1. DESIGN A WEB SKELITON IN THE Following requirements

1. a left navigation(200PX,COLOR #2381ff),RIGHT NAVIGATION(200PX COLOR #8123AA), CONTENT AREA

HEADER (300PX),FOOTER(100PX),REMAINING CONTENT AREA.

1. ALL THE SECTIONS SHOWN ABOVE ARE TO BE STYLED APPROPIATELY USING CSS

HTML

<!DOCTYPE html>

<html>

<head>

<title>Basic 3 Column Layout</title>

<link rel="stylesheet" href="STY.CSS" type="text/css" />

</head>

<body>

<div id="container">

<div id="header">

<b>This is Header

</b>

</div>

<div>

<nav>

This is Navigation Bar

</nav>

</div>

<div id="sidebar-left">

<h2>Sidebar Left</h2>

<p>this is the side bar-left</p>

</div>

<div id="content">

<h1>Content</h1>

<p>this is the content bar</p>

</div>

<div id="sidebar-right">

<h2>Sidebar Right</h2>

<p>this is the sidebar right </p>

</div>

<div id="footer">

<h3>Footer</h3>

<p>this is the footer </p>

</div>

</div>

</body>

</html>

STY.CSS

body {

text-align: center;

}

#container {

margin: 0 auto;

text-align: left;

width: 960px;

}

#header {

width: 300px;

}

#navigation li {

display: inline;

}

#content {

float: left;

width: 560px;

}

#sidebar-left {

float: left;

width: 200px;

}

#sidebar-right {

float: right;

width: 200px;

}

#footer {

clear: both;

width: 100px;

}

body {

background-color: #eee;

color: white;

border-style:solid;

border-color:black;

}

#container {

background-color:pink;

}

#header {

background-color: aqua;

height: 100%;

width:100%;

text-align:center;

font-size:50px;

}

#content {

background-color: red;

height: 100%;

position:relative;

}

nav{

background-color:pink;

height: 100%;

font-size:30px;

text-align:center;

position:relative;

}

#sidebar-left {

background-color: green;

height:100%;

position:relative;

padding-top:13px;

}

#sidebar-right {

background-color: blue;

height: 100%;

position:relative;

padding-top:13px;

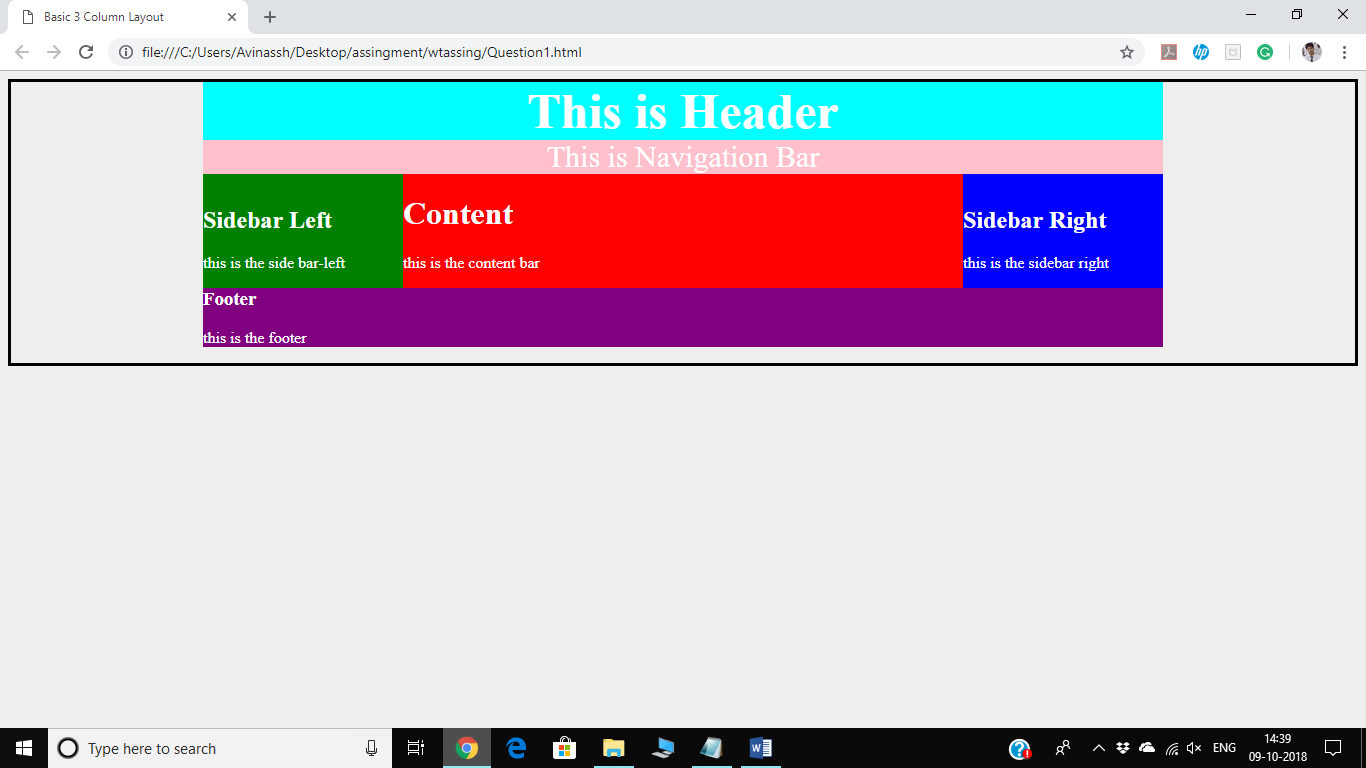
}

#footer {

background-color: purple;

width: 100%;

}



2) DEMOSTRATE THE USE OF CSS POSITIONING ELEMENTS STATIC ABSOLUTE,RELATIVE AND FIXED

<!DOCTYPE html>

<html>

<head>

<style>

div.static {

position:static;

border:3px solid grey;

}

div.fixed{

position:fixed;

bottom:0;

right:0;

width:300px;

border:3px solid red;

}

div.relative {

position: relative;

width: 400px;

height: 200px;

border: 3px solid pink;

}

div.absolute {

position: absolute;

top: 80px;

right: 0;

width: 200px;

height: 100px;

border: 3px solid yellow;

}

div.sticky{

position: -webkit-sticky;

position:sticky;

top:0;

padding:5px;

border: 3px solid green;}

</style>

</head>

<body>

<div class="static">normal</div>

<div class="relative">relative</div>

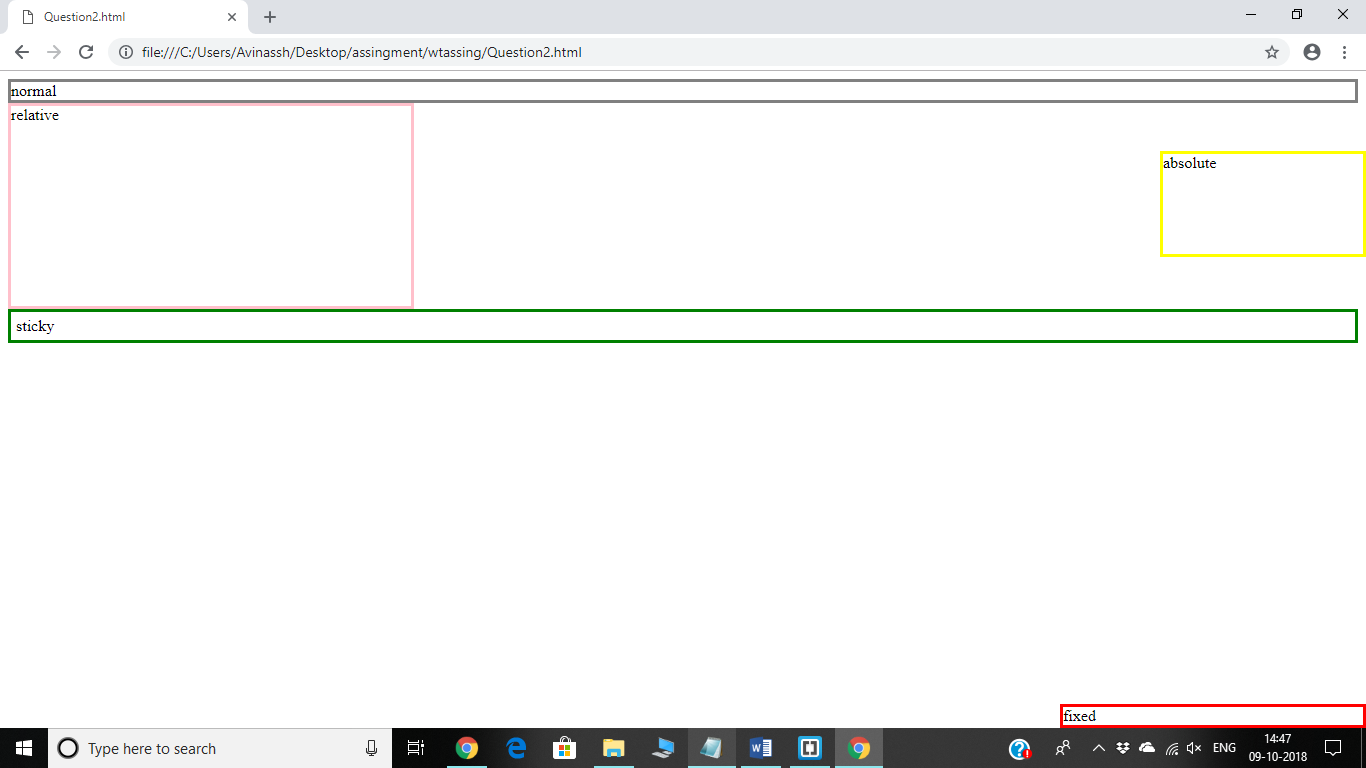
<div class="fixed">fixed</div>

<div class="absolute">absolute</div>

<div class="sticky">sticky</div>

</body>

</html>



3) CREATE A HTML5 FORM AND VALIDATE THE FORM CONTROLS USING

"pattern" attribute. Form must contain the following elements

a. Email

b. a text field

c. number field

d. range

e. min length, max length

f. alpha-numeric

HTML

<!DOCTYPE html>

<html>

<head>

<style>

body{

text-align:center;

background-color:;

}

</style>

</head>

<body>

<header>

<h1>SignUp Form</h1>

</header>

<div style="">

<form method="POST">

<label>Email id</label><br>

<input id="textbox" type="text" name="email" pattern="[a-z0-9.\_%+-]+@[a-z0-9.-]+\.[a-z]{2,3}$" required><br>

<label>Password</label><br>

<input type="password" name="password" pattern="(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}" required><br>

<label>Age</label><br>

<input type="number" name="age" pattern="[10-50]{1,2}$" required><br>

<label>Enter text here</label><br>

<input type="text" name="text" required><br>

<label>Number</label><br>

<input type="text" name="number" pattern="[0-9]{10}" required><br>

<label>Range bar</label><br>

<input type="range" min="1" max="100" value="50" pattern="[0-100]"><br>

<label>Password</label><br>

<input type="password" name="password" pattern="[a-z0-9]{8,}" required><br>

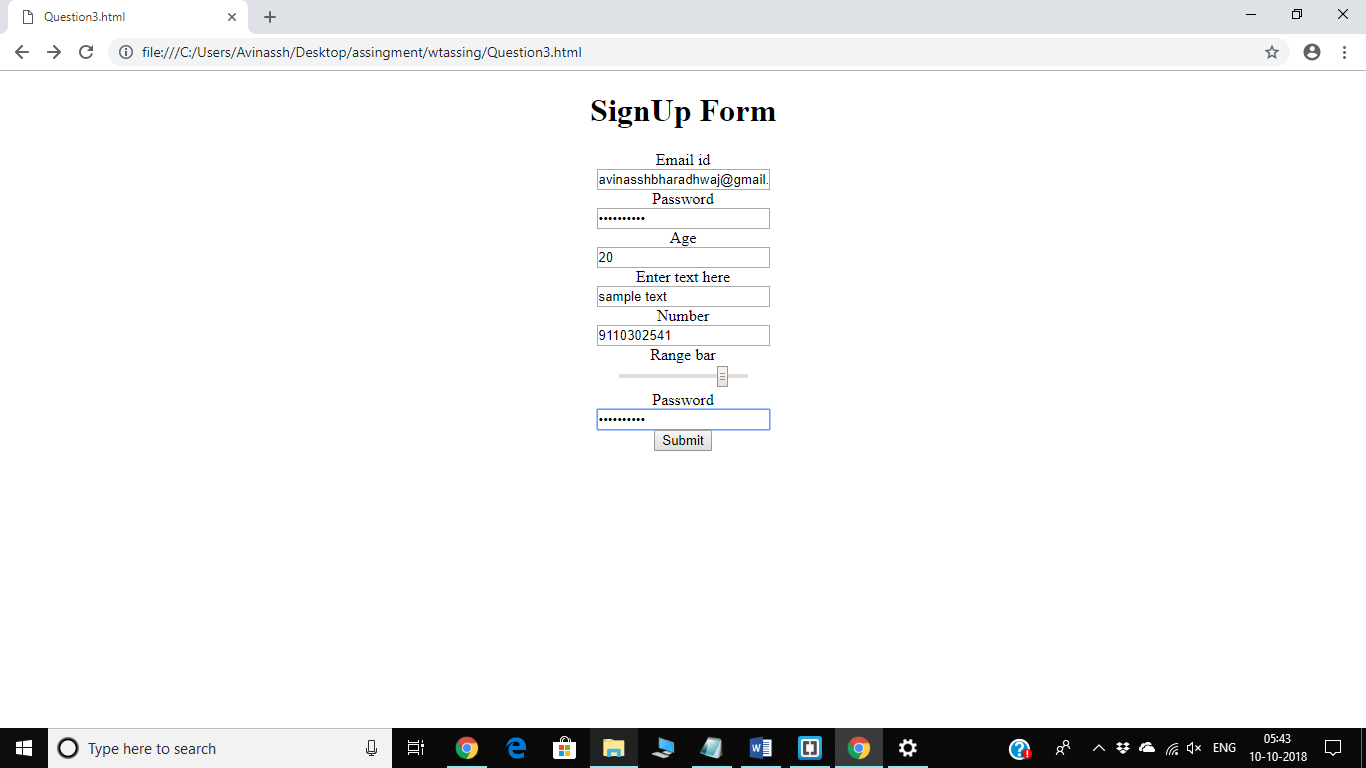
<input type="submit" name="submit"><br>

</form>

</div>

</body>

</html>



4) Write a DTD for Aadhar registration process and write a well formed and valid XML for the above created DTD.

<?xml version="1.0"?>

<!DOCTYPE AADHAR\_REG [

<!ENTITY ISSUER "Government of India">

<!ENTITY COMPANY "UDAAI">

<!ENTITY EMAIL "ADHAARINDIA.com">

<!ELEMENT ISSUER (ACARD+)>

<!ELEMENT ACARD

(NAME+,EMAIL+,NUMBER+,ADDRESS+,BIOMETRIC?)>

<!ELEMENT NAME (#PCDATA)>

<!ELEMENT EMAIL (#PCDATA)>

<!ELEMENT NUMBER (#PCDATA)>

<!ELEMENT ADDRESS (#PCDATA)>

<!ELEMENT BIOMETRIC (#PCDATA)>

]>

<ISSUER>

<ACARD>

<NAME>Avinassh</NAME>

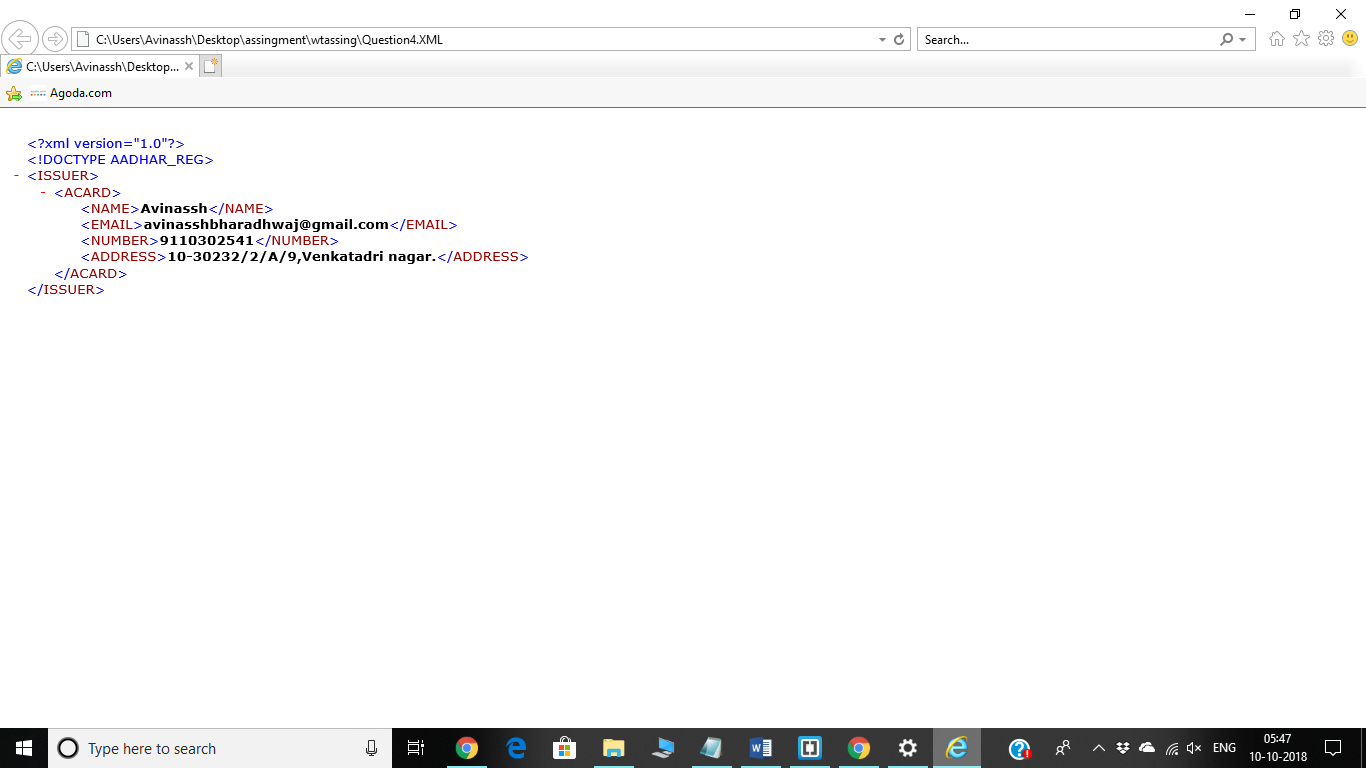
<EMAIL>avinasshbharadhwaj@gmail.com</EMAIL>

<NUMBER>9110302541</NUMBER>

<ADDRESS>10-30232/2/A/9,Venkatadri nagar.</ADDRESS>

</ACARD>

</ISSUER>



5) Write a XML schema for library management system and write a valid and well formed XML example.

XSD

<?xml version="1.0"?>

<xs:schema>

<xs:element name="Issue">

<xs:complexType>

<xs:sequence>

<xs:element name="to" type="xs:string"/>

<xs:element name="toid" type="xs:integer"/>

<xs:element name="bookid" type="xs:string"/>

<xs:element name="dateissue" type="xs:date"/>

<xs:element name="datereturn" type="xs:date"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

XML

<?xml version="1.0"?>

<Issue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="Libman.xsd">

<to>Student</to>

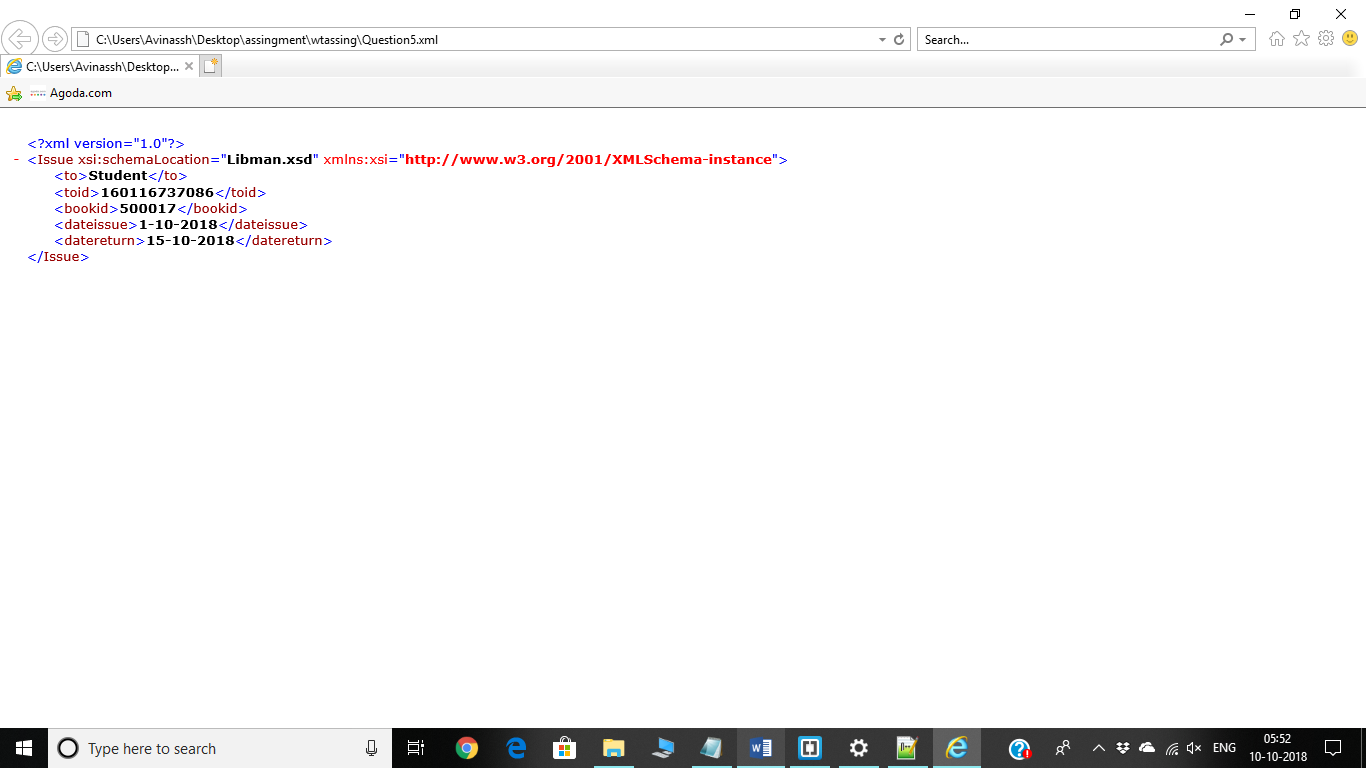
<toid>160116737086</toid>

<bookid>500017</bookid>

<dateissue>1-10-2018</dateissue>

<datereturn>15-10-2018</datereturn>

</Issue>



6.Write a simple servlet program that displays HelloWorld message

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class NewServlet extends HttpServlet

{

public void doGet(HttpServletRequest request,HttpServletResponse response) throws ServletException,IOException

{

response.setContentType("text/html");

PrintWriter pw = response.getWriter();

pw.println("<html>");

pw.println("<head><title>Hello World</title></title>");

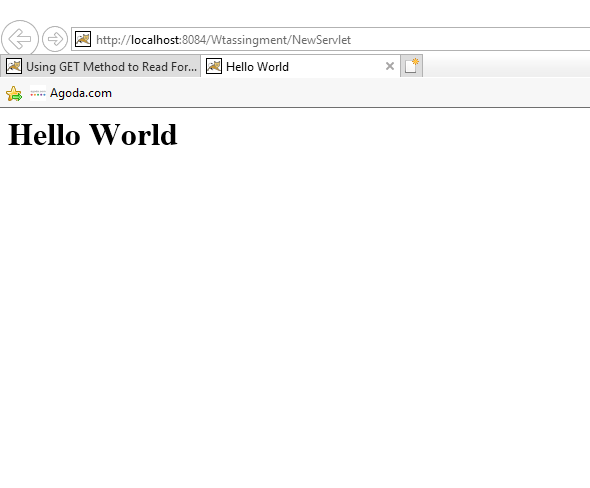
pw.println("<body>");

pw.println("<h1>Hello World</h1>");

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

}

}



7.Write a jsp program to print the first name and last name of user from a html form

NewServlet1.java

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

// Extend HttpServlet class

public class NewServlet1 extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Set response content type

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String title = "Using GET Method to Read Form Data";

String docType =

"<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";

out.println(docType +

"<html>\n" +

"<head><title>" + title + "</title></head>\n" +

"<body bgcolor = \"#f0f0f0\">\n" +

"<h1 align = \"center\">" + title + "</h1>\n" +

"<ul>\n" +

" <li><b>First Name</b>: "

+ request.getParameter("first\_name") + "\n" +

" <li><b>Last Name</b>: "

+ request.getParameter("last\_name") + "\n" +

"</ul>\n" +

"</body>" +

"</html>"

);

}

}

Index.html

<html>

<body>

<form action = "NewServlet1" method = "GET">

First Name: <input type = "text" name = "first\_name">

<br />

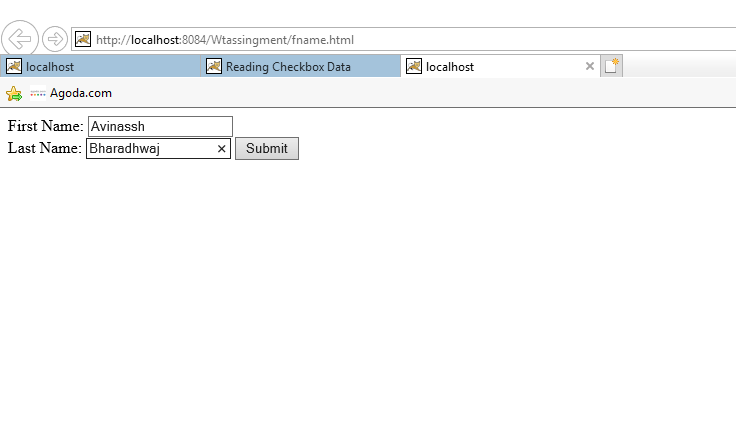
Last Name: <input type = "text" name = "last\_name" />

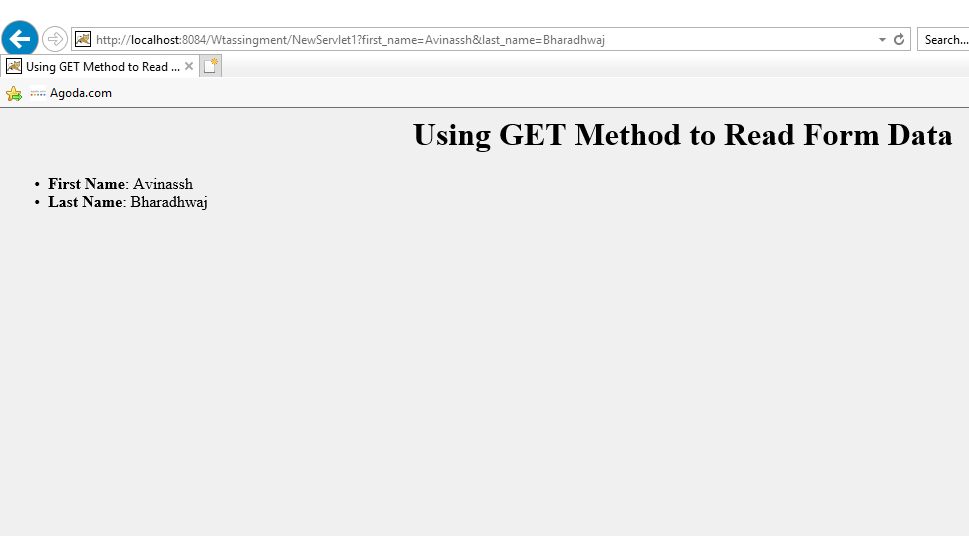
<input type = "submit" value = "Submit" />

</form>

</body>

</html>





8.Write a Servlet program on Reading of checkbox data.

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

// Extend HttpServlet class

public class NewServlet3 extends HttpServlet {

// Method to handle GET method request.

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Set response content type

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String title = "Reading Checkbox Data";

String docType =

"<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";

out.println(docType +

"<html>\n" +

"<head><title>" + title + "</title></head>\n" +

"<body bgcolor = \"#f0f0f0\">\n" +

"<h1 align = \"center\">" + title + "</h1>\n" +

"<ul>\n" +

" <li><b>Maths Flag : </b>: "

+ request.getParameter("maths") + "\n" +

" <li><b>Physics Flag: </b>: "

+ request.getParameter("physics") + "\n" +

" <li><b>Chemistry Flag: </b>: "

+ request.getParameter("chemistry") + "\n" +

"</ul>\n" +

"</body>"

"</html>"

);

}

// Method to handle POST method request.

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doGet(request, response);

}

}

Checkbox.html

<html>

<body>

<form action = "NewServlet3" method = "POST" target = "\_blank">

<input type = "checkbox" name = "maths" checked = "checked" /> Maths

<input type = "checkbox" name = "physics" /> Physics

<input type = "checkbox" name = "chemistry" checked = "checked" />

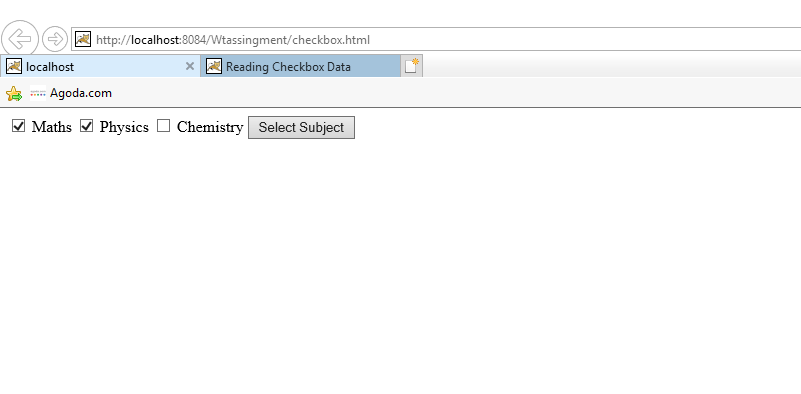
Chemistry

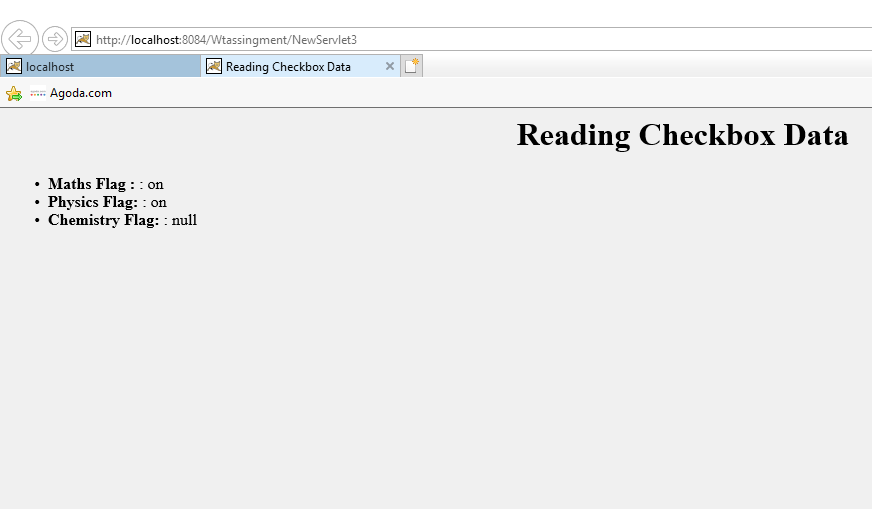
<input type = "submit" value = "Select Subject" />

</form>

</body>

</html>





9.Write a Program for implementing GenericServlet

import java.io.\*;

import javax.servlet.\*;

public class Generic extends GenericServlet{

public void service(ServletRequest req,ServletResponse res)

throws IOException,ServletException{

res.setContentType("text/html");

PrintWriter out=res.getWriter();

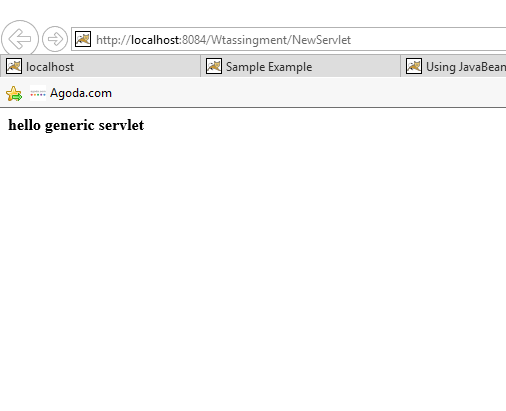
out.print("<html><body>");

out.print("<b>hello generic servlet</b>");

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

}

}



10.Write A Jsp program to display today’s date

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

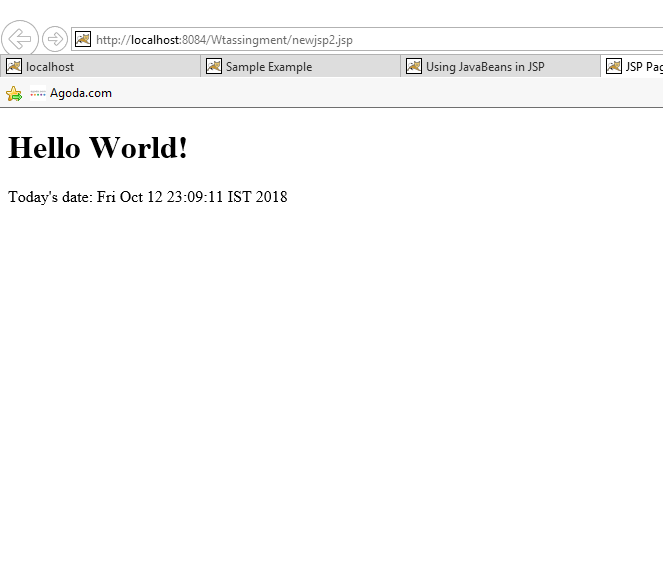
<body>

<h1>Hello World!</h1>

<p>Today's date: <%= (new java.util.Date())%></p>

</body>

</html>



11. Wap to insert values into a table in database using prepared statement

import java.sql.\*;

public class Insert {

    public static void main(String args[])

    {

        try{

            Class.forName("com.mysql.jdbc.Driver");

            Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/webtech1","root","");

            Statement st=con.createStatement();

            PreparedStatement stmt=con.prepareStatement("insert into emp values(?,?,?)");

            stmt.setInt(1,086);

            stmt.setString(2,"Avinassh");

            stmt.setInt(3,16);

            stmt.executeUpdate();

            con.close();

        }

        catch(Exception e)

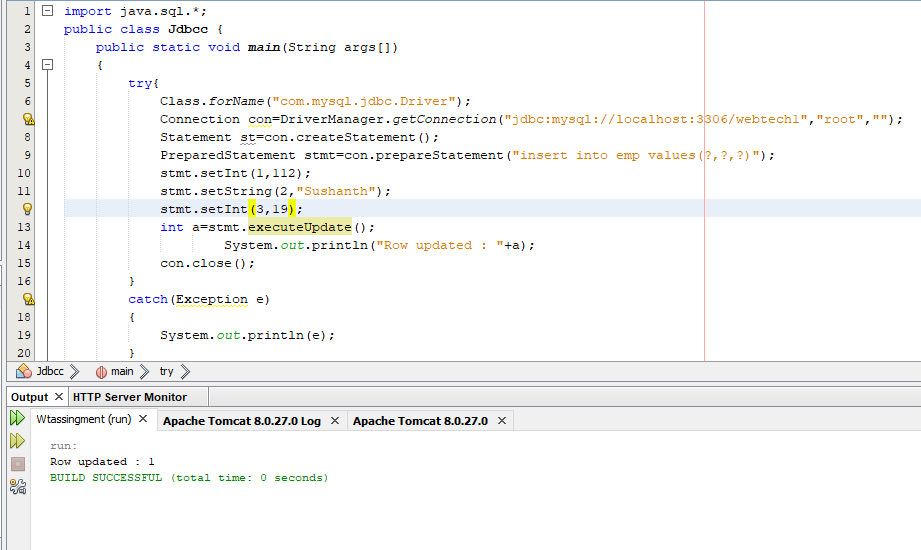
        {

            System.out.println(e);

        }

    }

}



12**) A program on standard actions using JSP**

package action;

public class NewClass {

private String message = "No message specified";

public String getMessage() {

return(message);

}

public void setMessage(String message) {

this.message = message;

}

}

Actions.jsp

<html>

<head>

<title>Using JavaBeans in JSP</title>

</head>

<body>

<center>

<h2>Using JavaBeans in JSP</h2>

<jsp:useBean id = "test" class = "action.NewClass" />

<jsp:setProperty name = "test" property = "message"

value = "Hello JSP..." />

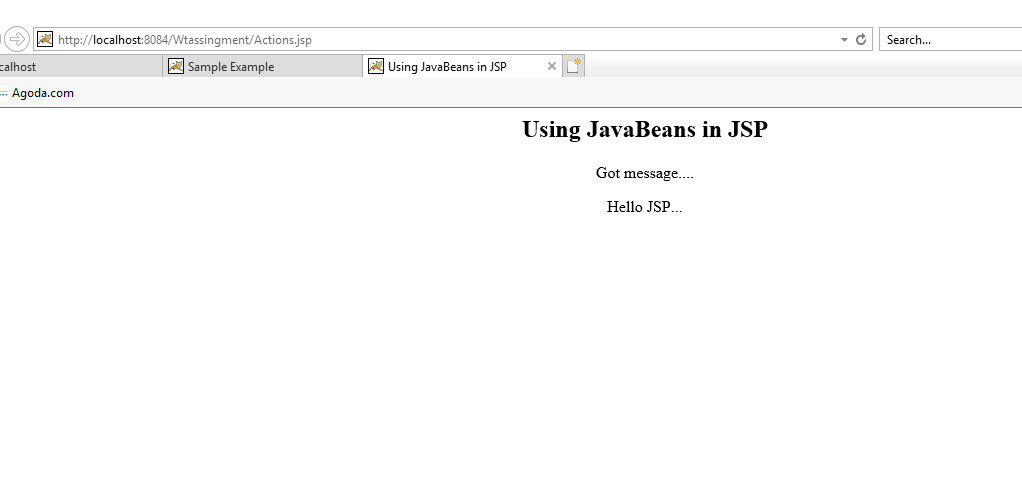
<p>Got message....</p>

<jsp:getProperty name = "test" property = "message" />

</center>

</body>

</html>



13. **Program to print welcome msg using scriptlets**

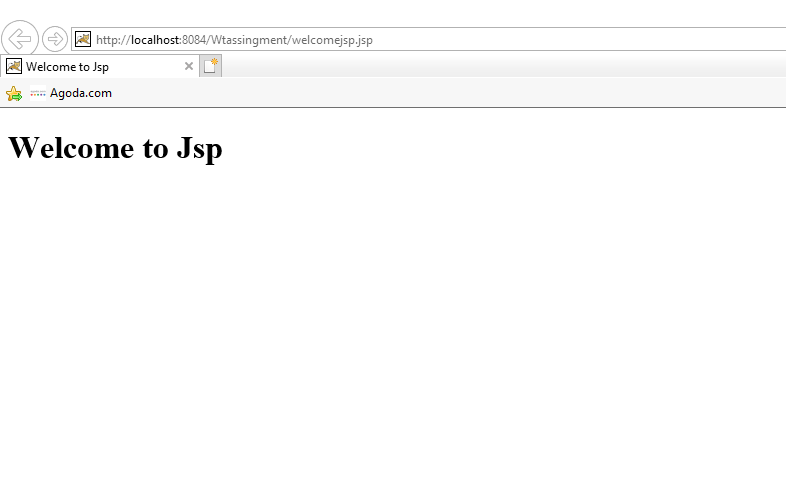
<html>

<body>

<% out.print("welcome to jsp"); %>

</body>

</html>



14. **A JSP program to display name**

Name.html

<html>

<body>

<form action="newjsp3.jsp">

<input type="text" name="uname">

<input type="submit" value="go"><br/>

</form>

</body>

</html>

newjsp3.jsp

<html>

<body>

<%

String name=request.getParameter("uname");

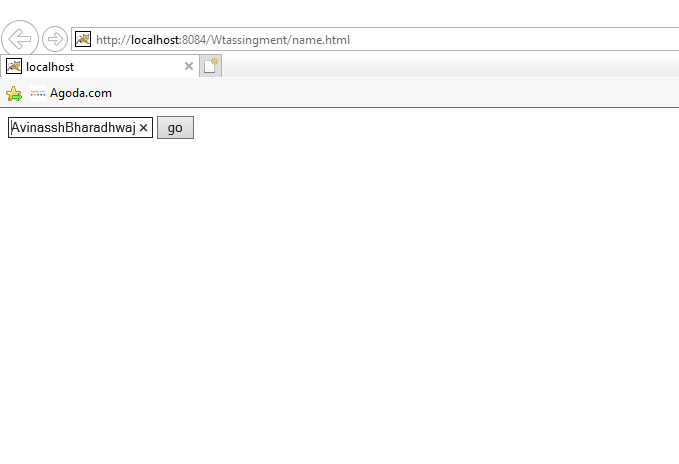
out.print("welcome "+name);

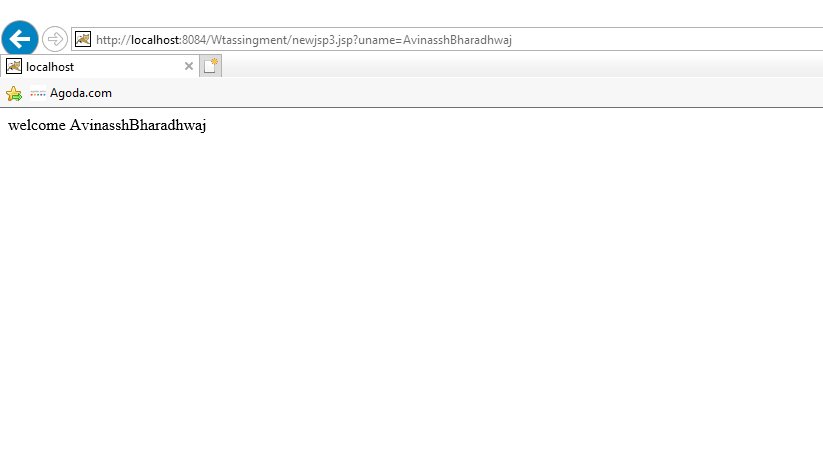
%>

</form>

</body>

</html>





15. **JSP program to perform arithmetic operations**

newjsp7.jsp

<%@ page language="java"%>

<%@ page import="java.lang.\*"%>

<html>

<body>

<H1><center>Result for <%=request.getParameter("a1")%></center></H1>

<%

int i=Integer.parseInt(request.getParameter("t1"));

int j=Integer.parseInt(request.getParameter("t2"));

int k=0;

String str=request.getParameter("a1");

if(str.equals("add"))

k=i+j;

if(str.equals("mul"))

k=i\*j;

if(str.equals("div"))

k=i/j;

%>

Result is <%=k%>

</body>

</html>

Arithematic.html

<html>

<title>Sample Example </title>

<body>

<h1> <center> Example of JSP </center> </h1>

<b> Mathematics</b>

<hr>

<form method="post" action="newjsp7.jsp">

<font size=5 face="Times New Roman">

<input type="radio" name="a1" value="add" checked>Addition</input><br>

<input type="radio" name="a1" value="mul" >Multiplication</input><br>

<input type="radio" name="a1" value="div" >Division</input><br>

</font>

<br><br>

Enter first Value &nbsp &nbsp &nbsp<input type="text" name="t1" value=""><br>

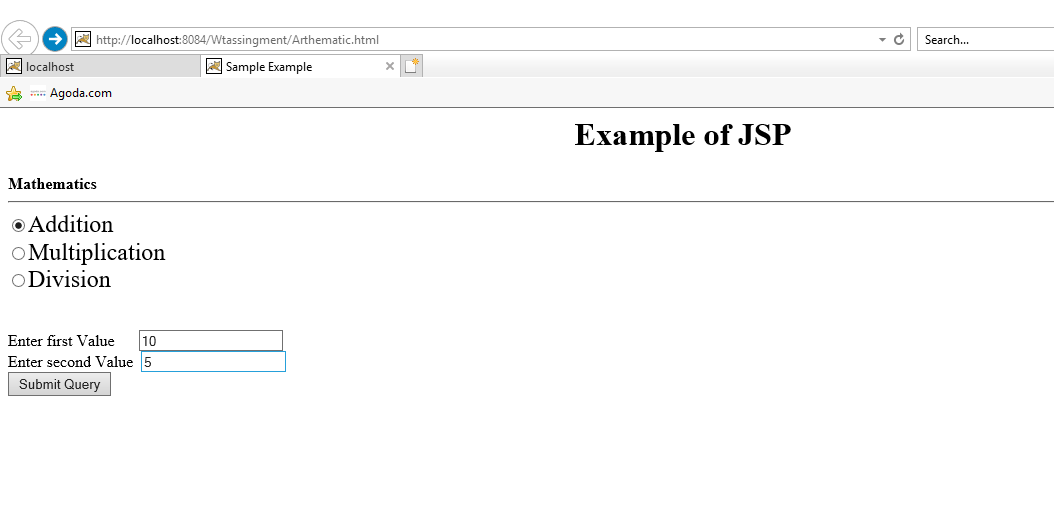
Enter second Value &nbsp<input type="text" name="t2" value=""><br>

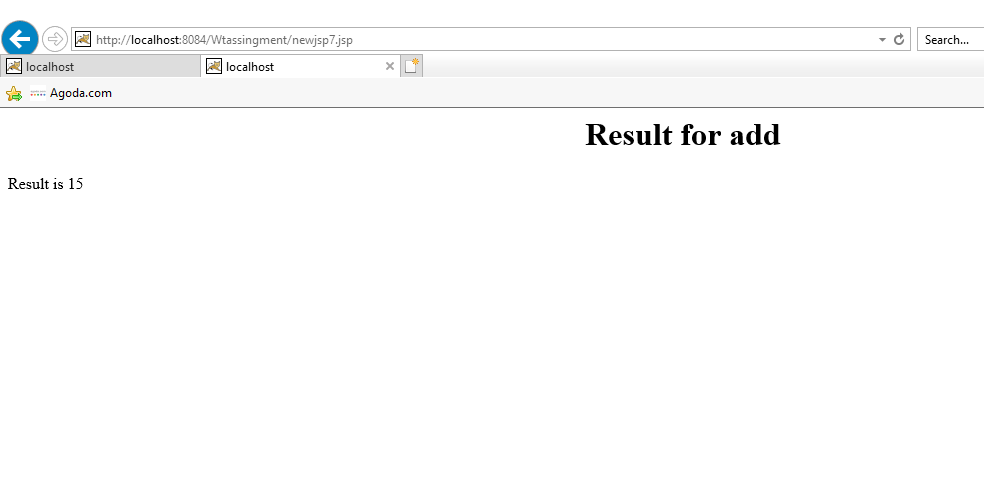
<input type="submit" name="result">

</form>

</body>

</html>





16.WAP to retrive data from a table in database

**import** java.sql.\*;

**class** MysqlCon{

**public** **static** **void** main(String args[]){

**try**{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection(  "jdbc:mysql://localhost:3306/sonoo","root","root");

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from emp");

**while**(rs.next())

System.out.println(rs.getInt(1)+"  "+rs.getString(2)+"  "+rs.getString(3));

con.close();

}**catch**(Exception e){ System.out.println(e);}

}



17.Basic Html page with inline styling

<!doctype html>

<html>

<head>

<title>This is inline styling page</title>

</head>

<body>

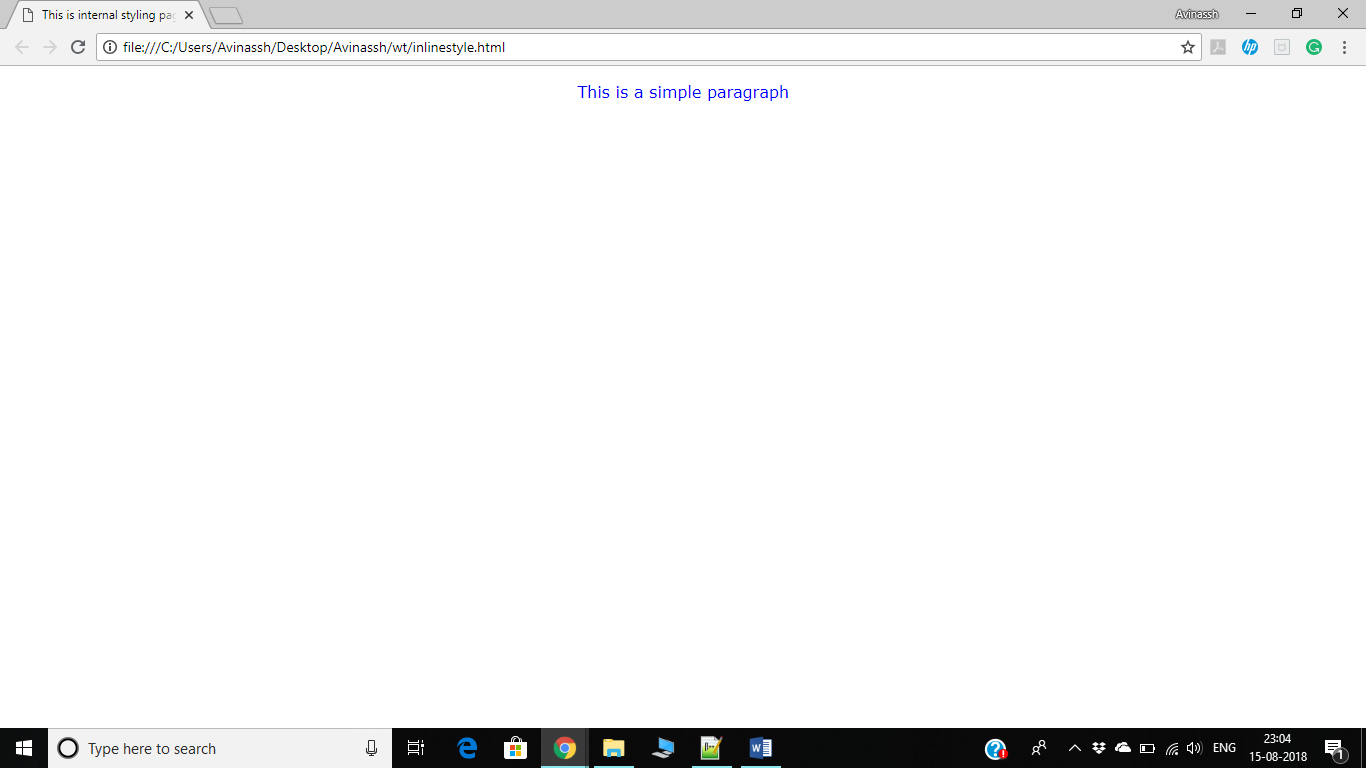
<center>

<p style="color:blue;font-family:verdana;"> This is a simple paragraph </p>

</center>

</body>

</html>



18.HTML page with internal Styling.

<!doctype html>

<html>

<head>

<style>

body {background-color: white;}

h1 {color: blue; font-family:verdana;}

p {color: black; font-family:verdana;}

</style>

<title>This is internal styling page</title>

</head>

<body>

<center>

<h1>Operating system</h1></center>

<p>The dominant desktop operating system is Microsoft Windows with a market share of around 82.74%. macOS by Apple Inc. is in second place (13.23%),

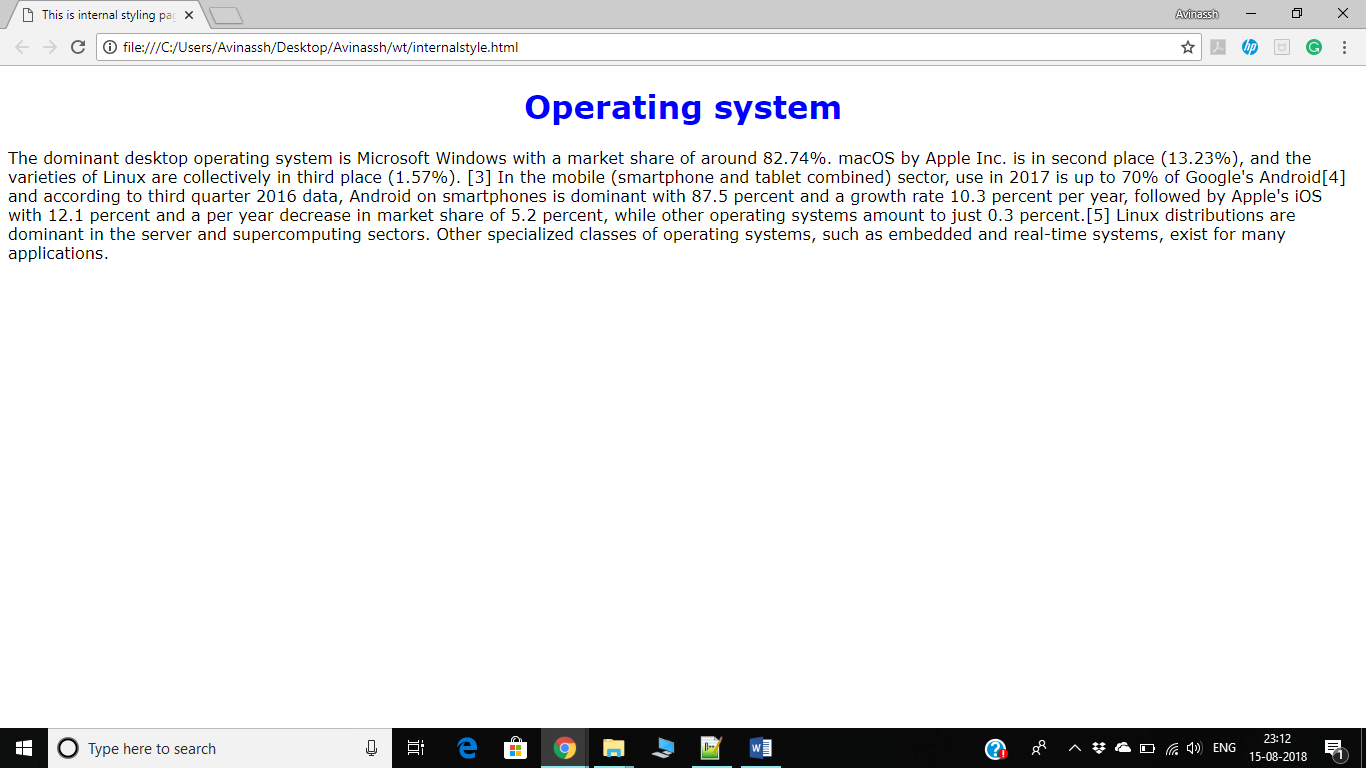
and the varieties of Linux are collectively in third place (1.57%).

[3] In the mobile (smartphone and tablet combined) sector, use in 2017 is up to 70% of Google's Android[4]

</p>

</body>

</html>



19.HTML page with External Styling.

<!doctype html>

<html>

<head>

<link rel="stylesheet" href="exter.css">

<title>This is internal styling page</title>

</head>

<body>

<center>

<h1>Operating system</h1></center>

<p>The dominant desktop operating system is Microsoft Windows with a market share of around 82.74%. macOS by Apple Inc.

is in second place (13.23%), and the varieties of Linux are collectively in third place (1.57%).

[3] In the mobile (smartphone and tablet combined) sector, use in 2017 is up to 70% of Google's Android[4]

and according to third quarter 2016 data,

</p>

<d>The dominant desktop operating system is Microsoft Windows with a market share of around 82.74%. macOS by Apple Inc.

is in second place (13.23%), and the varieties of Linux are collectively in third place (1.57%).

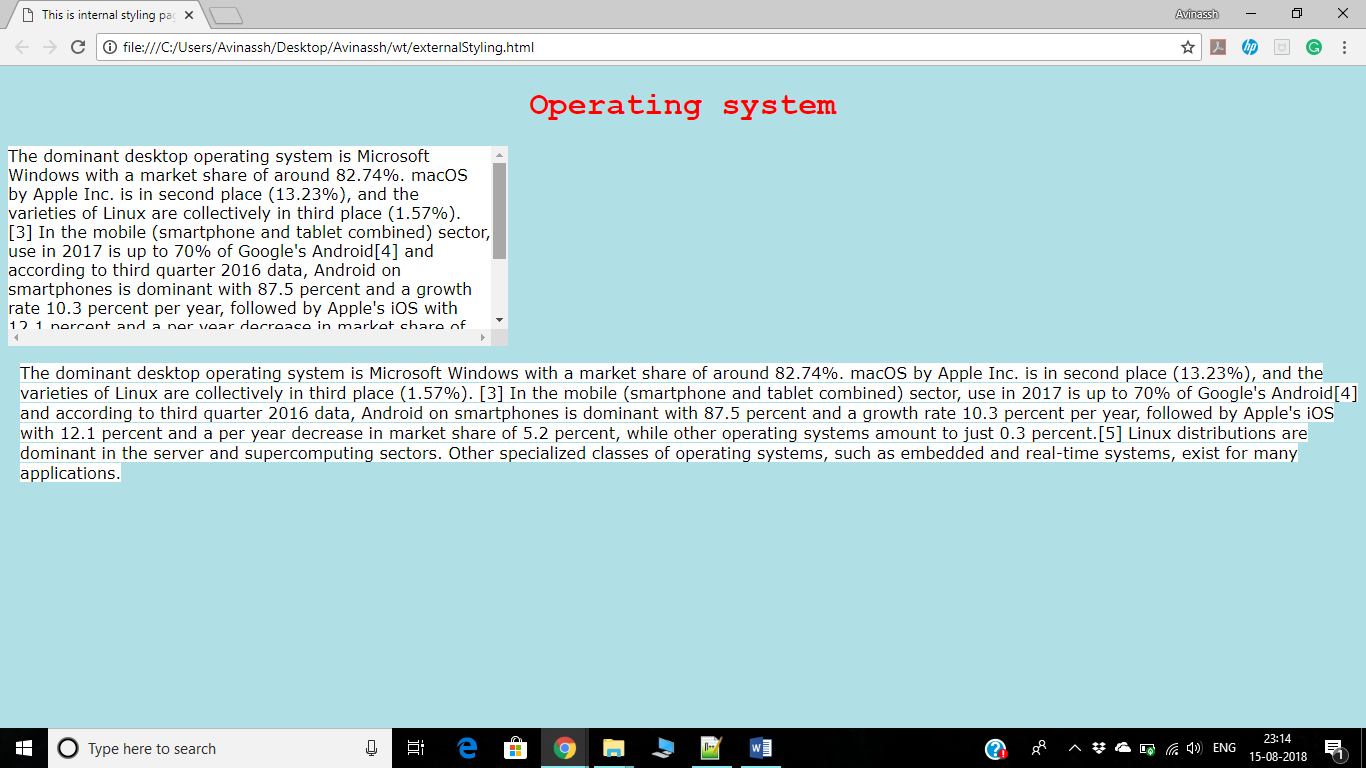
[3] In the mobile (smartphone and tablet combined) sector, use in 2017 is up to 70% of Google's Android[4]

and according to third quarter 2016 data,

</d>

</body>

</html>



External CSS file

body{

background-color :powderblue;

}

h1 {

color: red;

font-family:Courier New;

position :static;

}

p {

color : black;

font-family:verdana;

position :static;

width:500px;

height:200px;

background:white;

left:500px;

overflow:scroll;

}

d {

color : black;

font-family:verdana;

position :sticky;

width:500px;

height:200px;

background:white;

left:500px;

top:500px;

overflow:scroll;

}

20.HTML Page with Box model

<!DOCTYPE html>

<html>

<head>

<style>

div {

width:320px;

padding:10px;

border:5px solid red;

}

</style>

</head>

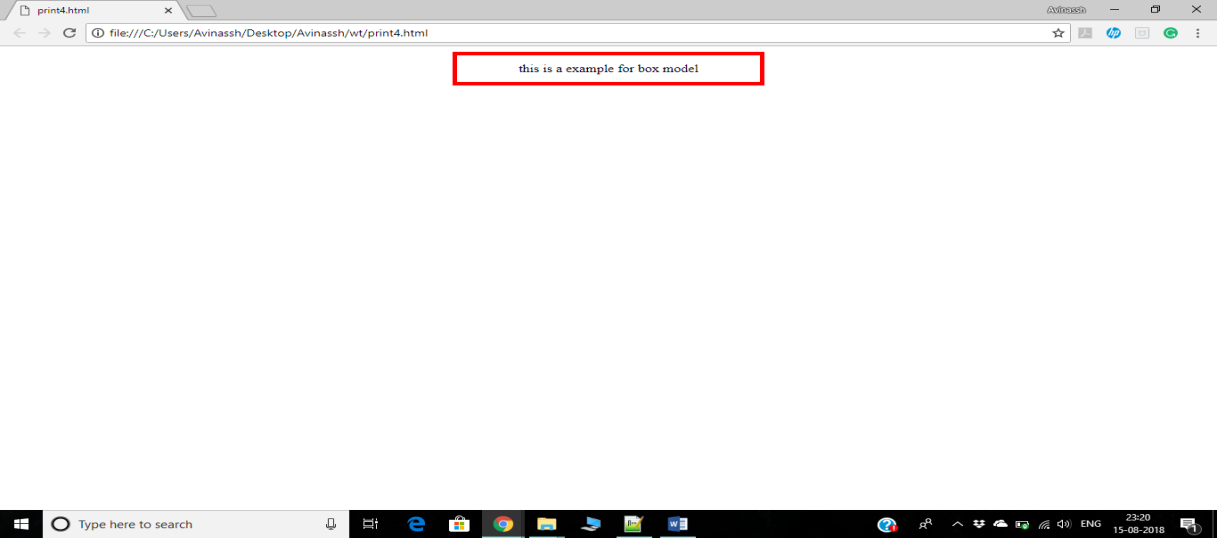
<body>

<center>

<div>this is a example for box model</div>

</body>

</html>



21.HTML page with Basic table.

<!DOCTYPE html>

<html>

<head>

<style>

table,th,tr,td{border:1px solid black;border-collapse:collapse}

</style>

</head>

<body>

<h2>Basic HTML Table</h2>

<table style="width:100%">

<tr>

<th>cars</th>

<th>bikes</th>

<th>bus</th>

</tr>

<tr>

<td>maruthi</td>

<td>bajaj</td>

<td>mahindra</td>

</tr>

<tr>

<td>hyundai</td>

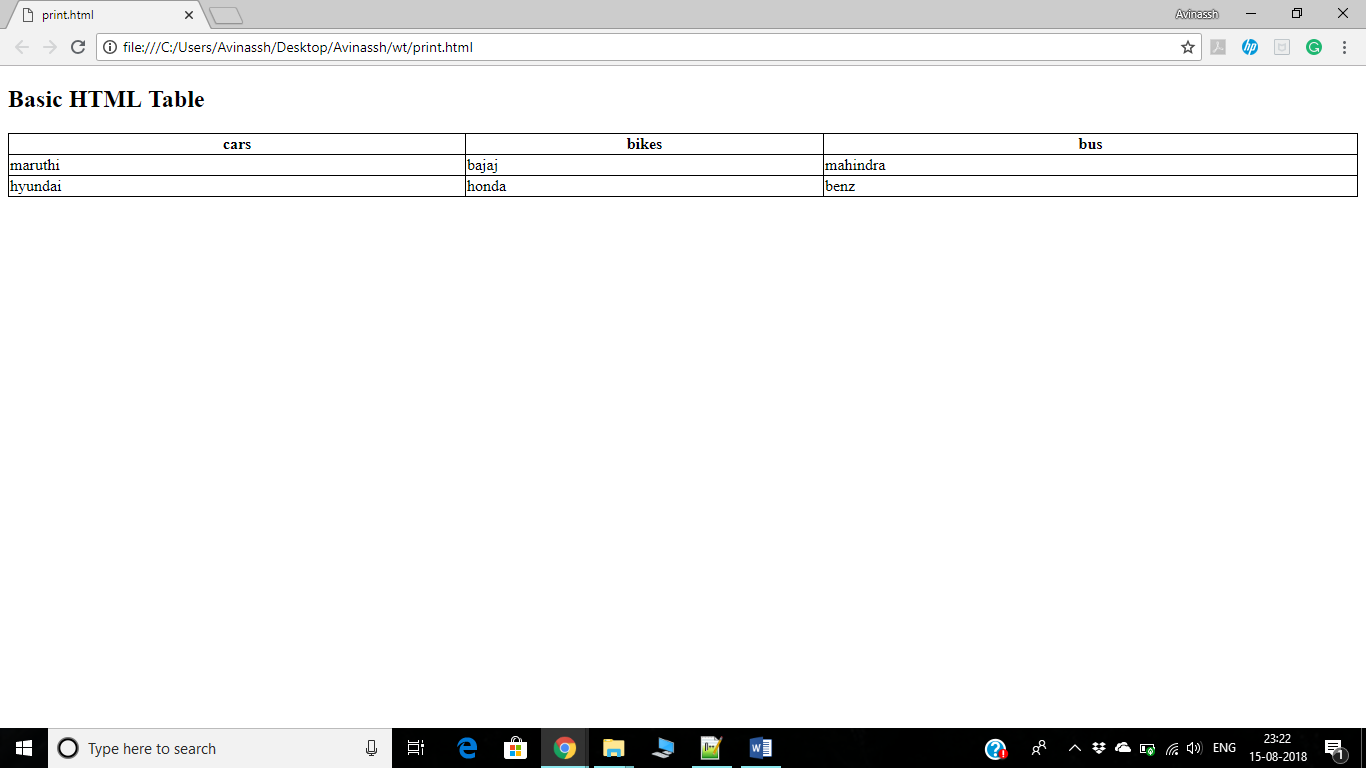
<td>honda</td>

<td>benz</td>

</tr>

</table>

</body>



22.JavaScript page

<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" href="exter.css">

<script type="text/javascript">

var x=1;

function message(msg){

if(x==1)

{

x=0;

document.getElementById('demo').innerHTML=msg;

}

else

{

x=1;

document.getElementById('demo').innerHTML='Before click';

}

}

</script>

</head>

<body>

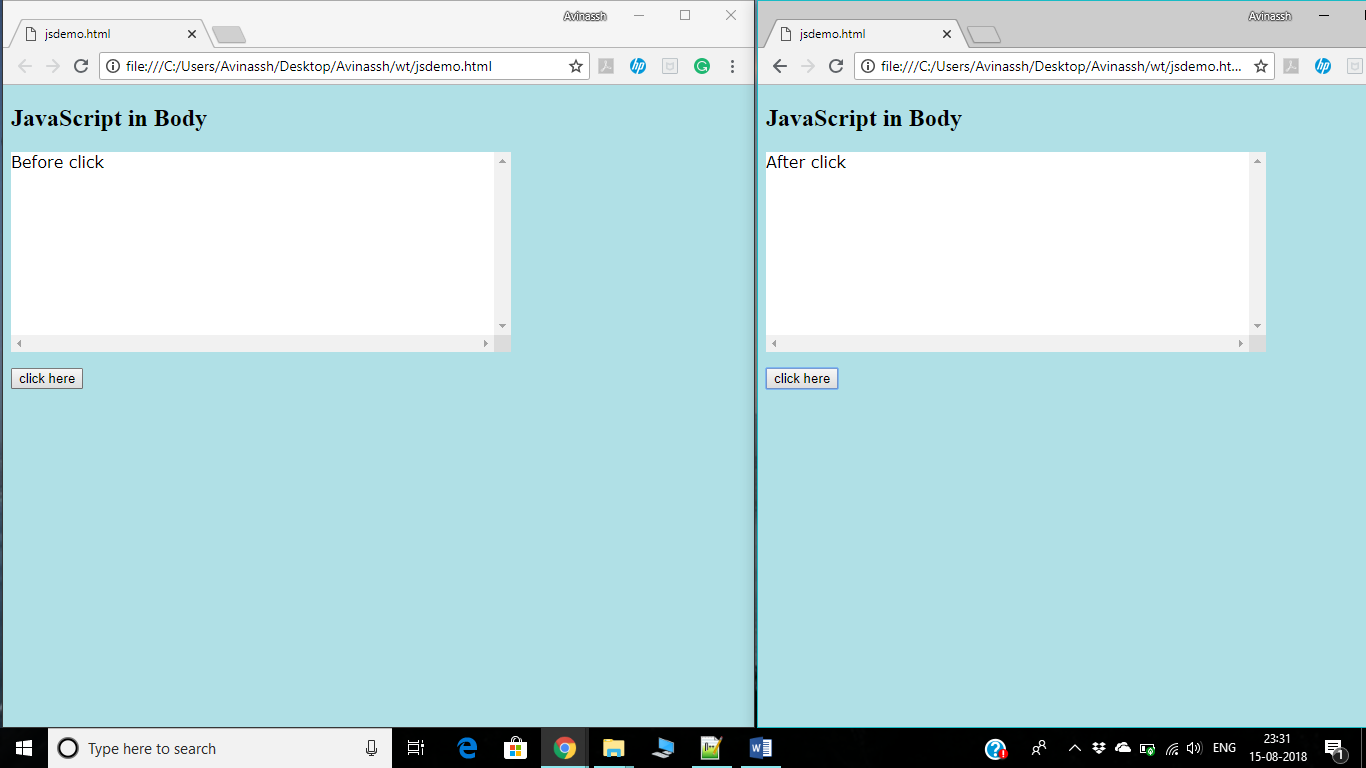
<h2>JavaScript in Body</h2>

<p id="demo" >Before click</p>

<input type=button onclick="message('After click')" value='click here'>

</body>

</html>



23.JavaScript page using Date() function

<!DOCTYPE html>

<html>

<head>

<script>

function myFunction() {

var sa=new Date();

alert(sa);

}

</script>

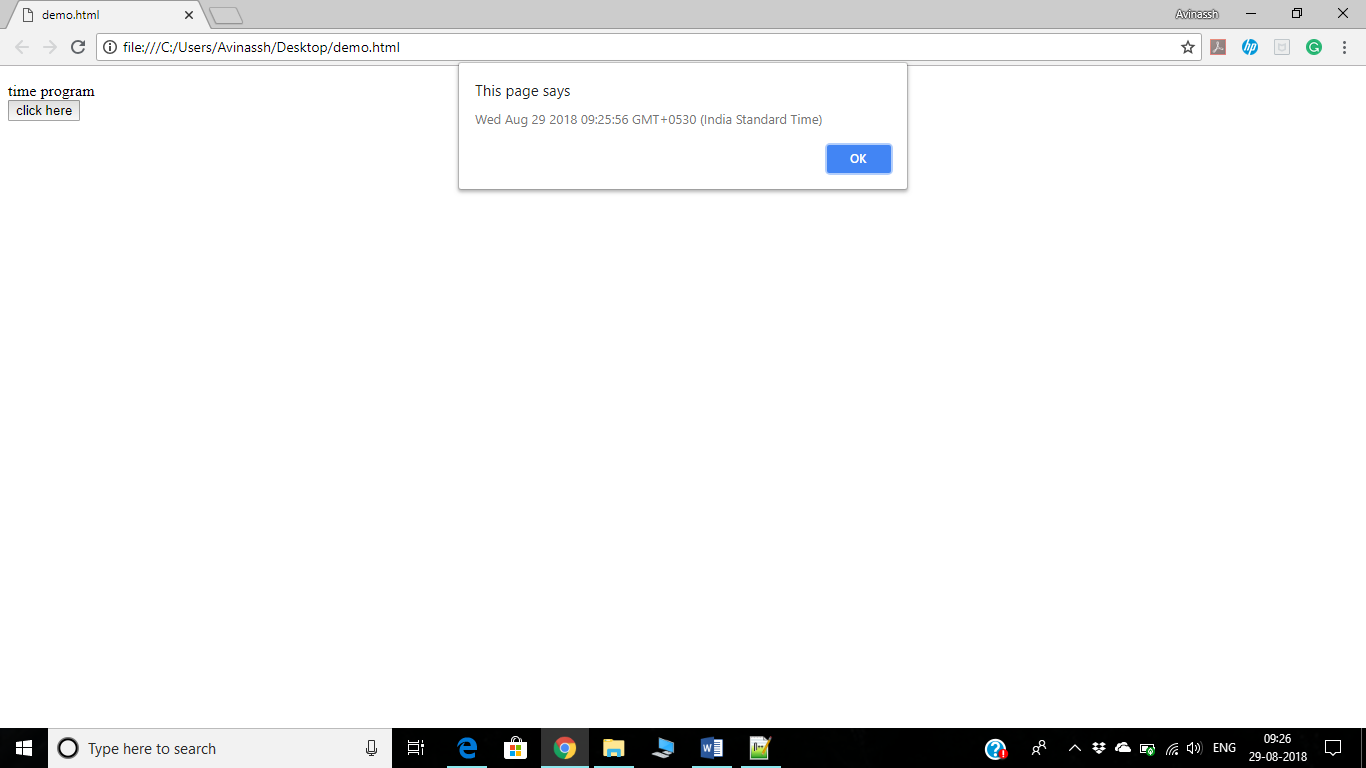
<body>

<p id="demo">time program </h2><br>

<button onclick="myFunction()">click here</button>

</body>

</html>



24.Jquery Program to hide content of html.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("#demo").hide();

});

});

</script>

</head>

<body>

<h2>jquery program to hide content of html which belongs to particular id</h2>

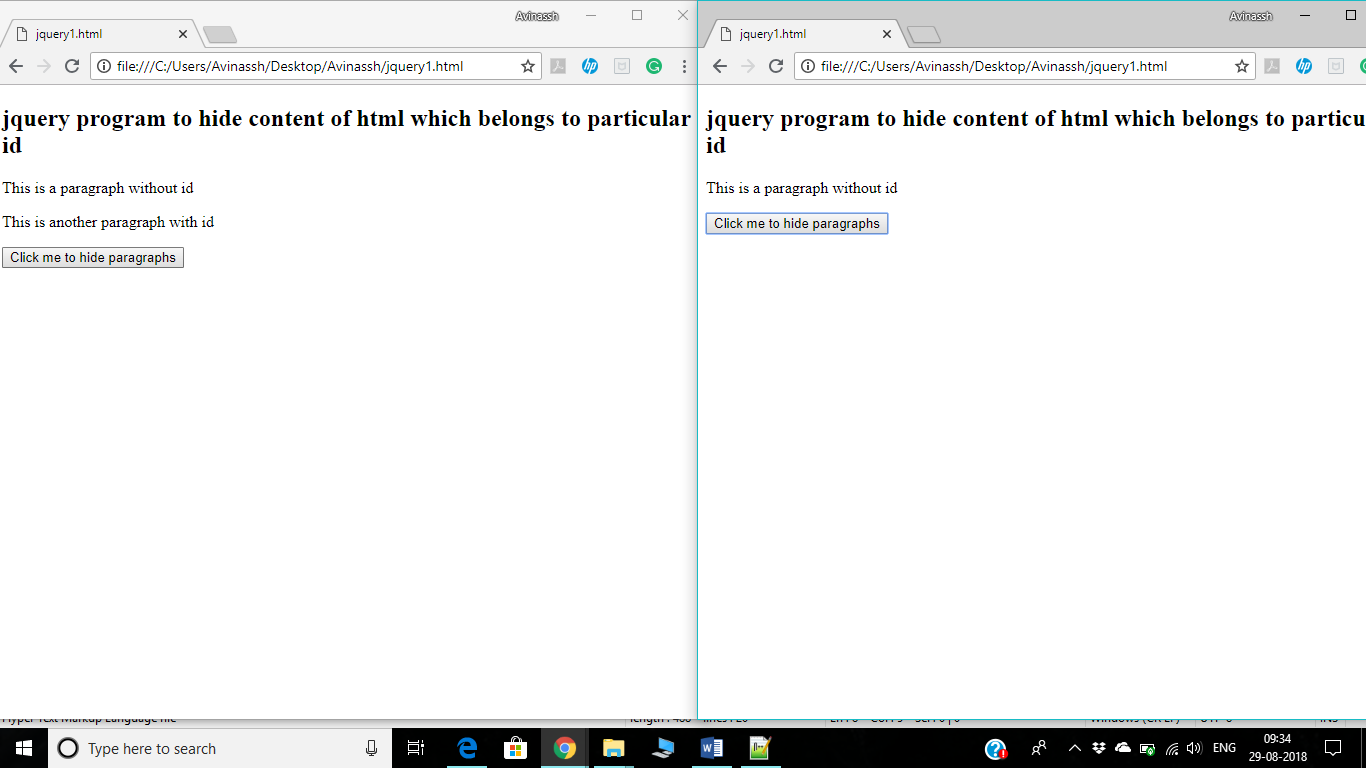
<p>This is a paragraph without id</p>

<p id='demo'>This is another paragraph with id</p>

<button>Click me to hide paragraphs</button>

</body>

</html>



25.Jquery Form Validation

|  |  |
| --- | --- |
|  | <html lang="en"> |
|  | <head> |
|  | <script src="<https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js>"></script> |
|  | </head> |
|  |  |
|  | <!-- Styles --> |
|  | <style> |
|  | #contact label{ |
|  | display: inline-block; |
|  | width: 100px; |
|  | text-align: right; |
|  | } |
|  | #contact\_submit{ |
|  | padding-left: 100px; |
|  | } |
|  | #contact div{ |
|  | margin-top: 1em; |
|  | } |
|  | textarea{ |
|  | vertical-align: top; |
|  | height: 5em; |
|  | } |
|  |  |
|  | .error{ |
|  | display: none; |
|  | margin-left: 10px; |
|  | } |
|  |  |
|  | .error\_show{ |
|  | color: red; |
|  | margin-left: 10px; |
|  | } |
|  |  |
|  | input.invalid, textarea.invalid{ |
|  | border: 2px solid red; |
|  | } |
|  |  |
|  | input.valid, textarea.valid{ |
|  | border: 2px solid green; |
|  | } |
|  | </style> |
|  |  |
|  | <!-- JavaScript Code --> |
|  | <script> |
|  | $(document).ready(function() { |
|  | <!-- Real-time Validation --> |
|  | <!--Name can't be blank--> |
|  | $('#contact\_name').on('input', function() { |
|  | var input=$(this); |
|  | var is\_name=input.val(); |
|  | if(is\_name){input.removeClass("invalid").addClass("valid");} |
|  | else{input.removeClass("valid").addClass("invalid");} |
|  | }); |
|  |  |
|  | <!--Email must be an email --> |
|  | $('#contact\_email').on('input', function() { |
|  | var input=$(this); |
|  | var re = /^[a-zA-Z0-9.!#$%&'\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/; |
|  | var is\_email=re.test(input.val()); |
|  | if(is\_email){input.removeClass("invalid").addClass("valid");} |
|  | else{input.removeClass("valid").addClass("invalid");} |
|  | }); |
|  |  |
|  | <!--Website must be a website --> |
|  | $('#contact\_website').on('input', function() { |
|  | var input=$(this); |
|  | if (input.val().substring(0,4)=='www.'){input.val('http://www.'+input.val().substring(4));} |
|  | var re = /(http|ftp|https):\/\/[\w-]+(\.[\w-]+)+([\w.,@?^=%&amp;:\/~+#-]\*[\w@?^=%&amp;\/~+#-])?/; |
|  | var is\_url=re.test(input.val()); |
|  | if(is\_url){input.removeClass("invalid").addClass("valid");} |
|  | else{input.removeClass("valid").addClass("invalid");} |
|  | }); |
|  |  |
|  | <!--Message can't be blank --> |
|  | $('#contact\_message').keyup(function(event) { |
|  | var input=$(this); |
|  | var message=$(this).val(); |
|  | console.log(message); |
|  | if(message){input.removeClass("invalid").addClass("valid");} |
|  | else{input.removeClass("valid").addClass("invalid");} |
|  | }); |
|  |  |
|  | <!-- After Form Submitted Validation--> |
|  | $("#contact\_submit button").click(function(event){ |
|  | var form\_data=$("#contact").serializeArray(); |
|  | var error\_free=true; |
|  | for (var input in form\_data){ |
|  | var element=$("#contact\_"+form\_data[input]['name']); |
|  | var valid=element.hasClass("valid"); |
|  | var error\_element=$("span", element.parent()); |
|  | if (!valid){error\_element.removeClass("error").addClass("error\_show"); error\_free=false;} |
|  | else{error\_element.removeClass("error\_show").addClass("error");} |
|  | } |
|  | if (!error\_free){ |
|  | event.preventDefault(); |
|  | } |
|  | else{ |
|  | alert('No errors: Form will be submitted'); |
|  | } |
|  | }); |
|  |  |
|  |  |
|  |  |
|  | }); |
|  | </script> |
|  |  |
|  | <body> |
|  | <form id="contact" method="post" action=""> |
|  | <!-- Name --> |
|  | <div> |
|  | <label for="contact\_name">Name:</label> |
|  | <input type="text" id="contact\_name" name="name"></input> |
|  | <span class="error">This field is required</span> |
|  | </div> |
|  | <!-- Email --> |
|  | <div> |
|  | <label for="contact\_email">Email:</label> |
|  | <input type="email" id="contact\_email" name="email"></input> |
|  | <span class="error">A valid email address is required</span> |
|  | </div> |
|  | <!--Website --> |
|  | <div> |
|  | <label for="contact\_website">Website:</label> |
|  | <input type="url" id="contact\_website" name="website"></input> |
|  | <span class="error">A valid url is required</span> |
|  | </div> |
|  | <!-- Message --> |
|  | <div> |
|  | <label for="contact\_message">Message:</label> |
|  | <textarea id="contact\_message" name="message"></textarea> |
|  | <span class="error">This field is required</span> |
|  | </div> |
|  | <!-- Submit Button --> |
|  | <div id="contact\_submit"> |
|  | <button type="submit">Submit</button> |
|  | </div> |
|  | </form> |
|  | </body> |
|  | </html> |
|  |  |

26.Basic Xml program

<?xml version="1.0" encoding="UTF-8"?>

<note>

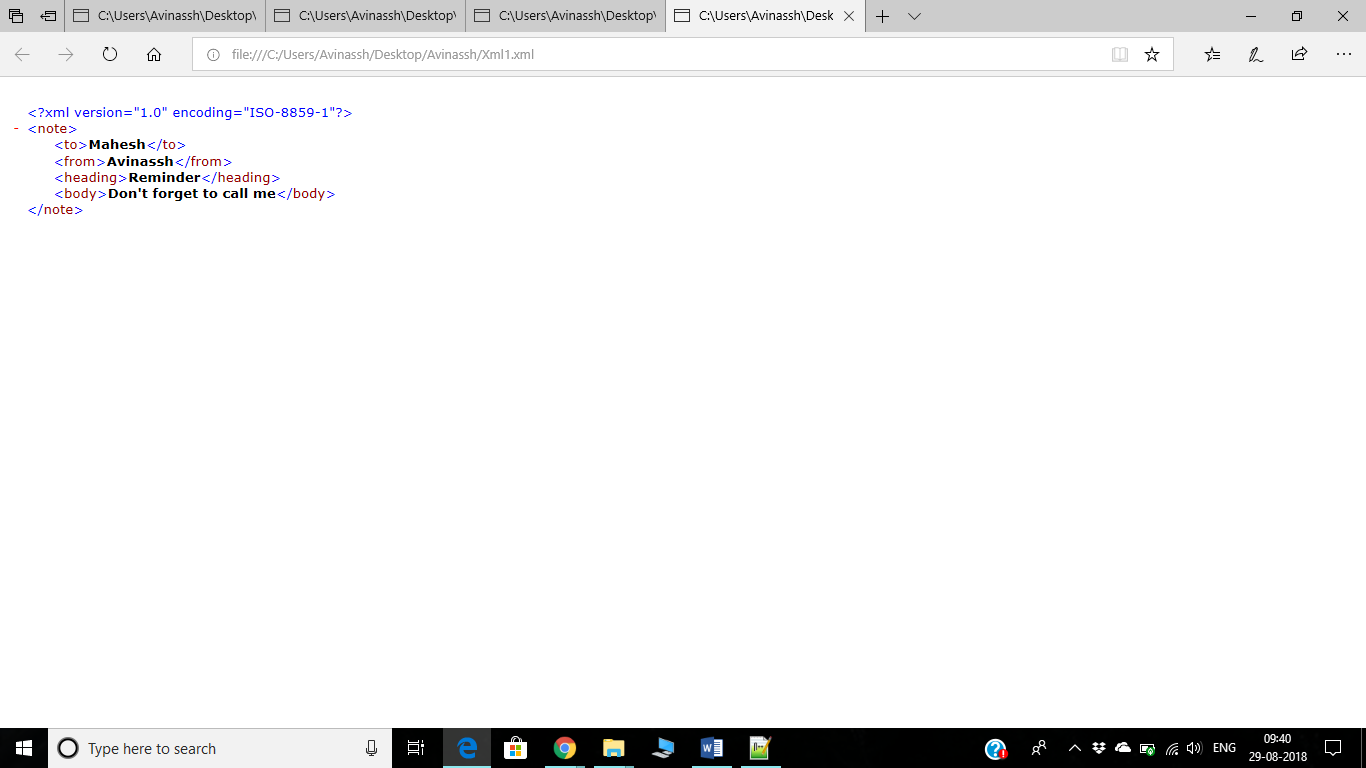
<to>Mahesh</to>

<from>Avinassh</from>

<heading>Reminder</heading>

<body>Don't forget to call me</body>

</note>



27.Internal DTD xml program for library management.

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE Library[

<!ELEMENT Library (fac,stud,book) >

<!ELEMENT fac (id,name,email)>

<!ELEMENT id (#PCDATA) >

<!ELEMENT name (#PCDATA) >

<!ELEMENT email (#PCDATA)>

<!ELEMENT stud (sid,sname,semail)>

<!ELEMENT sid (#PCDATA) >

<!ELEMENT sname (#PCDATA) >

<!ELEMENT semail (#PCDATA)>

<!ELEMENT book (title,author,publication)>

<!ELEMENT title (#PCDATA) >

<!ELEMENT author (#PCDATA) >

<!ELEMENT publication (#PCDATA)>]>

<Library>

<fac>

<ids>16016</ids>

<name>Harsha</name>

<email>harsha@cbit.org.in</email>

</fac>

<stud>

<sid>1601</sid>

<sname>Mahesh</sname>

<semail>mahesh.student@cbit.org.in</semail>

</stud>

<book>

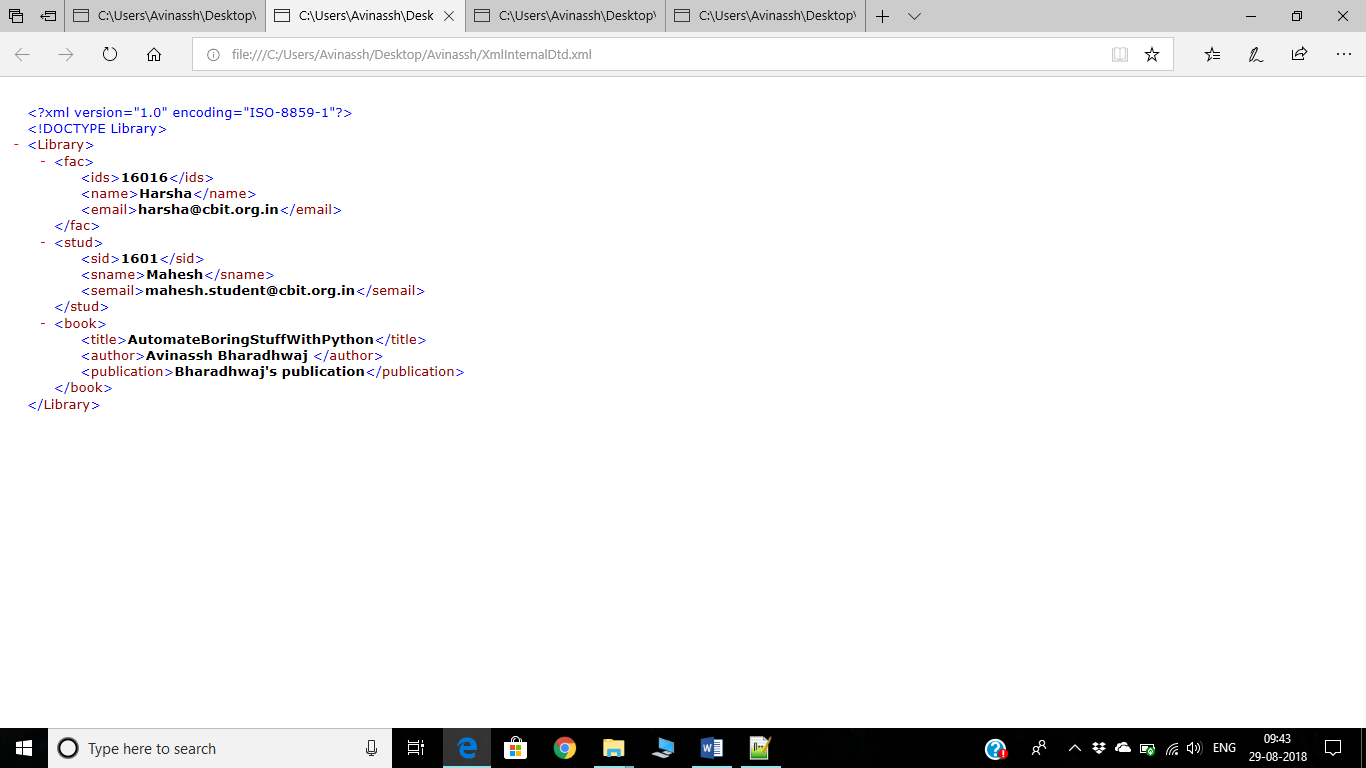
<title>AutomateBoringStuffWithPython</title>

<author>Avinassh Bharadhwaj </author>

<publication>Bharadhwaj's publication</publication>

</book>

</Library>



28. External DTD xml program for library management.

DTD file

<!ELEMENT Library (fac,stud,book) >

<!ELEMENT fac (id,name,email)>

<!ELEMENT id (#PCDATA) >

<!ELEMENT name (#PCDATA) >

<!ELEMENT email (#PCDATA)>

<!ELEMENT stud (sid,sname,semail)>

<!ELEMENT sid (#PCDATA) >

<!ELEMENT sname (#PCDATA) >

<!ELEMENT semail (#PCDATA)>

<!ELEMENT book (title,author,publication)>

<!ELEMENT title (#PCDATA) >

<!ELEMENT author (#PCDATA) >

<!ELEMENT publication (#PCDATA)>

Xml file

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE Library SYSTEM "C:\Users\ITLAB3\Desktop\it-86\extrenaldtd.dtd">

<Library>

<fac>

<id>16016</id>

<name>Harsha</name>

<email>harsha@cbit.org.in</email>

</fac>

<stud>

<sid>1601</sid>

<sname>Mahesh</sname>

<semail>mahesh.student@cbit.org.in</semail>

</stud>

<book>

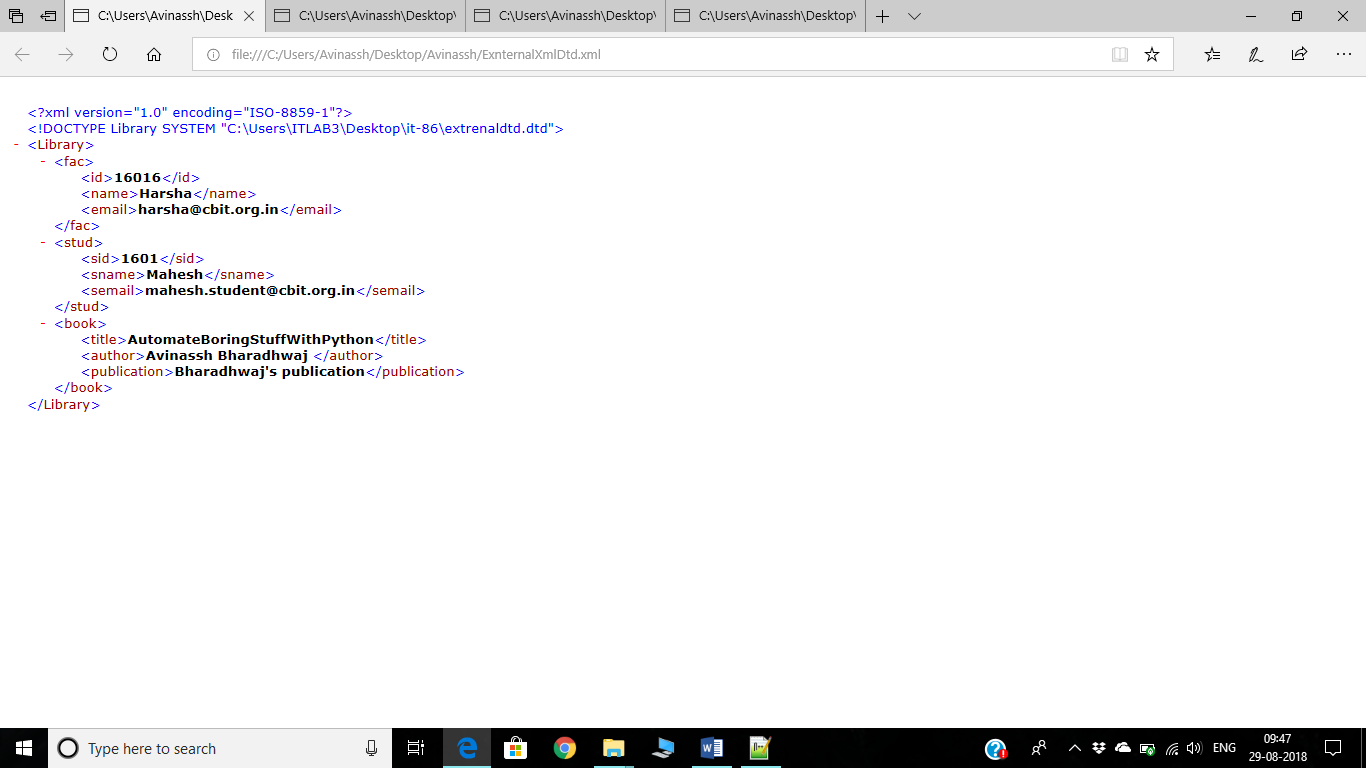
<title>AutomateBoringStuffWithPython</title>

<author>Avinassh Bharadhwaj </author>

<publication>Bharadhwaj's publication</publication>

</book>

</Library>



29.Converting XML To HTML USING XSLT Process

XML:

<?xml version="1.0" encoding="UTF-8"?>

<!-- cd.xml -->

<?xml-stylesheet type = "text/xsl" href="cd.xsl" ?>

<catalog>

<cd>

<title>Empire Burlesque</title>

<artist>Bob Dylan</artist>

<country>USA</country>

<company>Columbia</company>

<price>10.90</price>

<year>1985</year>

</cd>

<cd>

<title>Hide your heart</title>

<artist>Bonnie Tyler</artist>

<country>UK</country>

<company>CBS Records</company>

<price>9.90</price>

<year>1988</year>

</cd>

<cd>

<title>Greatest Hits</title>

<artist>Dolly Parton</artist>

<country>USA</country>

<company>RCA</company>

<price>9.90</price>

<year>1982</year>

</cd>

</catalog>

XSL:

<?xml version="1.0" encoding="UTF-8"?>

<!-- cd.xsl -->

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">

<html>

<body>

<h2>My CD Collection</h2>

<table border="1">

<tr bgcolor="#9acd32">

<th style="text-align:left">Title</th>

<th style="text-align:left">Artist</th>

</tr>

<xsl:for-each select="catalog/cd">

<tr>

<td><xsl:value-of select="title"/></td>

<td><xsl:value-of select="artist"/></td>

</tr>

</xsl:for-each>

</table>

</body>

</html>

</xsl:template>

</xsl:stylesheet>



30.XML Styling Using CSS

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/css" href= "D:\san1\Rule.css"?>

<books>

<heading>Welcome To GeeksforGeeks </heading>

<book>

<title>Title -: Web Programming</title>

<author>Author -: Chrisbates</author>

<publisher>Publisher -: Wiley</publisher>

<edition>Edition -: 3</edition>

<price> Price -: 300</price>

</book>

<book>

<title>Title -: Internet world-wide-web</title>

<author>Author -: Ditel</author>

<publisher>Publisher -: Pearson</publisher>

<edition>Edition -: 3</edition>

<price>Price -: 400</price>

</book>

<book>

<title>Title -: Computer Networks</title>

<author>Author -: Foruouzan</author>

<publisher>Publisher -: Mc Graw Hill</publisher>

<edition>Edition -: 5</edition>

<price>Price -: 700</price>

</book>

</books>

**Rule.css**

.books {

color: white;

background-color : gray;

width: 100%;

}

.headingg {

color: green;

font-size : 40px;

background-color : powderblue;

}

.headingg, title, author, publisher, edition, price {

display : block;

}

.title {

font-size : 25px;

font-weight : bold;

}

