDriver();
    DriverManager.registerDriver(d);
    Connection con=DriverManager.getConnection
        ("jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger");
    Statement stmt=con.createStatement();
    String sqlstmt="select id from login";
    ResultSet rs=stmt.executeQuery(sqlstmt);
    int flag=0;
    while(rs.next())
    {
        if(id.equals(rs.getString(1)))
        {
            flag=1;
        }
    }

```

```

    }
    if(flag==1)
    {
        out.println("SORRY LOGIN ID ALREADY EXISTS TRY AGAIN WITH
                    NEW ID <br><br>");
        out.println("<a href=\""/tr1/reg.html\">press REGISTER to
                    RETRY</a>");
    }
    else
    {
        Statement stmt1=con.createStatement ();
        stmt1.executeUpdate ("insert into login values
                             (+name+", "+addr+", "+no+", "+id+", "+pwd+)");
        out.println ("YOU DETAILS ARE ENTERED <br><br>");
        out.println ("<a href =\""/tr1/login.html\">press
                    LOGIN to login</a>");
    }
    out.println ("</body></html>");
%>

```

Profile.jsp:

```

<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<%
    out.println ("<html><body bgcolor=\"pink\">");
    String id=request.getParameter("id");
    Driver d=new oracle.jdbc.driver.OracleDriver();
    DriverManager.regiserDriver(d);
    Connection con=DriverManager.getConnection
        ("jdbc:oracle:thin:@localhost:1521:orcl","scott",
        "tiger");

    Statement stmt=con.createStatement ();
    String sqlstmt="select * from login where id="+id+"";
    ResultSet rs=stmt.executeQuery (sqlstmt);
    int flag=0;
    while(rs.next())
    {
        out.println ("<div align=\"center\">");
        out.println("NAME          :"+rs.getString(1)+"<br>");
    }

```

```

        out.println ("ADDRESS      :"+rs.getString(2)+"<br>");
        out.println ("PHONE NO    :"+rs.getString(3)+"<br>");
        out.println ("</div>");
        flag=1;
    }
    if(flag==0)
    {
        out.println("SORRYINVALID ID TRY AGAIN ID <br><br>");
        out.println("<a href=\"\"/tr1/profile.html\">press HERE
                                to RETRY </a>");
    }
    out.println ("</body></html>");
%>

```

Catalog.jsp:

```

<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<%
    out.println ("<html><body bgcolor=\"pink\">");
    String title=request.getParameter ("title");
    Driver d=new oracle.jdbc.driver.OracleDriver ();
    DriverManager.regiserDriver (d);
    Connection con=DriverManager.getConnection
        ("jdbc:oracle:thin:@localhost:1521:orcl",
        "scott","tiger");
    Statement stmt=con.createStatement ();
    Stringsqlstmt="select * from book where title="+title+"";
    ResultSet rs=stmt.executeQuery (sqlstmt);
    int flag=0;
    while(rs.next())
    {
        out.println ("<div align=\"center\">");
        out.println("TITLE      :"+rs.getString(1)+"<br>");
        out.println ("AUTHOR      :"+rs.getString(2)+"<br>");
        out.println ("VERSION:"+rs.getString(3)+"<br>");
        out.println("PUBLISHER   :"+rs.getString(4)+"<br>");
        out.println ("COST       :"+rs.getString(5)+"<br>");
        out.println ("</div>");
        flag=1;
    }
    if(flag==0)
    {
        out.println("SORRY INVALID ID TRY AGAIN ID<br><br>");
        out.println("<a href=\"\"/tr1/catalog.html\">press HERE
                                to RETRY </a>");
    }
    out.println ("</body></html>");

```


%>

Order.jsp:

```
<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<%
    out.println("<html><body bgcolor=\"pink\">");
    String id=request.getParameter("id");
    String pwd=request.getParameter("pwd");
    String title=request.getParameter("title");
    String count1=request.getParameter("no");
    String date=request.getParameter("date");
    String cno=request.getParameter("cno");
    int count=Integer.parseInt(count1);
    Driver d=new oracle.jdbc.driver.OracleDriver ();
    DriverManager.regiserDriver (d);
    Connection con=DriverManager.getConnection
        ("jdbc:oracle:thin:@localhost:1521:orcl",
        "scott","tiger");
    Statement stmt=con.createStatement ();
    String sqlstmt="select id, password from login";
    ResultSet rs=stmt.executeQuery (sqlstmt);
    int flag=0, amount, x;
    while(rs.next())
    {
        if(id.equals(rs.getString(1)) &&
        pwd.equals(rs.getString(2)))
        {
            flag=1;
        }
    }
    if(flag==0)
    {
        out.println("SORRY INVALID ID TRY AGAIN ID <br>");
        out.println("<a href=\"/tr1/order.html\">press HERE
            to RETRY </a>");
    }
    else
    {
        Statement stmt2=con.createStatement();
        String s="select cost from book where title=
            "+title+"";
        ResultSet rs1=stmt2.executeQuery(s);
        int flag1=0;
        while(rs1.next())
        {
            flag1=1;
        }
    }
%>
```

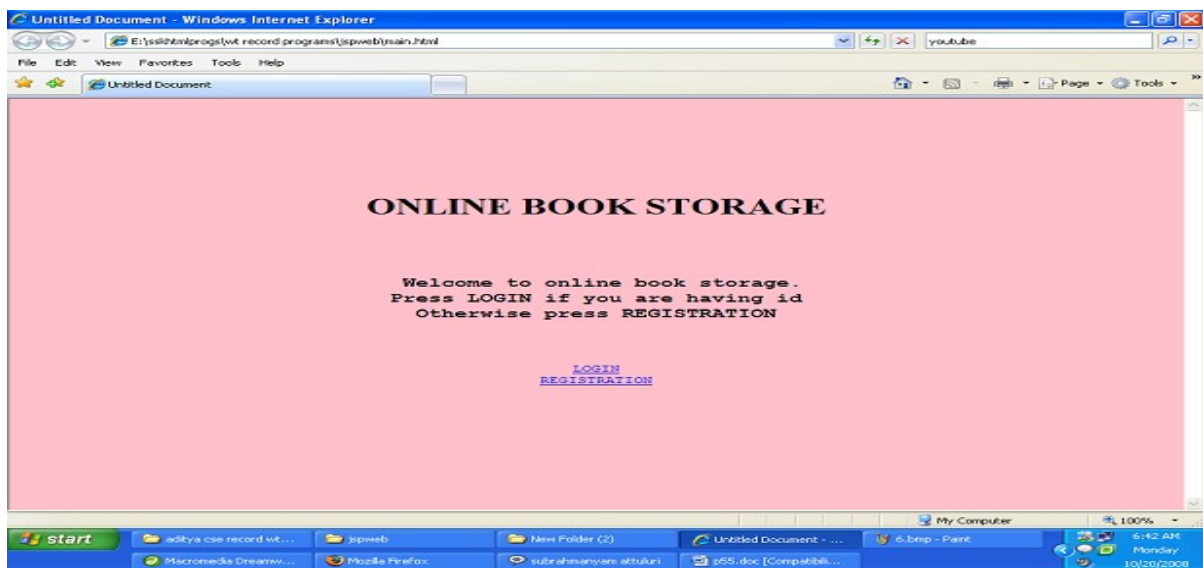
```

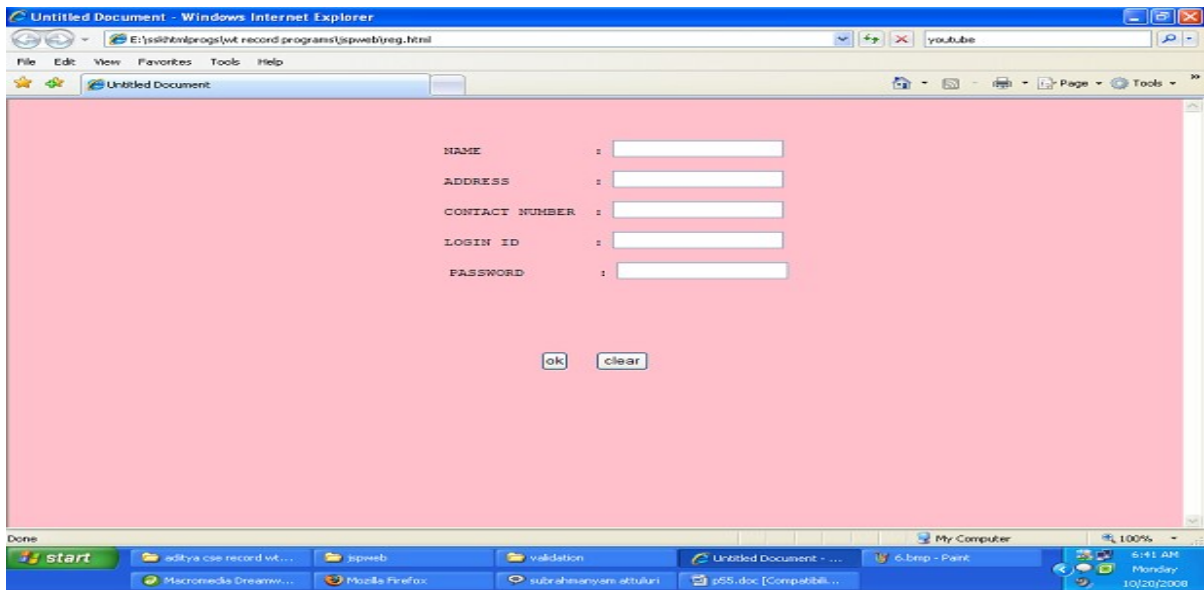
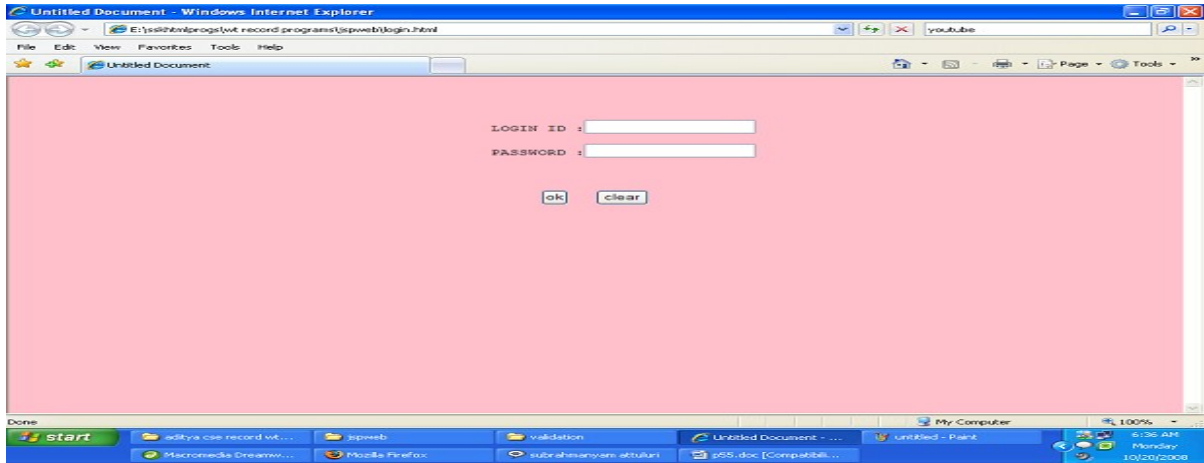
x=Integer.parseInt(rs1.getString(1));
amount=count*x;
out.println("AMOUNT:"+amount+"<br><br>");
Statement stmt1=con.createStatement ();
stmt1.executeUpdate("insert into details ("+id+",
        "+title+", "+amount+", "+date+", "+cno+)");
out.println ("YOU ORDER HAS TAKEN<br>");
    }

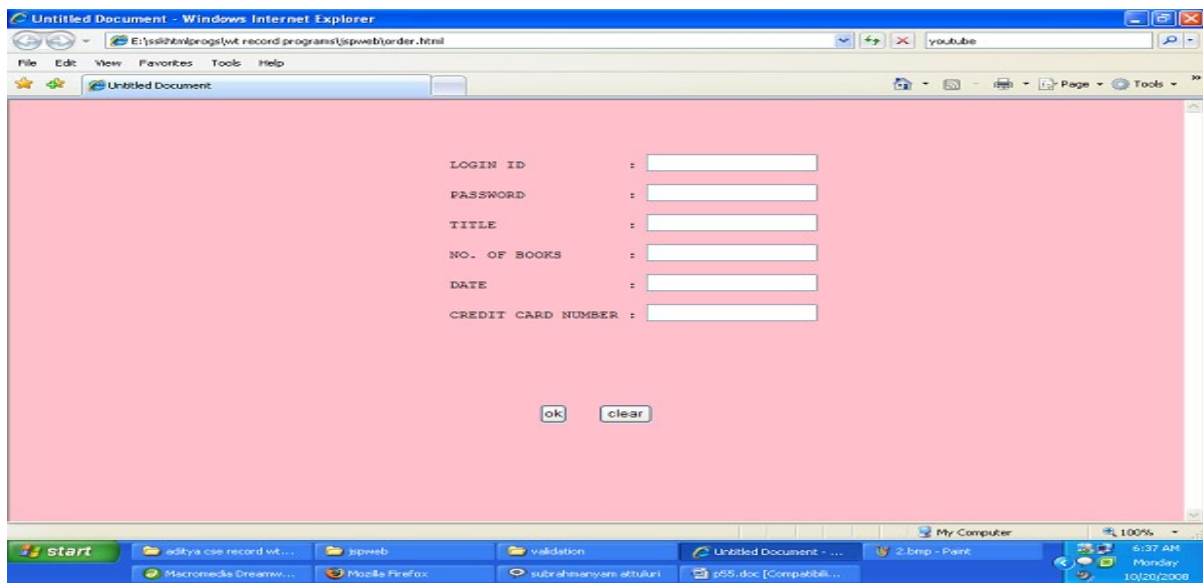
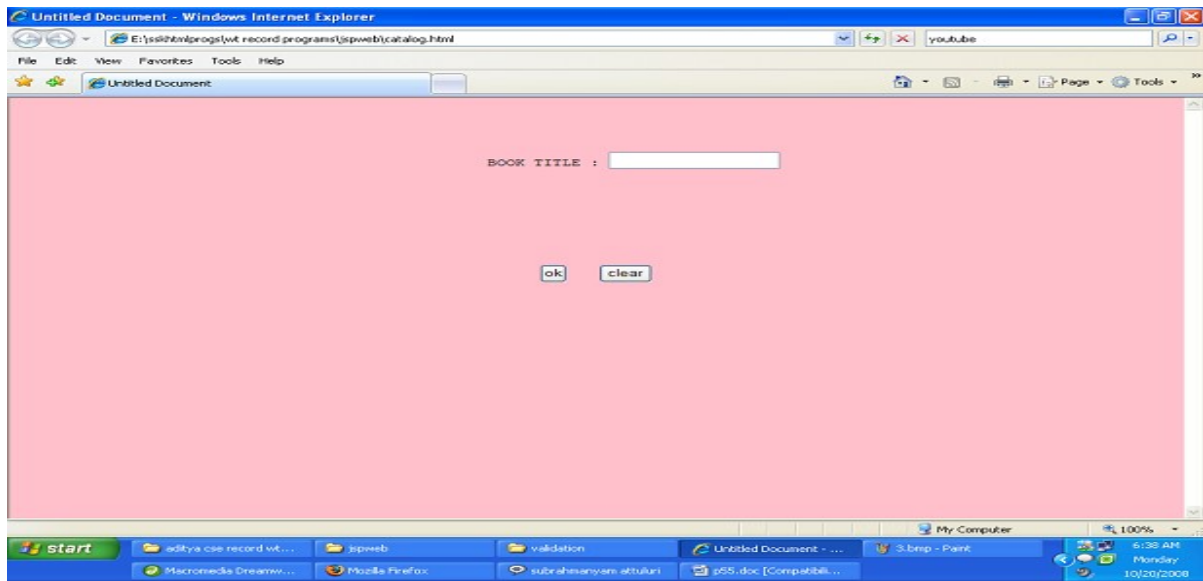
if(flag1==0)
{
    out.println("SORRY INVALID BOOK TRY AGAIN <br><br>");
    out.println("<a href='\"/tr1/order.html\"'>press HERE to
        RETRY </a>");
}
}
out.println ("</body></html>");
%>

```

Output:







Program 37:

Write a jdbc program to retrieve empno,ename,job from the emp Table

ResultSetEx.java:

```

import java.sql.*;
public class ResultSetEx
{
    public static void main(String args[]) throws Exception
    {
        Class.forName("oracle.jdbc.driver.OracleDriver");
        Connection con=DriverManager.getConnection("jdbc:
            oracle:thin:@localhost:1521:mrec","scott","tiger");
        Statement st=con.createStatement();
        ResultSet rs=st.executeQuery("select empno,ename,job from
                                        emp");

        while(rs.next())
        {
            int no=rs.getInt(1);
            String name=rs.getString(2);
            String job=rs.getString("job");
            System.out.println(no+"\t"+name+"\t"+job+"\n");
        }
        rs.close();
        st.close();
        con.close();
    }
}

```

Compilation:

Javac ResultSetEx.java

Execution:

Java ResultSetEx

Output:

```

7369 SMITH      CLERK
7499 ALLEN      SALESMAN
7521 WARD       SALESMAN
7566 JONES      MANAGER
7654 MARTIN     SALESMAN
7698 BLAKE      MANAGER
7782 CLARK      MANAGER
7788 SCOTT      ANALYST
7839 KING       PRESIDENT
7844 TURNER     SALESMAN
7876 ADAMS      CLERK

```

7900	JAMES	CLERK
7902	FORD	ANALYST
7934	MILLER	CLERK

Program 38:

Write a jdbc program to insert empno,empname,job into emp table using PreparedStatement

InsertEx.java:

```
import java.sql.*;
public class InsertEx
{
    public static void main(String args[]) throws Exception
    {
        Class.forName("oracle.jdbc.driver.OracleDriver");
        Connection con=DriverManager.getConnection("jdbc:
            oracle:thin:@localhost:1521:mrec","scott","tiger");
        PreparedStatement ps=con.prepareStatement("insert into
            emp(empno,ename,job) values(?,?,?)");
```

```

    ps.setInt(1,101);
    ps.setString(2,"satish");
    ps.setString(3,"teamlead");
    int i=ps.executeUpdate();
    System.out.println("row inserted successfully");
    ps.setInt(1,201);
    ps.setString(2,"prasad");
    ps.setString(3,"manager");
    i+=ps.executeUpdate();
    System.out.println(i+"rows inserted successfully");
    con.close();
}
}

```

Compilation:

Javac InsertEx.java

Execution:

Java InsertEx

Output:

row inserted successfully
2rows inserted successfully

Program 39:

Write a jdbc program to retrieve data from a given table name using ResultSetMetaData interface

RSMDEX.java:

```

import java.sql.*;
import java.util.*;
public class RSMDEX
{
    public static void main(String s[])throws Exception
    {
        String tablename=s[0];
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection con=DriverManager.getConnection("jdbc:
            odbc:cseb","scott","tiger");
        Statement st=con.createStatement();
        ResultSet rs=st.executeQuery("select * from "+tablename);
        ResultSetMetaData rsmd=rs.getMetaData();
        int col_count=rsmd.getColumnCount();
        System.out.println("-----");
        for(int i=1;i<=col_count;i++)

```

```

    {
        System.out.print(rsmd.getColumnName(i) + "\t");
    }
    System.out.println();
    System.out.println("-----");
    while(rs.next())
    {
        for(int i=1;i<=col_count;i++)
        {
            int type=rsmd.getColumnType(i);
            if(type==Types.INTEGER)
            {
                System.out.print(rs.getInt(i) + "\t");
            }
            else if(type==Types.DOUBLE)
            {
                System.out.print(rs.getDouble(i) + "\t");
            }
            else
            {
                System.out.print(rs.getString(i) + "\t");
            }
        }
        System.out.println();
    }
}

```

Output:

```

-----
DEPTNO      DNAME          LOC
-----
10    ACCOUNTING    NEW YORK
20    RESEARCH      DALLAS
30    SALES         CHICAGO
40    OPERATIONS    BOSTON

```