Mayur Gorane
48 TE (A)

Assignment 1

Problem Statement: Installation of TOMCAT Server & configuration of it

Theory - Concept -

Web application - It runs over the internet.

Web application contains 5 components

1. HTTP Server - eg. Google web server, Apache TOMCAT Apache HTTP server, etc.
2. HTTP client (web browser) - Firefox, google chrome, etc
3. Database - MySQL, SQLite, oracle, etc.
4. Client side programs - It can be written in HTML FORM Javascript, etc.
5. Server Side programs - Could be written in Javascript Servlet JSP, PHP, Python, etc.

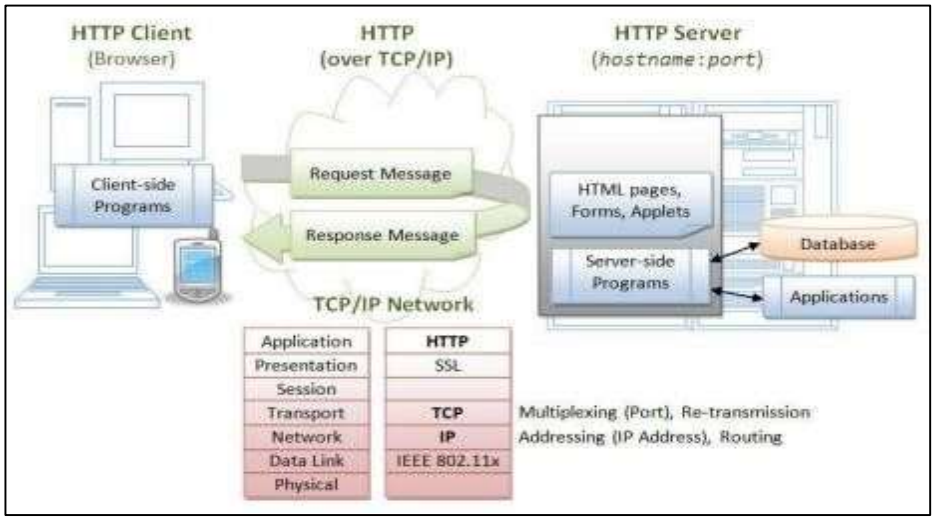
Apache TOMCAT - It is an open source project, under the Apache software foundation.

Execution steps -

1. Goto http://tomcat.apache.org
2. Select tomcat version
3. Download & run .exe file
4. Use default setting provide a password you will remember.

How to run TOMCAT -

1. Find the start program in programs menu look under Apache tomcat & select 'start TOMCAT'


HTTP Client
(Browser)
HTTP
(over TCP/IP)
HTTP Server
(hostname:port)
Client-side Programs
Request Message
Response Message
HTML pages, Forms, Applets
Server-side Programs
Database
Applications
TCP/IP Network
Application
Presentation
Session
Transport
Network
Data Link
Physical
HTTP
SSL
TCP
IP
IEEE 802.11x
Multiplexing (Port), Re-transmission
Addressing (IP Address), Routing

2. Open any webbrowser & type given 'URL' http://localhost:8080/ Now you will see TOMCAT HOMEPAGE.

Q. What is servlet container life cycle?

→ A servlet lifecycle consist of series of events which defines how the servlet is loaded & instantiated, initialized, how it handles request from client & how is taken of service.

Q. what service are provided by TOMCAT?

→ 1. Servlet life cycle.
2. Handle Web Requests
3. Database connection pooling
4. Clustering
5. High availability

Q. Explain directory structure of TOMCAT?

→ 1. bin directory - Contains binaries & scripts
2. conf - It contains systemwide configuration files, such as server.xml, web.xml.
3. webapps - contains webapps to be deployed
4. lib - Contains TOMCAT's system wide library JAR files accessible by all webapps.
5. logs - contains TOMCAT's log files
6. work - TOMCAT's working directory used by JSP for JSP to servlet conversion

Conclusion - Hence we have learned how to install & configure TOMCAT's server.

PAGE NO. 3
DATE / /

Mayur Gorane
48 TE (A)

Assignment 2

Title - HTML, CSS, XML

Problem Statement - Write a program to design registration form for students by using HTML & CSS.

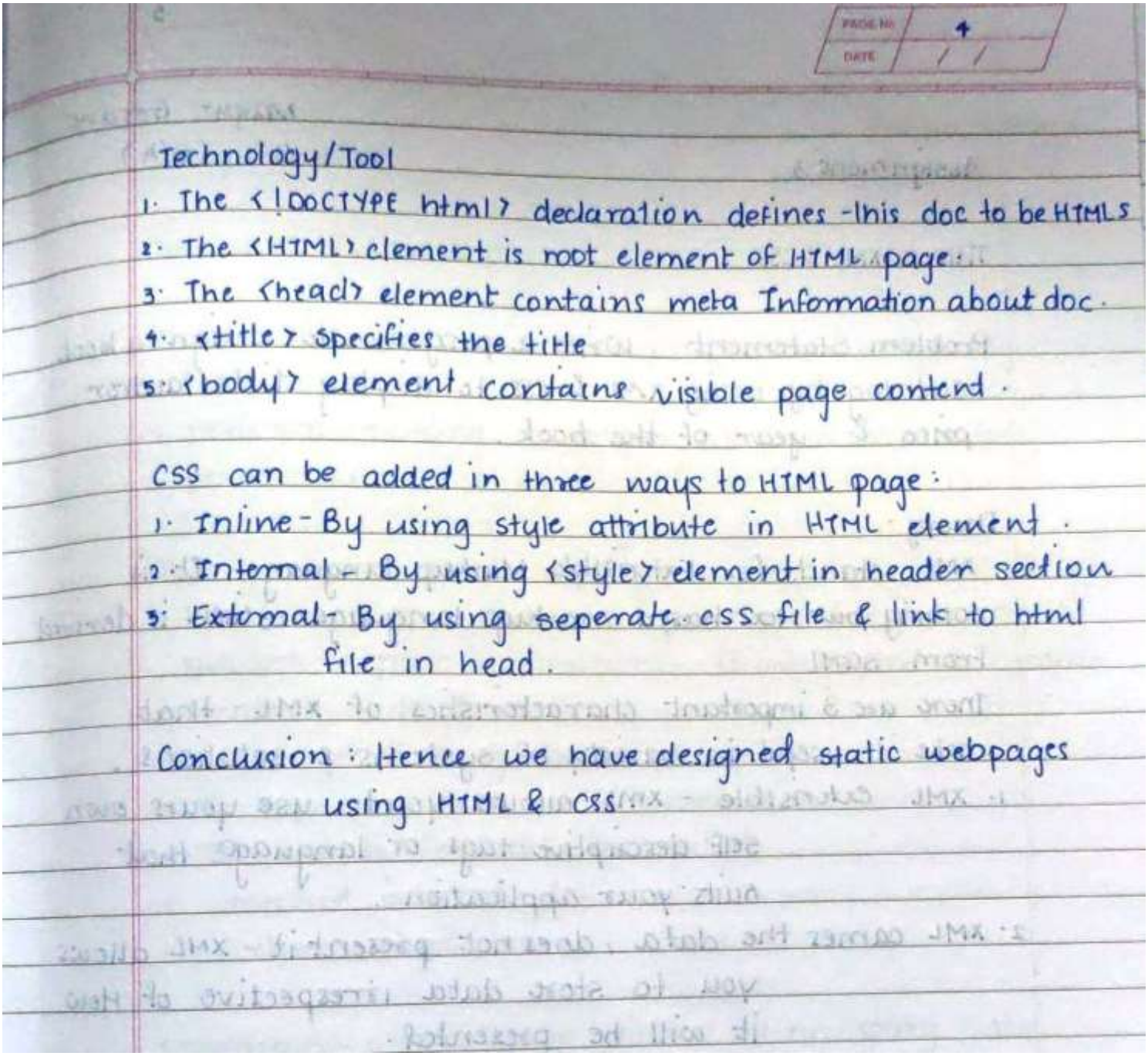
Theory -

HTML - It is markup language for creating webpages

- HTML - Hypertext Markup Language
- HTML elements are building blocks of HTML pages
- HTML elements are represented by a tag
- Browsers donot display HTML tags, but use them to render the content of page.

CSS - Cascading style sheets.
It handles feel & look part of a web page. By css, one can control color of text, styles of fonts, spacing between paragraphs, layout designs.

CSS Modules - 1. Box Model
2. Selectors
3. Background
4. Border
5. Image values & Replaced content.
6. Text Effects
7. Animations
8. 2D/3D transformations
9. User Interface.



Technology/Tool

1. The <!DOCTYPE html> declaration defines -this doc to be HTML5
2. The <HTML> element is root element of HTML page.
3. The <head> element contains meta Information about doc.
4. <title> specifies the title
5. <body> element contains visible page content.

CSS can be added in three ways to HTML page:

1. Inline - By using style attribute in HTML element.
2. Internal - By using <style> element in header section
3. External - By using seperate css file & link to html file in head.

Conclusion : Hence we have designed static webpages using HTML & CSS.

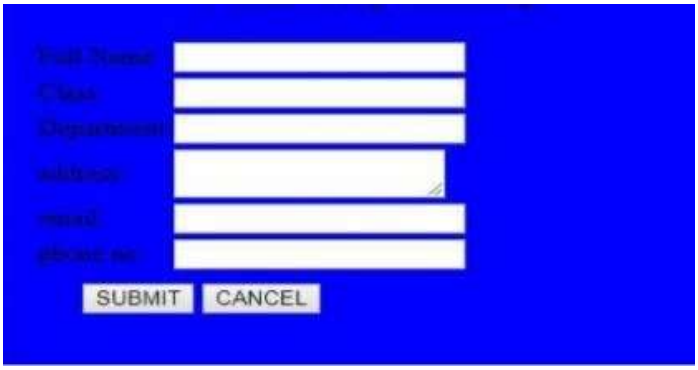
Program and Code:

Registration.html

```
<html>
<head>
<title>regform</title>
<link href="samp.css" rel="stylesheet"></link>
</head>
<body>
<div class="con">
<form action="display.html" method="get" class="con2">
<imgsrc="D:/handshake.jpg" width=380px height=100px></img>
</div>
<div class="main">
<table align="center">
```

```
<tr>
<td><b><br>Welcome to Registration
Page<br><br></td></tr>
</table>
<div class="con1">
<table align="center">
<tr>
<td>Full Name:</td>
<td><input type="text" value=""></input></td>
</tr>
<tr>
<td>Class</td>
<td><input type="text" value=""></input></td>


</tr>
<tr>
<td>Department</td>
<td><input type="text" value=""></input></td>
</tr>
<tr>
<td>address:</td>  <td><textarea row="3"
column="2"></textarea></td>
</tr>
<tr>
<td>email:</td>
<td><input type="text" value=""></input></td>
</tr>
<tr>
<td>phone:</td>
<td><input type="text" value=""></input></td>
</tr>
<tr></tr><tr></tr>
</table>
<table align="center">
<tr>
<td><input type="submit" value="SUBMIT"></td>
<td><input type="reset" value="CANCEL"></td>
</tr>
```

```
</table>
</div>
</div>
</div>
</body>
</html> samp.css
 .con
 {
 width:400px;  height:400px;  background-color:blue;
 color:blue;
 padding:0px 10px 0px 10px;  margin:auto;
 }
 .main
 {
 width:400px;  height:80px;  float:left;
 padding:60px 10px 10px10px;  margin: auto;
 }
 .con1
 {
 width:200px;  height:30px;  float:left;
 margin:auto;  padding:0px 0px 0px0px;
 }

.con2
  {
width:150px;
  height:30px;  float:left;
  margin:10px 10px 10px10px;  padding:0px 0px
  0px0px;
}
  Display.css
  <html>
  <head>
  <title> display</title>
  </head>
  <body>
</html>
```


SUBMIT
CANCEL

Page No. 5
Date / /

Mayur Gorane
48 TE(A)

Assignment 3

Title :- XML & CSS

Problem Statement - Write a program to design a book catalog by using XML & CSS to display title author price & year of the book.

Theory -
XML stands for Extensible Markup language. It is nothing but text based markup language which is derived from SGML
There are 3 important characteristics of XML that make it useful in variety of systems & solutions.
1. XML extensible - XML allows you to use yours own self descriptive tags or language that suits your application.
2. XML carries the data, does not present it - XML allows you to store data irrespective of How it will be presented.

Design / Execution steps -
Following steps are used to create & execute web app^n
1. Write XML code & save with .xml extensions
2. write CSS code.
3. Import css file in XML page.
4. Open XML page in browser.

Conclusion - Hence we have designed static web pages using XML & CSS.

Program and Code:

Book.xml

```
<?xml version="1.0" encoding="UTF-8"?> <?xml-stylesheet type="text/css"
href="book_catalog.css"?> <CATALOG>

<BOOK>

<TITLE>Database Management System</TITLE>
<AUTHOR>Korth</AUTHOR>

<PRICE>500</PRICE>

        </BOOK>

        <BOOK>

        <TITLE>Computer Network</TITLE>

        <AUTHOR>Tenenbaum</AUTHOR>

        <PRICE>600</PRICE>

        <YEAR>1985</YEAR>

        </BOOK>

        <BOOK>

        <TITLE>Software Engineering and project Management</TITLE>

        <AUTHOR>Roger Pressman</AUTHOR>
        <PRICE>600</PRICE>

        <YEAR>1985</YEAR>

        </BOOK>

       </CATALOG>
```

book_catalog.css

```
BOOK {

Display:  block;  margin-left:0;

magin-bottom: 30pt;

}

CATALOG {

Width:100%; background-color: #ffffff;

}

TITLE {

Color: ff0000; display: block;

font-size: 20pt;


}

AUTHOR {
```

```
        display: block; color:
        #0000ff; font-size: 20pt;
    }
    YEAR, PRICE {
        Color:#00000;
        Display: block;
        Margin-left: 20pt;
}
```

Database Management System
Korth
500
1985

Computer Network
Tenenbaum
600
1985

Software Engineering and project Management
Roger Pressman
600
1985

PAGE NO. 6.
DATE / /

Mayur Gorone
48 TE (A)

Assignment - 4

Title - HTML & Javascript

Problem Statement - Write a program to design registration form for students by using HTML, CSS & Javascript & perform following validations all fields mandatory phone number & email address validation.

Theory - Javascript is a programming language of html as well as web. Its preferred for creating Network centric applications. It integrated & complimentary with Java.

Advantage of using Javascript are -

1. It requires a less server interaction
2. Immediate feedback to visitors
3. Increased interactivity
4. Richer interfaces.

Validation - When client enters all necessary data and submit button from validation is done at server side if data entered by a client is incorrect or missing server needs to send all data back to client & request for resubmission of form with correct information

Design & execution steps:

1. Write on HTML code in notepad with HTML extension
2. Write function for validation of email id & phone no & enclosed this function in script.
3. Call this function on 'onclick' event of submit button.
4. open html page in browser.

PAGE NO. 7.
DATE / /

Conclusion - Hence we applied validate data using Javascript

Q1. Differences between undefined & not defined in Javascript

→

Undefined	Not defined.
whenever we declare a variable without assigning any value to it Javascript implicity assigns its value as undefined	Not defined is a variable which is not declared at a given point of time with declaration like var, let or const

Q2. What is closure in Javascript ?

→ 1. A closure is combination of function bundled together with reference to its surrounding state.
2. Closure gives you access to an outer functions scope from an inner function.

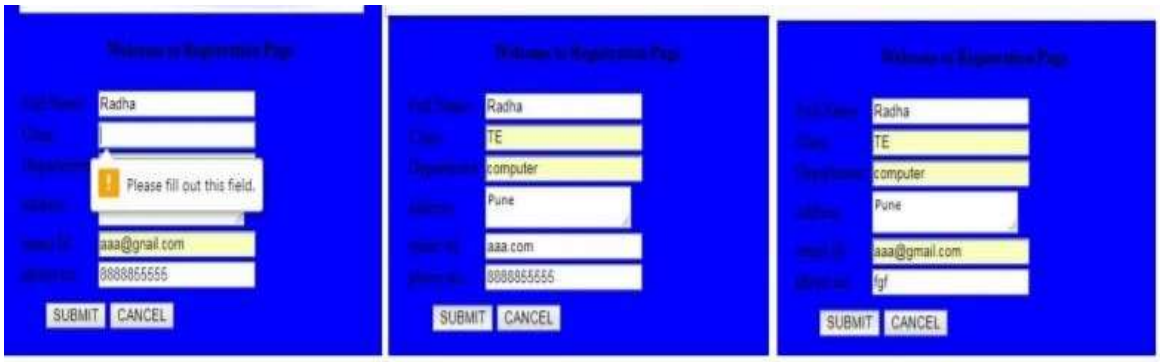
```
<html>
<head>
<title>regform</title>
<link href="samp.css" rel="stylesheet"></link>
<script type="text/javascript">  function
Validation()
{
varemailID = document.myForm.EMail.value;
varnum=document.myForm.ph.value;  atpos =
```

```
emailID.indexOf("@"); dotpos = emailID.lastIndexOf(".");
if (atpos< 1 || ( dotpos - atpos< 2 ))
{  alert("Please enter correct email ID")
document.myForm.EMail.focus() ;  return
false;
}  else if (isNaN(num)){
document.getElementById("ph");
alert("Please enter numeric only.")  return
false;  }else{  return true;
}
}
</script>
</head>
<body>
<div class="con">
<form name="myForm" action="display.xml" method="get" align="center">
<div class="con2">
<imgsrc="D:/handshake.jpg" width=380px height=100px></img>
</div>
<div class="main">
<table align="center">
<tr><td><b><br>Welcome to Registration
Page<br><br></td></tr> </table>
<div class="con1">
<table align="center">
<tr>

<table align="center">
<td><input type="submit" value="SUBMIT" onclick="return
(Validation());"></td>
<td><input type="reset" value="CANCEL"></td>
</tr>
</table>
</div>
</div>
</form>
```

</body>

</html>

Output:



PAGE No. 8
DATE / /

Mayur Gorane
48 TE(A)

Assignment 5

Title - JSP, Servlet & MYSQL (Backend).

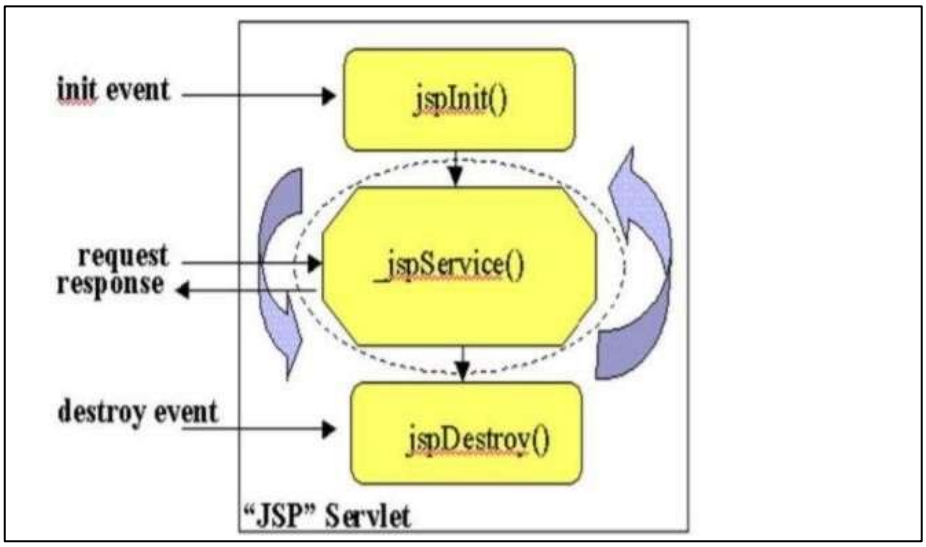
Problem Statement -
1. Design & build login page using JSP, servlet & MySQL
2. Design & Build employee login page using JSP, Servlet & MySQL

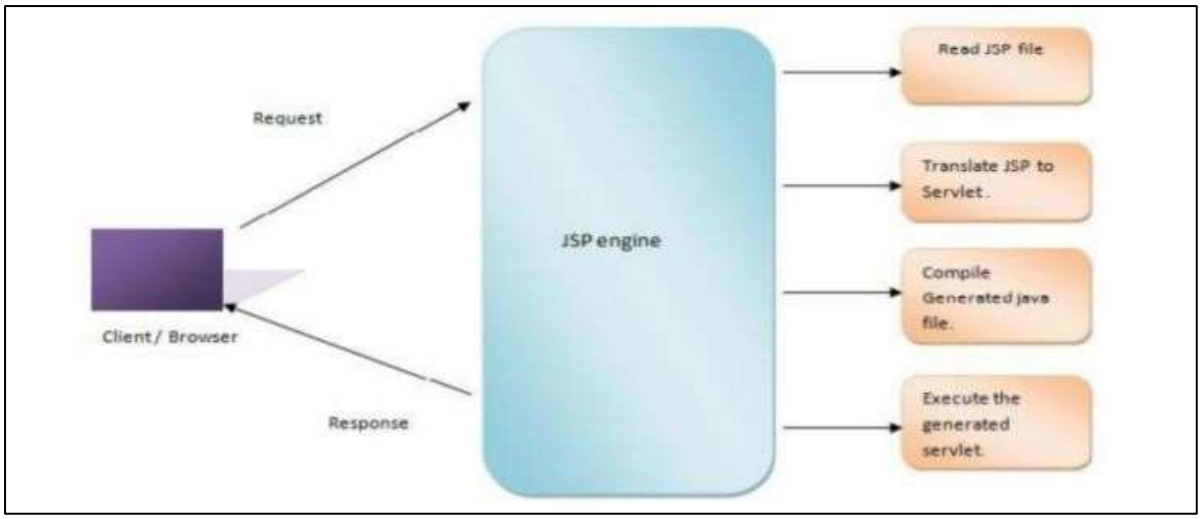
Theory -
Java Server pages - It is a server side programming technology that is used to create dynamic - webbased app". JSP have right to use compute JAVA APIs, Including JDBC API to access database.

Why we need JSP.
JSP is used for design dynamic web page & servlet is used to code logic that is present.

Architecture of JSP -
1. The request / response part of a JSP is defined in below arch.
2. The client initiate request using for JSP file using browser
3. WebServer invokes JSP file & interpret JSP file produce a java code
4. One servlet is created JSP engine compiles servlet.
5. Now servlet classes is loaded by container & executes it
6. engine send response back to client.

Syntax of JSP
<% text %>


init event
jspInit()
request
response
_jspService()
destroy event
jspDestroy()
"JSP" Servlet


Request
Client / Browser
Response
JSP engine
Read JSP file
Translate JSP to Servlet.
Compile Generated java file.
Execute the generated servlet.

PAGE No. 9
DATE / /

Design & Execution steps :

1. Design HTML & JSP files with an extension on HTML & JSP.
2. Write database connection page using Servlet.
3. Set MySQL, username, password & database name in database connection page.
4. Start TOMCAT server with port Number.
5. Open browser & type localhost 8084

Conclusion :
Hence we have performed dynamic web app^n using JSP, servlet & MySQL.

What is JSP?
JavaServer pages technology enables you to mix regular static HTML with dynamically generated content.

what is servlet?
A servlet is an extension to a server that enhances server's functionality

What is the purpose of MySQL?
SQL is used to communicate with a database.

Program and Code:

Following pages required to run this application:

1. index.jsp
2. SignUp.jsp
3. User.jsp
4. LoginSuccess.jsp
5. LoginFailure.jsp
6. LoginServlet.jsp
7. Login.jsp

// **index.jsp**

```
<%--
Document : Login
Created on : Dec 31, 2017, 2:07:34 PM
Author : Admin --  %>
<%@page contentType="text/html" pageEncoding="UTF-
8"%> <!DOCTYPE html>

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

        <body>
        <table>
        <tr>
        <td width="300"></td>  td> <font color="#339900" size="3.5">User
 Login</font> <table>  <tr>
        <td>
        <form id="user1" method="post" action="User.jsp" name="s"
        onSubmit="return valid()">
        <table border="2.0">
        <tr>
        <td>
        User Id</td>
        <td><input type="text" name="user1"></td>
        </tr>
```

```
<tr>
<td>
Password</td>
<td><input type="password" name="pass"></td>
</tr>
<tr>
<td>
<center><input type="submit" name="user"
value="Login"></center> </td>
</tr></table>
<center><table>  <tr>

<td>
<a href="SignUp.jsp"><font color="#000000" size="2"><b>New
User?</b></font></a></td></tr></table>
</td>
</tr></table></center>
</td>
</tr>
</table>
</form>
</body>

</html>
```

//**SignUp.jsp**

```
<%@ page import="java.sql.*"%>
<%@ page import =
"java.util.Date,java.text.SimpleDateFormat,java.text.ParseException"%
> <%-- Document : SignUp
Created on : Dec 30, 2017, 11:52:03 PM
Author : Admin
--%>
<%@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 bgcolor="Skyblue">

<%


java.util.Date now = new java.util.Date();

String DATE_FORMAT1 = "dd/MM/yyyy";

SimpleDateFormat sdf1 = new
SimpleDateFormat(DATE_FORMAT1);

String strDateNew1 = sdf1.format(now);

%>

<table border="black" bgcolor=" black" align ="center" >
<tr>

<td> <strong><em><font color="#990000" size="+1" face="Times New
Roman" style="text-decoration:underline">User
Registration</font></em></strong> </td>

</tr>

</table>


<table border="black" align ="center">

<tr>

<td><form name="s" action="LoginServelt.jsp" method="get"
onSubmit="return

valid()">

<table align="center" cellpadding="6" cellspacing="6"
width="400" height="300"><td>

<tr>

<td><font face="Times New Roman"
size="+1"><strong>Name</strong></font></td><td> &n
bsp;   <input type="text" name="unn"
class="b"></td>


</tr>

<tr>

  <td><font face="Times New Roman" size="+1"><strong>User
ID</strong></font></td><td>    
<input type="text" name="uidd" id="name"></td></tr>

<tr>



</body>

</html>
```

//LoginSuccess.jsp

```
<%--

Document : LoginSuccess

Created on : Dec 31, 2017, 2:40:04 PM
```

```
Author : Admin

--%>

<%@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> User Logged In Successfully!...</h1>

</body>

</html>

//LoginFailure.jsp

<%--

Document : LoginFailure

Created on : Dec 31, 2017, 2:42:21 PM

Author : Admin

--%>

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

<html>

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

</head>

<body>

    <h1> <font color="Red"> Please Provide Correct Username or Password!..Retry
Again!.. </font> </h1>

<table>

<tr>

<td>

<form id="user1" method="post" action="User.jsp" name="s"
onSubmit="return valid()">

<table>

<tr>

<td>

User Id</td>

<td><input type="text" name="user1"></td>

</tr>

<tr>
```

```
<td>
Password</td>

<td><input type="password" name="pass"></td>
</tr>
<tr>
<td>
</td>
<td>
<input type="submit" name="user"  value="Login"></td>
</tr></table>  </form>

 </td>
 </tr></table>
 </body>
 </html>
```

//**LoginServlet.jsp**

```
<%@page import="java.sql.*"%>
<%@ page import="java.io.*"%>
<%@ page import =
"java.util.Date,java.text.SimpleDateFormat,java.text.ParseExcep
tion"%><%

String a = request.getParameter("uidd");
String x = request.getParameter("unn");
String b = request.getParameter("passs");
String c = request.getParameter("mobb");  String
d = request.getParameter("eidd");
session.setAttribute("d",d);

//String f=request.getParameter("date");
//out.print(strDateNew1);
//out.print(x);
//out.print(b);
//out.print(c);
//out.print(d);
//out.print(f);
java.util.Date now = new java.util.Date();
```

```
 String DATE_FORMAT = "yyyy-MM-dd";
 SimpleDateFormat sdf = new SimpleDateFormat(DATE_FORMAT);
 String strDateNew = sdf.format(now) ;




//response.sendRedirect("signup.jsp?message=success");
//String userid=null;  //String m="avl";

try

{

Class.forName("com.mysql.jdbc.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/loginpag e","root","root");

PreparedStatement ps=con.prepareStatement("insert into
student(sname,suserid,spass,mobile,email,date)
values(?,?,?,?,?,'"+strDateNew+"')"); //ResultSet
rs=ps.executeQuery();  ps.setString(1,x);  ps.setString(2,a);
ps.setString(3,b);  ps.setString(4,c);  ps.setString(5,d);

ps.executeUpdate();  response.sendRedirect("SignUp.jsp?success");

}  catch(Exception e1)
{  out.println(e1.getMessage());
response.sendRedirect("SignUp.jsp?Failure");%>
}

%>
```

//User.jsp

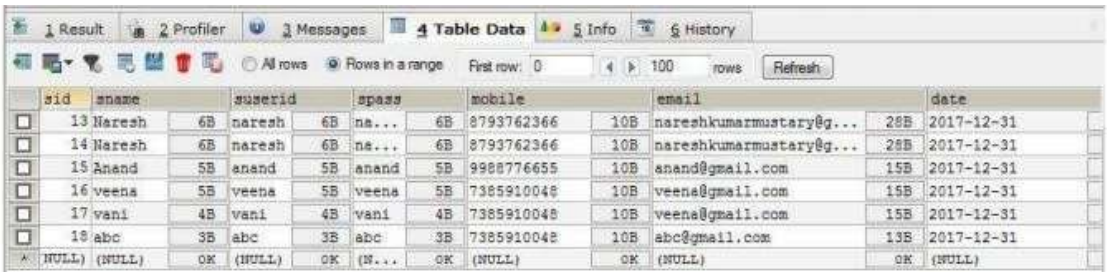
```
<%@ page import="java.sql.*;"%>
<%

String a=request.getParameter("user1");
String b=request.getParameter("pass");

String id=null,name=null,userid=null,email=null;  try{
Class.forName("com.mysql.jdbc.Driver");
Connection con =
```

```
DriverManager.getConnection("jdbc:mysql://localhost:3306/loginpag e","root","root");
//Connection con = databasecon.getconnection();

PreparedStatement ps=con.prepareStatement("select sid,sname,suserid,email
fromstudent wheresuserid='"+a+"' && spass='"+b+"'");

ResultSet rs=ps.executeQuery();


if(rs.next())

{  id=rs.getString("sid");  name=rs.getString("sname");

userid=rs.getString("suserid");  email=rs.getString("email");

session.setAttribute("sid",id);  session.setAttribute("sname",name);

session.setAttribute("suserid",userid);

session.setAttribute("email",email);

//response.sendRedirect("user5.jsp");

response.sendRedirect("LoginSuccess.jsp?Success");

//out.print(name2);
```

```
}
else
{
response.sendRedirect("LoginFailure.jsp?Failure");
}
}
catch(Exception e2){
out.println(e2.getMessage());
}
%>
```

//Database Name: loginpage

//Table Name: student



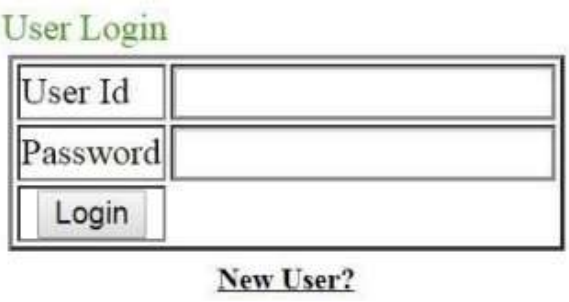
1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

All rows | Rows in a range | First row: 0 | 100 rows | Refresh

sid	sname	suserid	spass	mobile	email	date
13	Naresh 6B	naresh 6B	na... 6B	8793762366 10B	nareshkumarmustary@g... 28B	2017-12-31
14	Naresh 6B	naresh 6B	na... 6B	8793762366 10B	nareshkumarmustary@g... 28B	2017-12-31
15	Anand 5B	anand 5B	anand 5B	9988776655 10B	anand@gmail.com 15B	2017-12-31
16	veena 5B	veena 5B	veena 5B	7385910048 10B	veena@gmail.com 15B	2017-12-31
17	vani 4B	vani 4B	vani 4B	7385910048 10B	veena@gmail.com 15B	2017-12-31
18	abc 3B	abc 3B	abc 3B	7385910048 10B	abc@gmail.com 13B	2017-12-31
NULL	(NULL) 0K	(NULL) 0K	(N... 0K	(NULL) 0K	(NULL) 0K	(NULL)

Output:

Step 1: Run index.jsp page



User Login

User Id

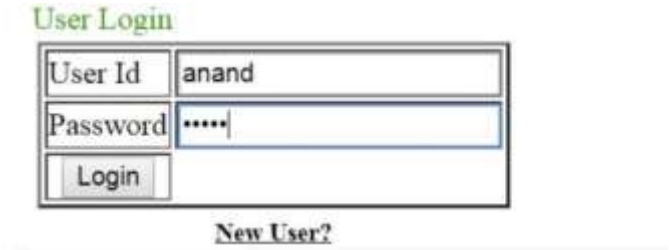
Password

Login

New User?

Step 2: If user is already registered then use own credentials else click New User for registration

Step 3: Once registered click Login directly



Step 4: If user gives wrong username and password, it will show error message

PAGE No. 10
DATE / /

Mayur Gorane
48 TE(A)

Assignment 6

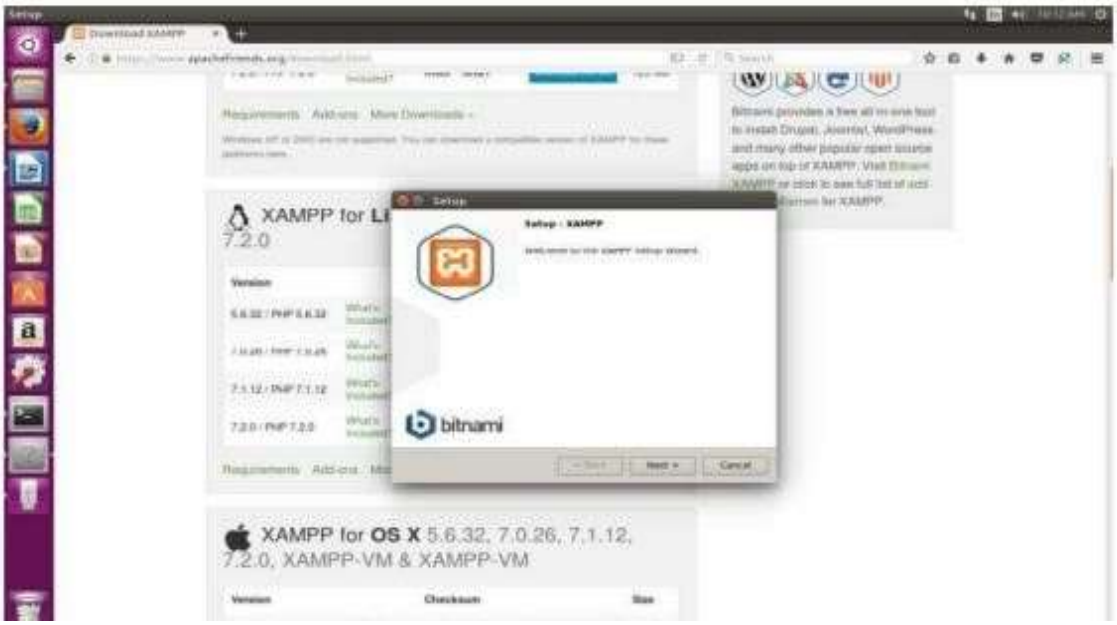
Title : Add dynamic web appⁿ essence using PHP, HTML & MySQL.

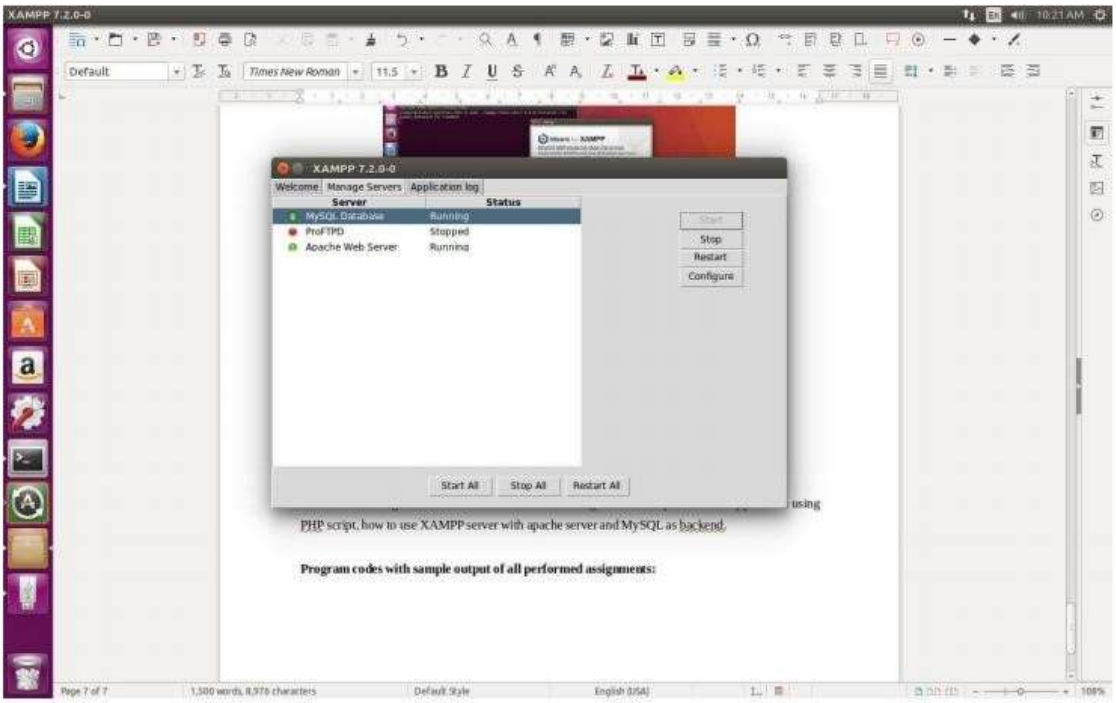
Problem Statement :
Design & develop dynamic web application using PHP & MySQL as a backend for employee data with insert, delete, view & update operations.

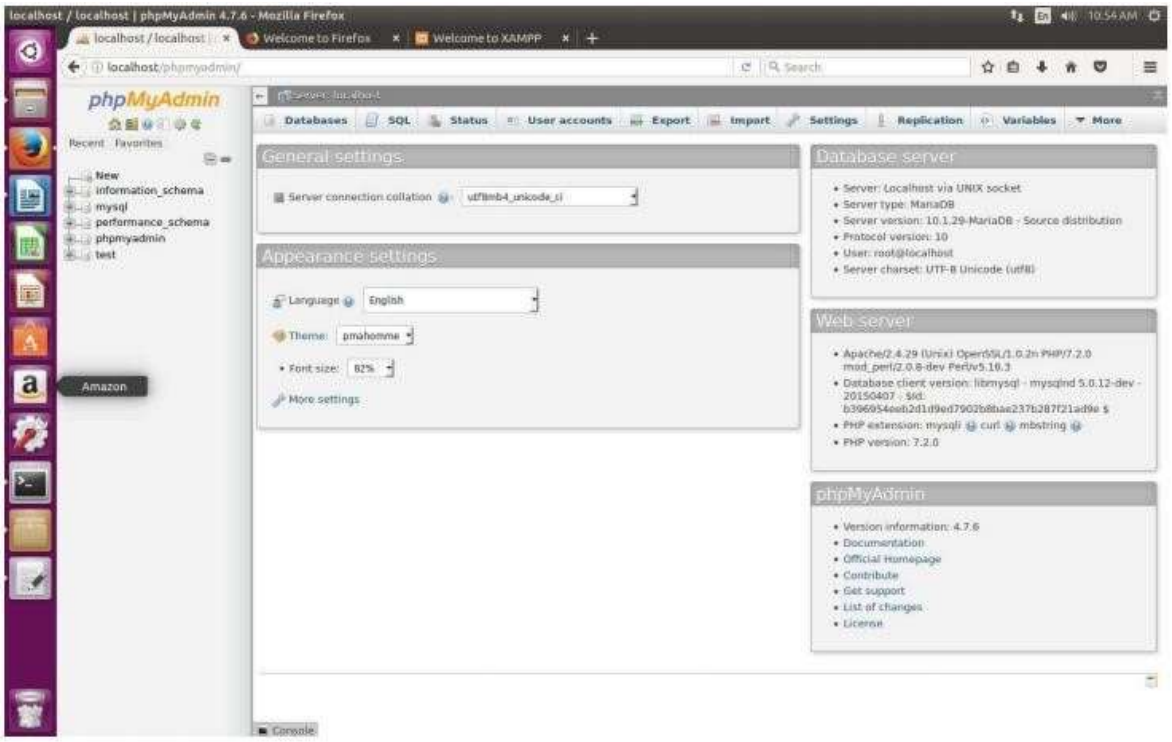
Theory -
1. PHP - PHP is a hypertext preprocessor began as little open source venture that advanced as an ever increasing no of individuals discovered how valuable it was. PHP is a server side scripting dialer that is installed in HTML. It is ~~va~~ utilize to oversee dynamic substance, database. It is incaporated with various prevalent database including MySQL PostgresSQb, oracle, sybase server.
2. $MySQL - Its most famous open source relational database Management System.

What is database ?
→ A database is different application that stores a gathering of information. Every database has atleast one mistakables API's for making/getting to over seeing & reacting information it holds.

Conclusion : In this assignment we have studied how to design & develop small web application using PHP script, XAMPP server with apache server & MySQL as


Setup
Download XAMPP
Requirements
Add-ons
More Downloadable
XAMPP for Li
7.2.0
Version
Setup - XAMPP
bitnami
Next >
Cancel
XAMPP for OS X 5.6.32, 7.0.26, 7.1.12, 7.2.0, XAMPP-VM & XAMPP-VM
Version
Checksum
Size


XAMPP 7.2.0-0
10:21 AM
Default
Times New Roman
11.5
XAMPP 7.2.0-0
Welcome
Manage Servers
Application log
Server
Status
MySQL Database
Running
ProFTPD
Stopped
Apache Web Server
Running
Start
Stop
Restart
Configure
Start All
Stop All
Restart All
using
PHP script, how to use XAMPP server with apache server and MySQL as backend.
Program codes with sample output of all performed assignments:
Page 7 of 7
1,300 words, 8,978 characters
Default Style
English (USA)
108%



Program and Code:

Here we have created **"**emp**"** directory in htdocs directory contain following files

1. index.php
2. add.html
3. add.php

4. edit.php

5.delete.php

6. config.php

7. database.sql(optional)

1. index.php

```
<?php
//including the database connection file  include_once("config.php");

//fetching data in descending order (lastest entry first)

//$result = mysql_query("SELECT * FROM users ORDER BY id DESC"); // mysql_query is deprecated

$result = mysqli_query($mysqli, "SELECT * FROM users ORDER BY id
DESC"); // using mysqli_query instead


?>


<html>
<head>
<title>Homepage</title>
</head>


<body>
<a href="add.html">Add New Data</a><br/><br/>


<table width='80%' border=0>


<tr bgcolor='#CCCCCC'>
<td>Name</td>
<td>Age</td>
<td>Email</td>
<td>Update</td>
</tr>
<?php
```

```
        //while($res = mysql_fetch_array($result)) { // mysql_fetch_array is deprecated, we needtouse mysqli_fetch_array  while($res = mysqli_fetch_array($result)) {  echo "<tr>";
echo "<td>".$res['name']."</td>";  echo "<td>".$res['age']."</td>";
echo "<td>".$res['email']."</td>";
                echo "<td><a href=\"edit.php?id=$res[id]\">Edit</a> | <ahref=\"delete.php?id=$res[id]\" onClick=\"return confirm('Are you sure you want todelete?')\">Delete</a></td>";
}
?>
</table>
</body>

</html>
```

2. add.html

```
<html>
<head>
<title>Employee Data</title>
</head>

<body>
<a href="index.php">Home</a>
<br/><br/>

<form action="add.php" method="post" name="form1">
<table width="25%" border="0">
<tr>
                <td>Name</td>
                <td><input type="text" name="name"></td>
</tr>
```

```
<tr>
                                        <td>Age</td>
                                            <td><input type="text" name="age"></td>
</tr>
<tr>
                                        <td>Email</td>
                                            <td><input type="text" name="email"></td>
</tr>
<tr>
                                        <td></td>
 <td><input type="submit" name="Submit" value="Add"></td>
</tr>
</table>
</form>
</body>
</html>
```

3. add.php

```
<html>
<head>
<title>Add Data</title>
</head>

<body>
<?php
        //including the database connection file  include_once("config.php");
        if(isset($_POST['Submit'])) {
        $name = mysqli_real_escape_string($mysqli, $_POST['name']);
```

```
$age = mysqli_real_escape_string($mysqli, $_POST['age']);

$email = mysqli_real_escape_string($mysqli, $_POST['email']);

// checking empty fields

if(empty($name) || empty($age) || empty($email)) {


if(empty($name)) {  echo "<font color='red'>Name field is empty.</font><br/>";

}


if(empty($age)) {  echo "<font color='red'>Age field is empty.</font><br/>";

}


if(empty($email)) {  echo "<font color='red'>Email field is empty.</font><br/>";

}


//link to the previous page

                echo "<br/><a href='javascript:self.history.back();'>Go
                                                                    Back</a>";  } else {


// if all the fields are filled (not empty)


//insert data to database

                    $result = mysqli_query($mysqli, "INSERT INTO
users(name,age,email) VALUES('$name','$age','$email')");


//display success message

echo "<font color='green'>Data added successfully.";

            echo "<br/><a href='index.php'>View Result</a>";

}
```

```
}

?>

</body>

</html>
```

4. edit.php <?php

```
// including the database connection file  include_once("config.php");

if(isset($_POST['update']))

{

$id = mysqli_real_escape_string($mysqli, $_POST['id']);


$name = mysqli_real_escape_string($mysqli, $_POST['name']);

$age = mysqli_real_escape_string($mysqli, $_POST['age']);

$email = mysqli_real_escape_string($mysqli, $_POST['email']);

 // checking empty fields

if(empty($name) || empty($age) || empty($email)) {


if(empty($name)) {  echo "<font color='red'>Name field is empty.</font><br/>";

}



if(empty($age)) {  echo "<font color='red'>Age field is empty.</font><br/>";

}


if(empty($email)) {  echo "<font color='red'>Email field is empty.</font><br/>";  }

} else {

//updating the table

                                                $result = mysqli_query($mysqli, "UPDATE users SET
name='$name',age='$age',email='$email' WHERE id=$id");
```

```
                                        //redirectig to the display page. In our case, it is index.php
        header("Location: index.php");
        }
        }
        ?>
        <?php

        //getting id from url
        $id = $_GET['id'];
        //selecting data associated with this particular id
        $result = mysqli_query($mysqli, "SELECT * FROM users
        WHERE id=$id"); while($res = mysqli_fetch_array($result)) {

        $name = $res['name'];
        $age = $res['age'];
        $email = $res['email'];
        }
        ?>
        <html>
        <head>
        <title>Edit Data</title>
        </head>

        <body>
        <a href="index.php">Home</a>
        <br/><br/>

<form name="form1" method="post" action="edit.php"> <table border="0">
```

```
<tr>
<td>Name</td>
<td><input type="text" name="name" value="<?phpecho
$name;?>"></td>
</tr>
<tr>
<td>Age</td>
<td><input type="text" name="age" value="<?phpecho
$age;?>"></td>
</tr>
<tr>
<td>Email</td>
<td><input type="text" name="email" value="<?phpecho
$email;?>"></td>
</tr>
<tr>
<td><input type="hidden" name="id" value=<?phpecho
$_GET['id'];?>></td>
<td><input type="submit" name="update" value="Update"></td>
</tr>
</table>
</form>
</body>
</html>
```

5. delete.php

```
<?php
include("config.php");
```

```
$id = $_GET['id'];

$result = mysqli_query($mysqli, "DELETE FROM users

WHERE id=$id"); header("Location:index.php");

?>
```

6. config.php

```
<?php

$database = 'localhost';


$dbName = 'test';

$dbUser = 'root';

$dbPass = '';

$mysqli = mysqli_connect($database, $dbUser, $dbPass, $dbName);


?>
```

7. database.sql

a. create database emp;

b. use emp;

c. CREATE TABLE `users` (

`id` int(11) NOT NULL auto_increment,

`name` varchar(100) NOT NULL,

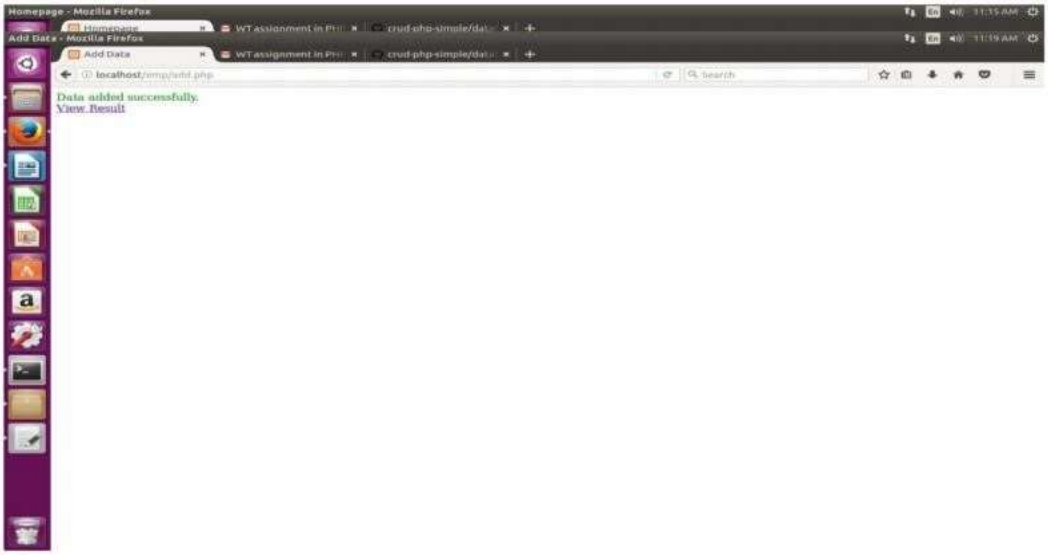
`age` int(3) NOT NULL,

`email` varchar(100) NOT NULL,

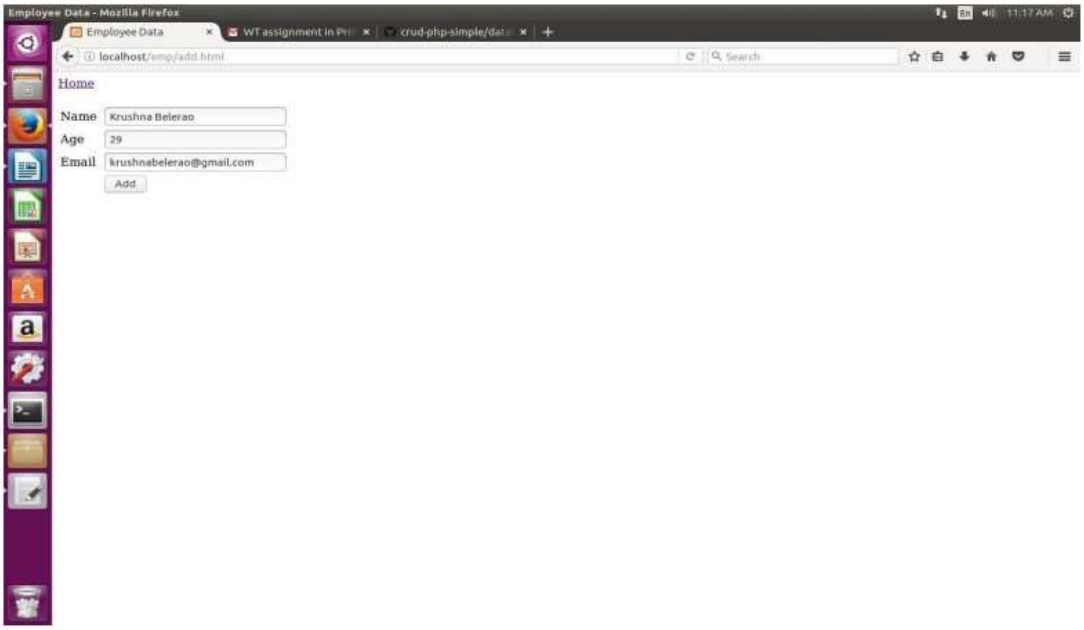
PRIMARY KEY (`id`)

);

1. First Window

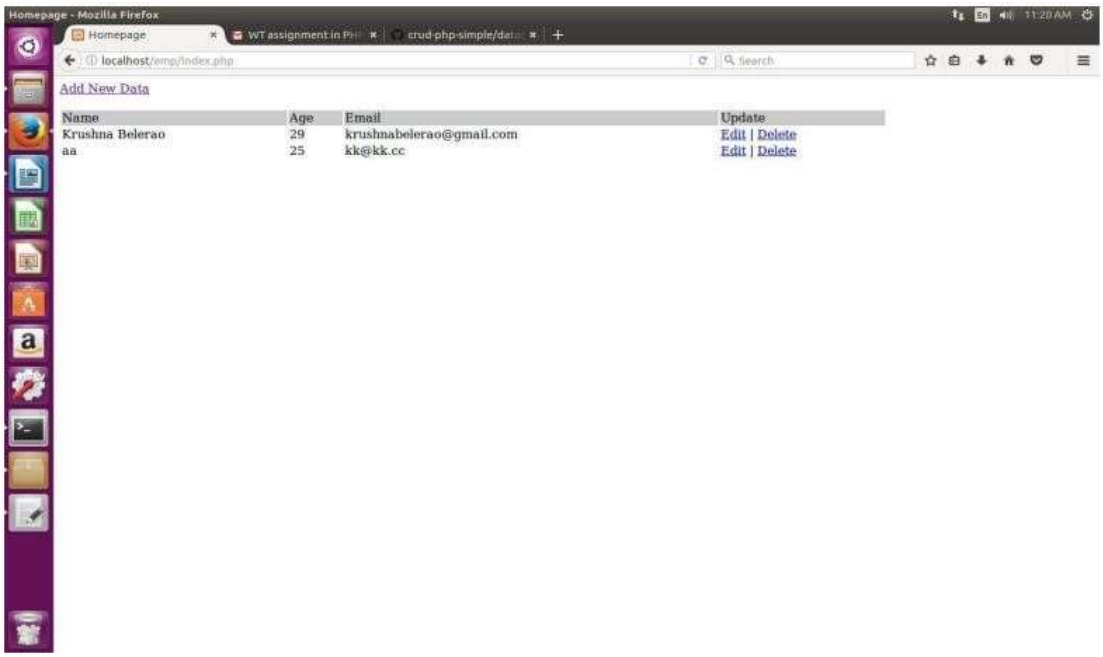


2. Second Window click on “Add New Data” see top left corner link



3. click “add” to add data in database and it will display below window

4. Click on “View Result” below image will be displayed with all details in Database



5. for delete and edit, use Update link from the right side

PAGE NO. 11
DATE / /

Assignment 7

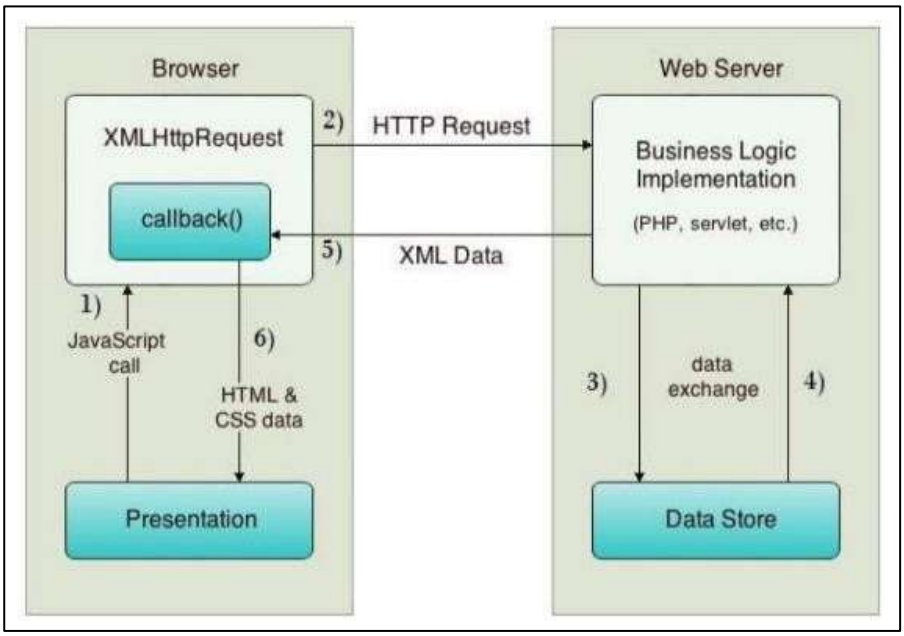
Title - Add dynamic web application using PHP, AJAX & MySQL.

Problem statement - Design & develop web application using PHP, AJAX & MySQL as a backend for employee data with insert & view operations.

Theory - Concept - Ajax remains Asynchronous JavaScript & XML, ajax is another procedure for making better, speedier & more intelligent dynamic web appn with distance of XML, HTML, CSS & Javascript Ajax utilizes XHTML for content, CSS for introduction, alongside Document object Model & Javascript for dynamic substance show AJAX instructional exercise covers ideas & cases of AJAX innovation for apprantices & experts.
AJAX is an acronym for Asynchronous Javascript & XML. Its gathering of between related innovation like Javascript, DOM, HTML, XML, CSS & so forth.

Technology/Tool - AJAX, PHP & MySQL
Design & execution steps - for design purpose HTML & CSS is to be used for design part contains GUI of web application. How its look like?

Conclusion - In this assignment we have studied how to design & develop small web application using PHP, javascript, ajax, XAMPP server with Apache Server.



Program and Code:

Here we have created **"**emp**"** directory in htdocs directory contain following files 1. index.html

2. insert.php

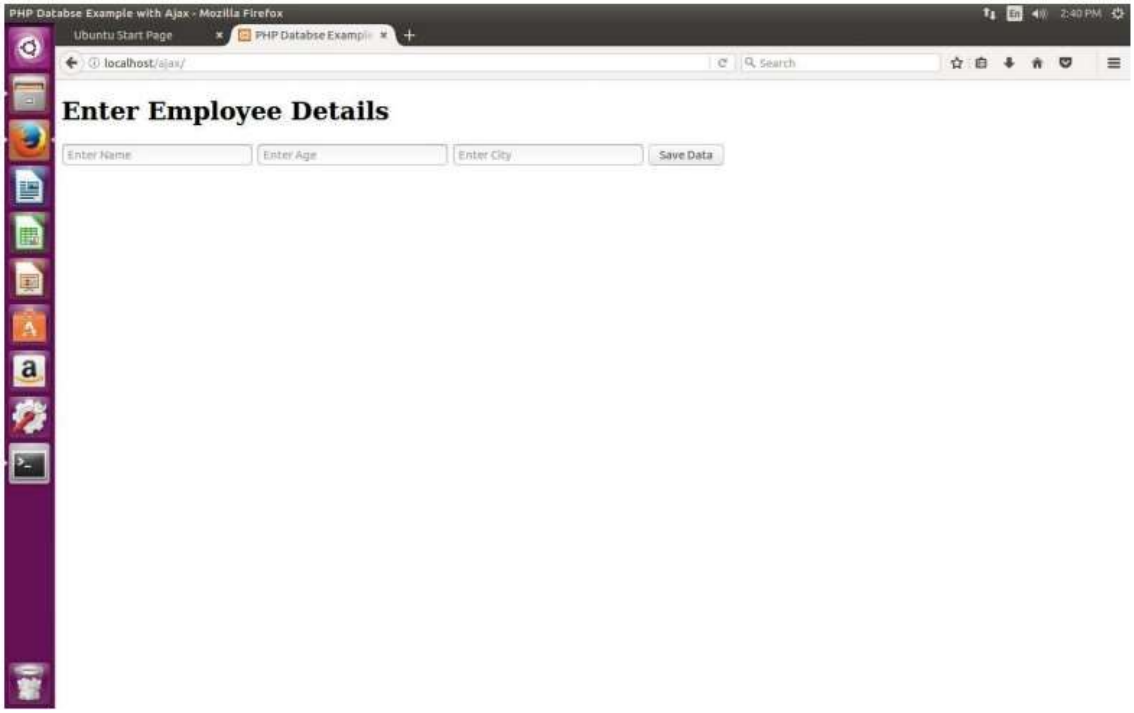
3. display.php

4. config.php

1.index.html

```
<!DOCTYPE html>
<html>
<head>
<title>PHP Databse Example with Ajax </title>
<script  src="https://code.jquery.com/jquery-3.2.1.min.js"
integrity="sha256-
hwg4gsxgFZhOsEEamdOYGBf13FyQuiTwlAQgxVSNgt4=" crossorigin="anonymous">
</script>
</head>
<body>
<h1> Enter Employee Details </h1>
<form method="post" action="insert.php">
```

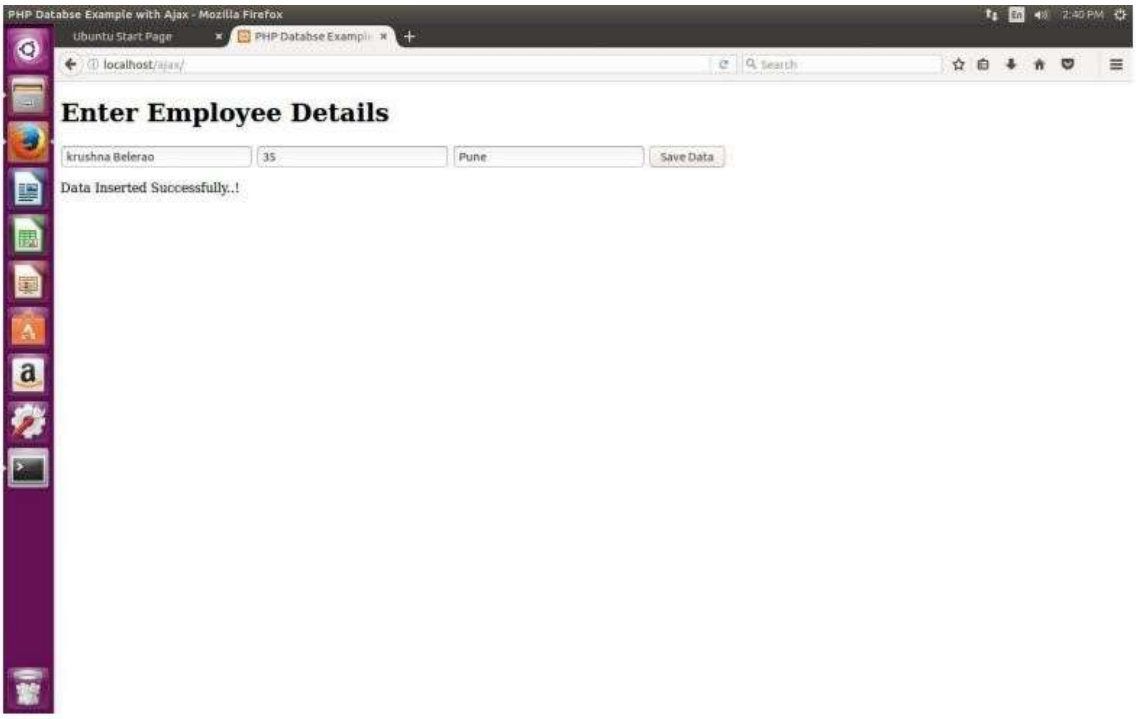
```
<input type="text" id="name" name="name" placeholder="Enter
Name" /> <input type="text" id="age" name="age" placeholder="Enter
Age" /> <input type="text" id="city" name="city" placeholder="Enter
City" /> <button> Save Data </button>

</form>
<p id="result">
</p>
<p id="result"></p>
<a href="display.php">Display</a>
<!--jquery and ajax code-->
<script>
$("form").submit(function(e){  e.preventDefault();

$.post(
"insert.php",
{  name: $("#name").val(),
age: $("#age").val(),  city:
$("#city").val()

},  function(result)  {
if(result == "success")
{
$("#result").html("Data Inserted Successfully..!");
}  else

{
$("#result").html("Error Occured!");
}
}
);
});
</script>
</script>
</body>
<html>
```



2. insert.php

```
<?php

$name = $_POST['name'];

$age = $_POST['age'];

$city = $_POST['city'];


$con = new mysqli('localhost', 'root', '', 'emp');  if($con->connect_error)

{  echo("Error");  }


$stmt = $con->prepare("insert into users(name,age,city) values (?,?,?)");
$stmt-
>bind_param("sis",$name,$age,$city);



if($stmt->execute())

{  echo("success");

}  else  {

echo("fail");  }


?>
```

in below image URL you can see that without refreshing whole page only part of page refreshed.



3. display.php <?php

include_once("config.php");

//$result = mysql_query("SELECT * FROM users ORDER BY id DESC"); // mysql_queryis deprecated

$result = mysqli_query($mysqli, "SELECT * FROM users ORDER BY id

DESC"); // usingmysqli_query instead

?>

<html>

<body>

Add New Data

<table width='80%' border=0>

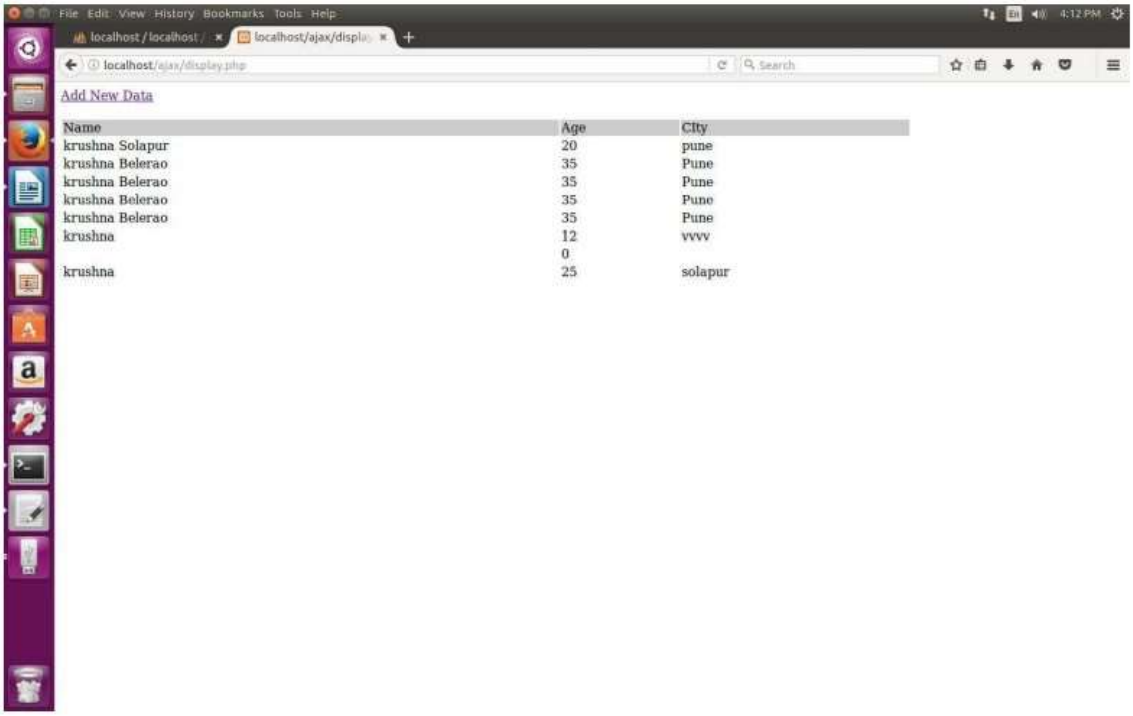
<tr bgcolor='#CCCCCC'>

<td>Name</td>

<td>Age</td>

<td>CIty</td>

```
</tr>
<?php
        //while($res = mysql_fetch_array($result)) { // mysql_fetch_array is deprecated, we needto use mysqli_fetch_array  while($res = mysqli_fetch_array($result)) {  echo "<tr>"; echo "<td>".$res['name']."</td>";  echo "<td>".$res['age']."</td>";  echo "<td>".$res['city']."</td>";  echo "</tr>";  }
?>
</table>
</body>
</html>
```



4. config.php

```
<?php

$host = 'localhost';
$dbname = 'emp';
$dabUser = 'root';
$dbPass = ' ';
$mysqli = mysqli_connect($host, $dbUser, $dbPass,  $dbname); ?>
```

Assignment 8

Mayur Gorane
4 BTECH

Title - Design and develope any web application using structs framework

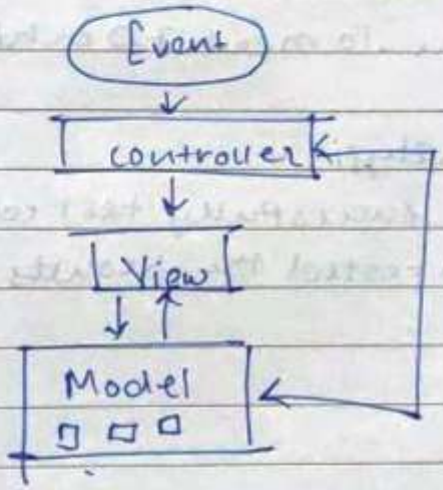
Problem statement-

Create a login module for web application using Struct formula

Theory -

model view controller Architecture.

model view constelle is a way to build application that promoks complete separation between bussiness logic and presentation.



What is Structs

Structs is a Framework that advance the utilize of model view controller engineering for planning substanial scale application The structure incorporate an arrangement of custom tabe libraries and their related Java classes, alongside different ability classes

Strud dogs

Common Attributes

Attribute	Used for
Id	the name of a bean Portemp - oracy use by tag
name	the name of pre-existing bean for use with the tag
Property	the property of bean named in name attribute
Scope	the scope to search fix bean named in the name attribute.

Technology / Tool -

① Eclipse IDE

② Apache .Tomcat 7.0 or higher

Conclusion / Analysis -

Hence we have successfully tested the Struts Framework and tested the results

```
// Creating a Plumber bean in the request scope
Plumber aPlumber = new Plumber();
request.setAttribute("plumber", aPlumber);
```

Beans can be created with the <jsp:useBean></jsp:useBean> tag:

```
<!-- If we want to do <jsp:setProperty ...></jsp:setProperty> or --
> <!-- <jsp:getProperty ... ></jsp:getProperty> -->
<!-- we first need to do a <jsp:useBean ... ></jsp:useBean> -->
<jsp:useBean id="aBean" scope="session" class="java.lang.String">
creating/using a bean in session scope of type java.lang.String
</jsp:useBean>
```

Most useful is the creation of beans with Struts tags:

```
<!-- Constant string bean -->
<bean:define id="greenBean" value="Here is a new constant string
bean; pun intended."/>
<!-- Copying an already existent bean, frijole, to a new bean, lima -->
<bean:define id="lima" name="frijole"/>
<!-- Copying an already existent bean, while specifying the class -->
<bean:define id="lima" name="frijole"
class="com.SomePackageName.Beans.LimaBean"/>
<!-- Copying a bean property to a different scope -->
<bean:define id="goo" name="foo" property="geeWhiz"
scope="request" toScope="application"/>
```

<bean:message ... >

```
<!-- looks up the error.divisionByZero resource -->
<!-- and writes it to the HttpResponse object -- >
<bean:message key="error.divisionByZero"/>
<!-- looks up the prompt.name resource -->
<!-- and writes it to the HttpResponse object; -- >
<!-- failing that, it writes the string -->
<!-- contained in the attribute arg0-- >
<bean:message key="prompt.name" arg0='Enter a name:'/>
```

This tag writes the string equivalent of the specified bean or bean property to the current HttpResponse object.

<bean:write ... >

```
<!-- writes the value of customer.getStreetAddress().toString() --
> <!-- to the HttpResponse object -->
<bean:write name="customer" property="streetAddress"/>
```

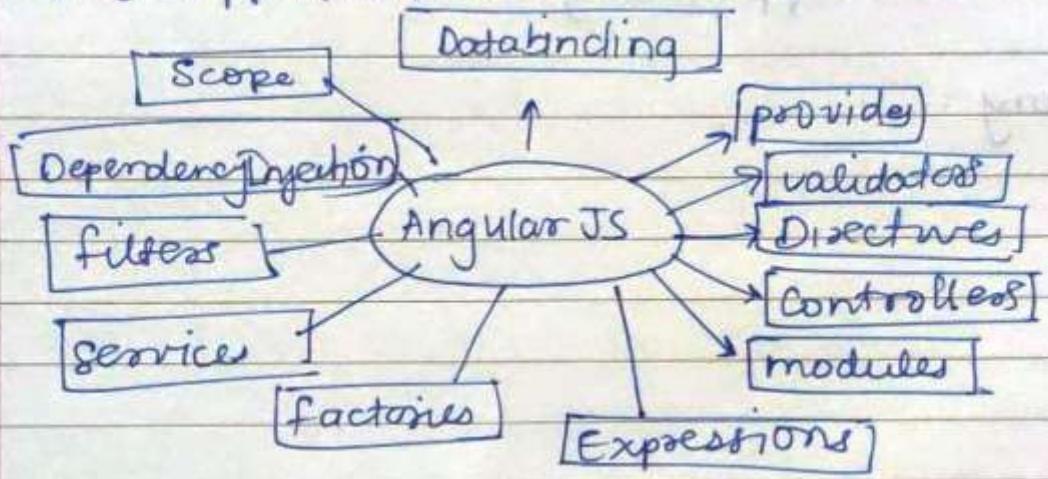
PAGE No.
DATE / /

Mayur Gorare
48 TE(A)

Assignment 9

Title - Design & develop any web app^n using Angular JS

Theory - Angular JS is an open source web application framework

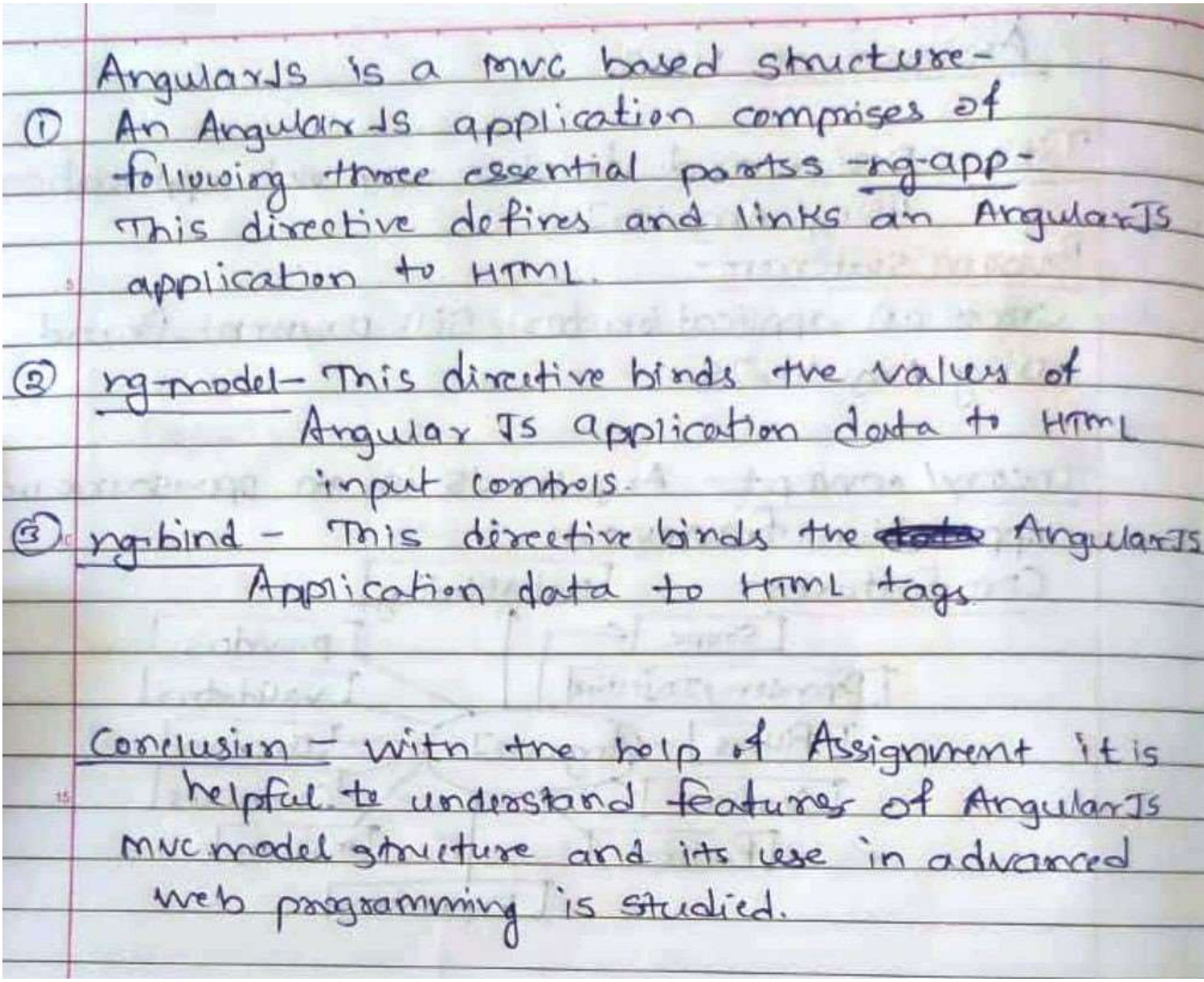


Advantages of AngularJS-

① It gives the ability to make single page Application in a spotless and viable way.
② AngularJS code is unit testable.
③ AngularJS gives reusable segments
④ with AngularJS, the engineers can accomplish great usefulness with short code.

Model view controller-

① Model - It is the most minimal level of exam... incharge of looking after information.
② view - It is in charge of showing all or a pa... of the information to client.
③ controller - It is product code that controls the... connections between the Model & view.



PROGRAM CODE:

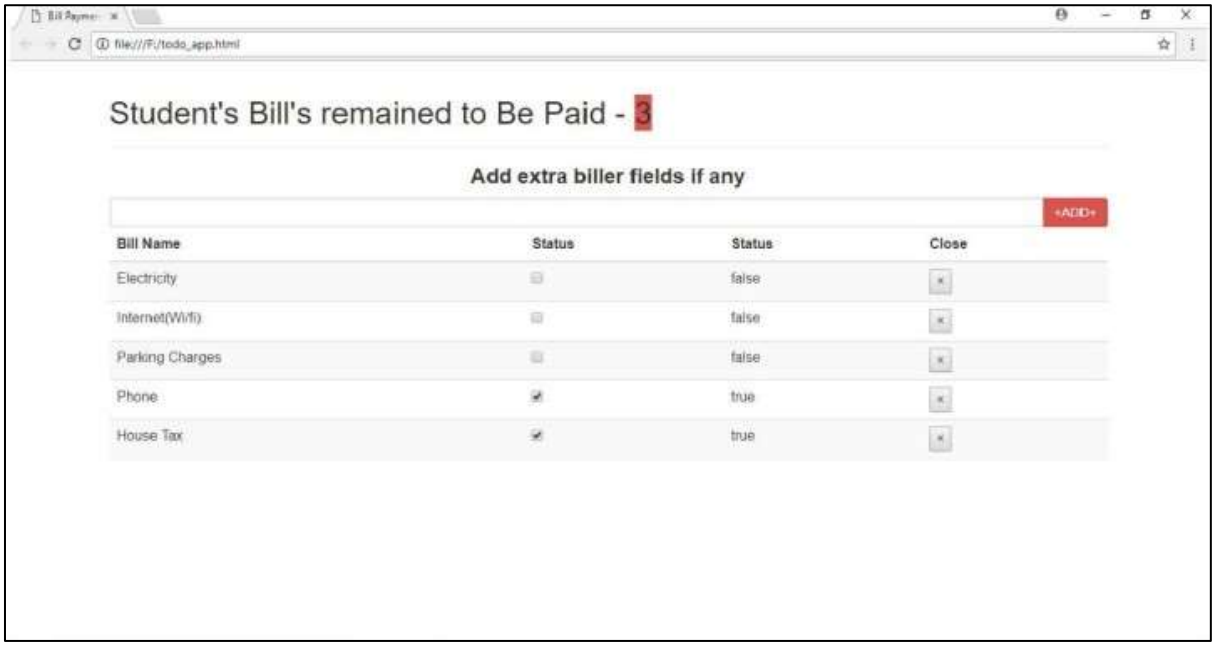
```
<html ng-app="billpayApp">
<!-- SCRIPTS TO BE ADDED IN HEAD TAG --
> <head>
<title>Bill Payment Record using angular and bootstram framework</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<!-- ACCESSING ANGULARJS BY CDN METHOD-->
<script
```

```
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.4/angular.min .js"></script>

<!-- ACCESSING STYLESHEET FOR DESIGN [OPTIONAL PART CAN BE

SKIP]-->

<link rel="stylesheet"

href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.mi n.css">

<!-- MODEL PART-->

<script> var model = {
customer: "Student",
items: [{ bill:
"Electricity", status: false

}, {

bill: "Internet(Wi/fi)", status: false

}, {

bill: "Parking Charges",

status: false

}, { bill:
"Phone",
status: true

}, { bill: "House Tax",
status: true

}

] }

varbillpayApp = angular.module("billpayApp", []); billpayApp.controller("billpayctrl",

function($scope)


{ $scope.billpay = model;
$scope.dueBills = function() { var items
= $scope.billpay.items; var counter = 0;
items.forEach((item) => { if
(!item.status) { counter++;

}

})

return counter;

} $ scope.redFlag =
function() {

return $scope.dueBills() <= 2 ? "label-success" : "label-danger"; }

$scope.addBills = function(billName)

{ obj = { bill:
billName, status:
false } $

scope.billpay.items.push(obj);
```

```
} $
scope.removeBills = function(rmvBills) {
$scope.billpay.items.splice($scope.billpay.items.indexOf(rmvBills), 1);
}
});
</script>
</head>
<!-- HTML BODY PART-->
<body ng-controller="billpayctrl">
<div class="container">
<div class="page-header">
<h1>{{billpay.customer}}'s Bill's remained to Be Paid -
<span class="lable" ng-class="redFlag()" ng-hide="dueBills()==0">
{{dueBills()}}
</span>
</h1>
</div>
<h3><center><b>Add extra biller fields if any</center></b></h3>
<div class="panel">
<div class="input-group">
<input class="form-control" ng-model="billName" />
<span class="input-group-btn"> <button
class="btnbtn-danger" ngclick="
addBills(billName)">+ADD+</button>
</span>
</div>
<table class="table table-striped">
<thead>
<tr>
<th>Bill Name</th>
<th>Status</th>
<th>Status</th>
<th>Close</th>
</tr>
</thead>
<tbodyng-model="rmvBills">
```

```
<trng-repeat="item in billpay.items" ng-model="item">
<td>{{item.bill}}</td>
<td><input type="checkbox" ng-model="item.status" /></td>
<td>{{item.status}}</td>
<td>
<button type="button" ng-click="removeBills(item)">&times;</button>
</td>
</tr>
</tbody>
</table>
</div>
</div>
</div>
</body>
</html>
```

Output:




Bill Payment Record usi
file:///F:/todo_app.html
Student's Bill's remained to Be Paid -
Add extra biller fields if any
+ADD+
Bill Name
Status
Status
Close
Internet(Wi-fi)
true
Phone
true
House Tax
true

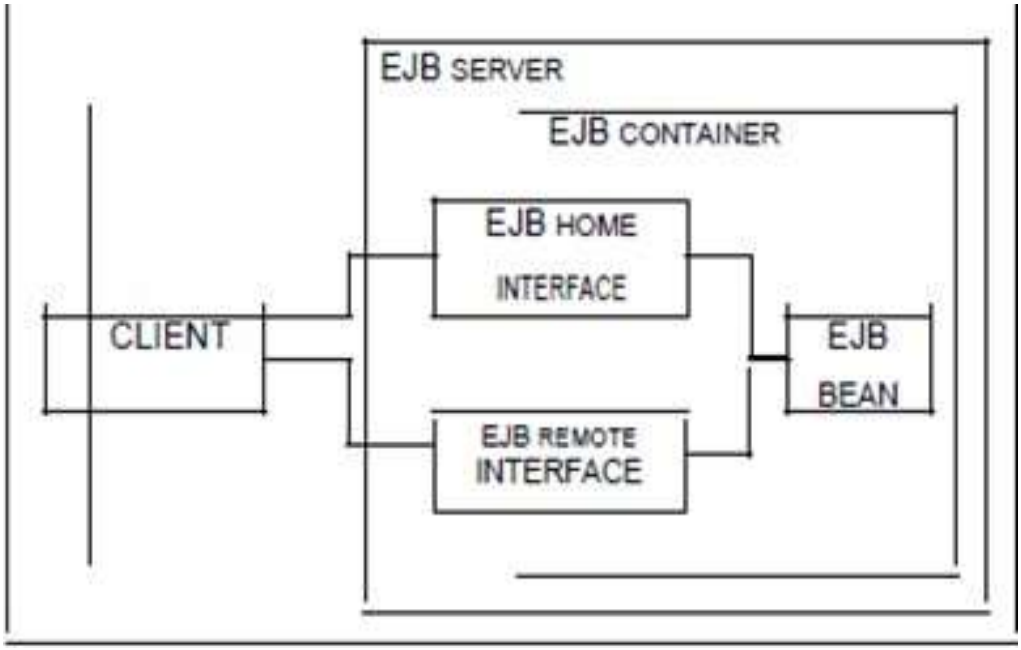
Assignment 10

Mayur Gorane
48 TE(A)

Title - Web application using EJB

Problem Statement - Design develop & deploy web appⁿ using EJB

Theory :



JavaBeans - J2EE application ~~containers~~ container contains the components that can be used by clients for executing the buissness logic These components are known as Enterprise Java beans (EJB).

Features of EJB's-

Some of features of an application server include following-

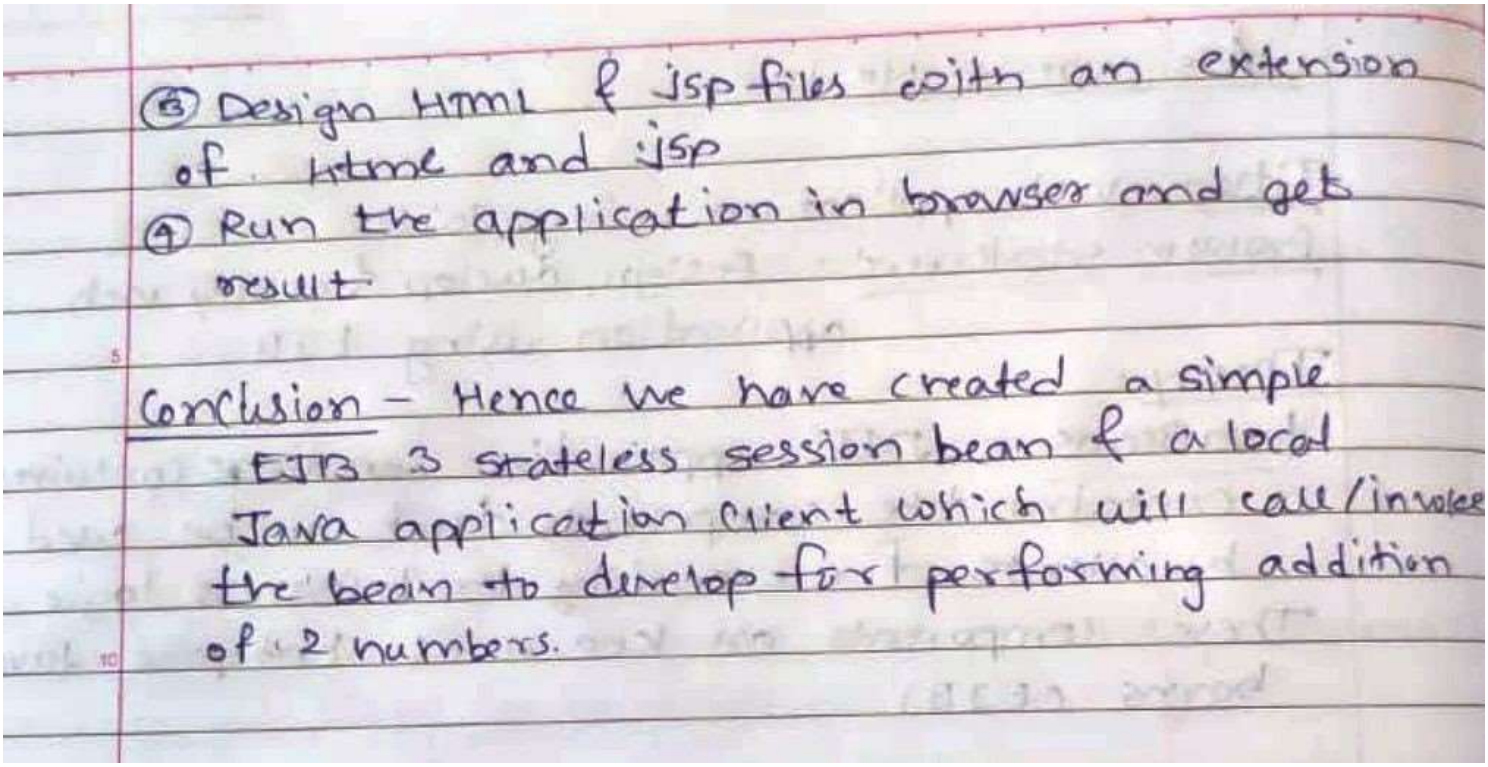
1) Client communication-
2) State Management
3) Transaction Management
4) Database connection Management
5) Asynchronous Messaging
6) Application server Administration.

Types of Enterprise Java Beans-

1. Session Beans
2. Entity Beans
3. Message driven Beans.

Design / Execution Steps-

1. Design EJB project
2. Start JBOSS & Deploy it on JBOSS Server.



PROGRAM CODE:

Create a new EJB Project:

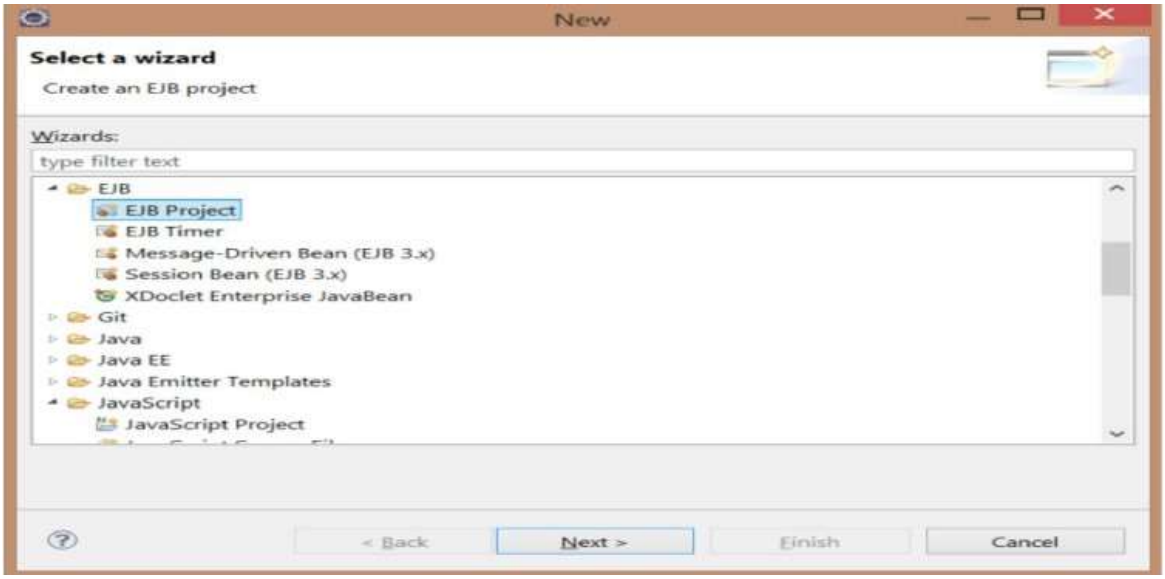
Open Eclipse IDE and create a new EJB project which can be done by clicking on,

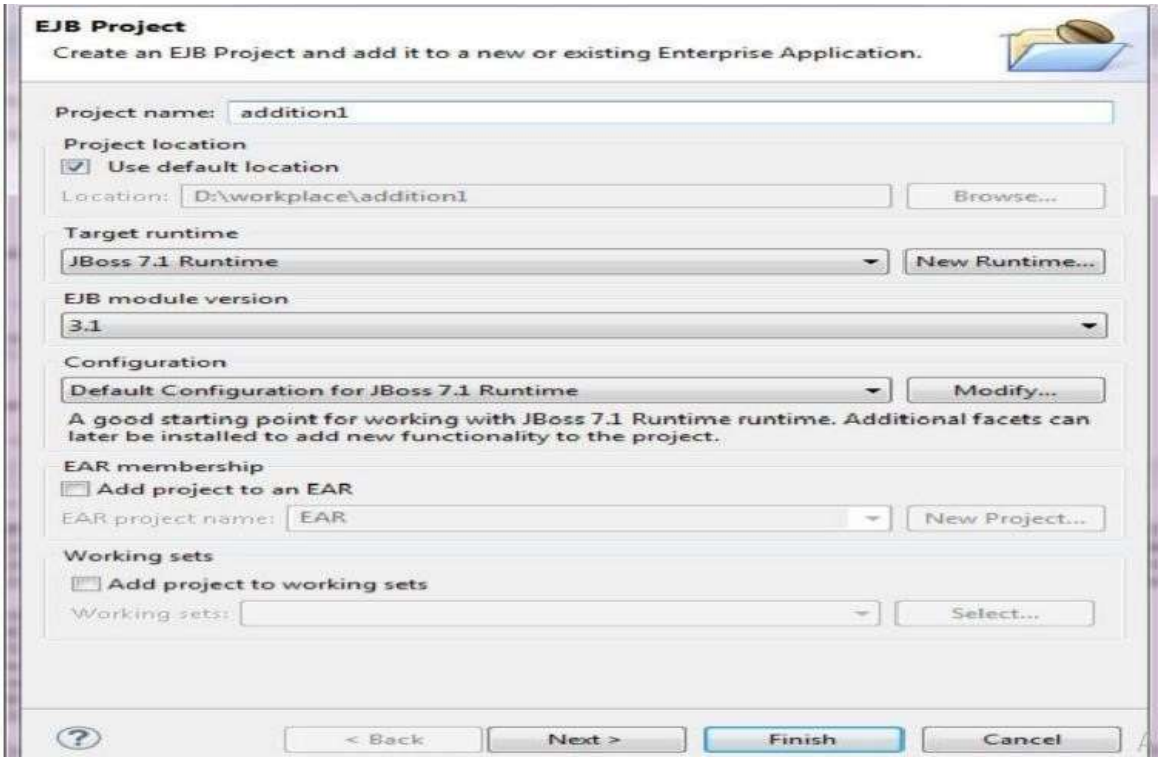
File menu -> New ->EJB Project

Step 1:

Create EJB project addition

Click File –> New –> Other –> EJB –> EJB Project –> Next

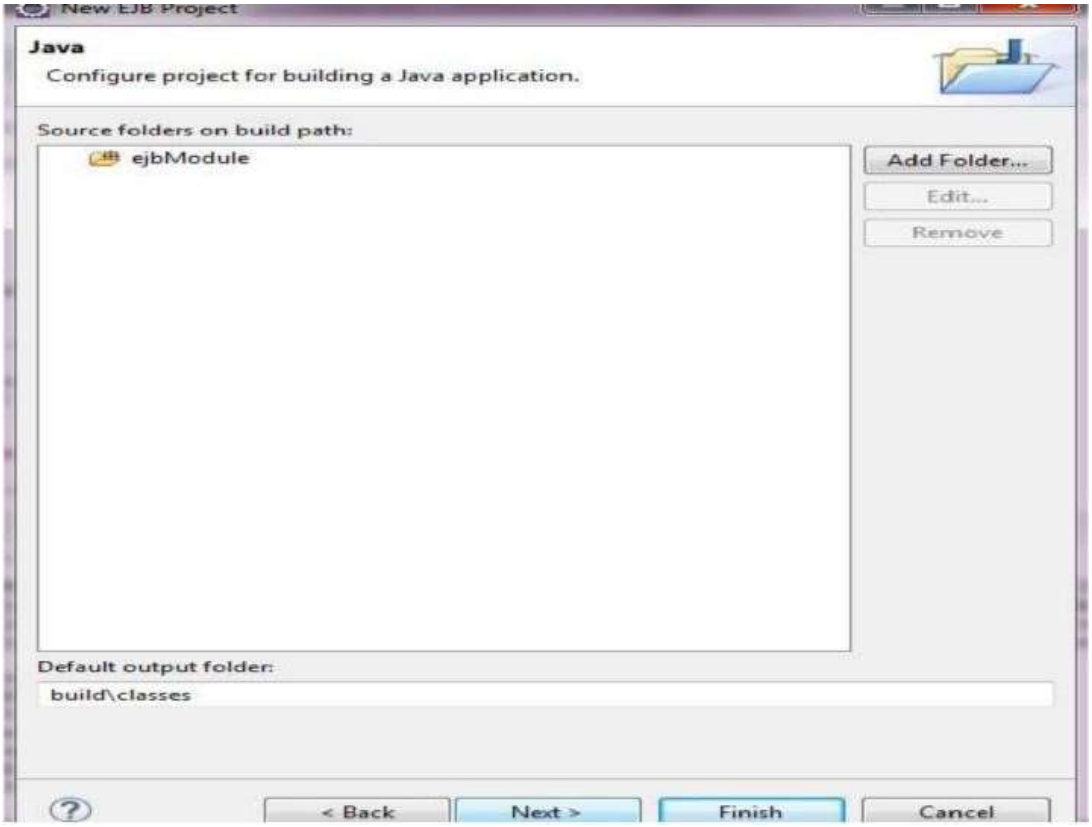




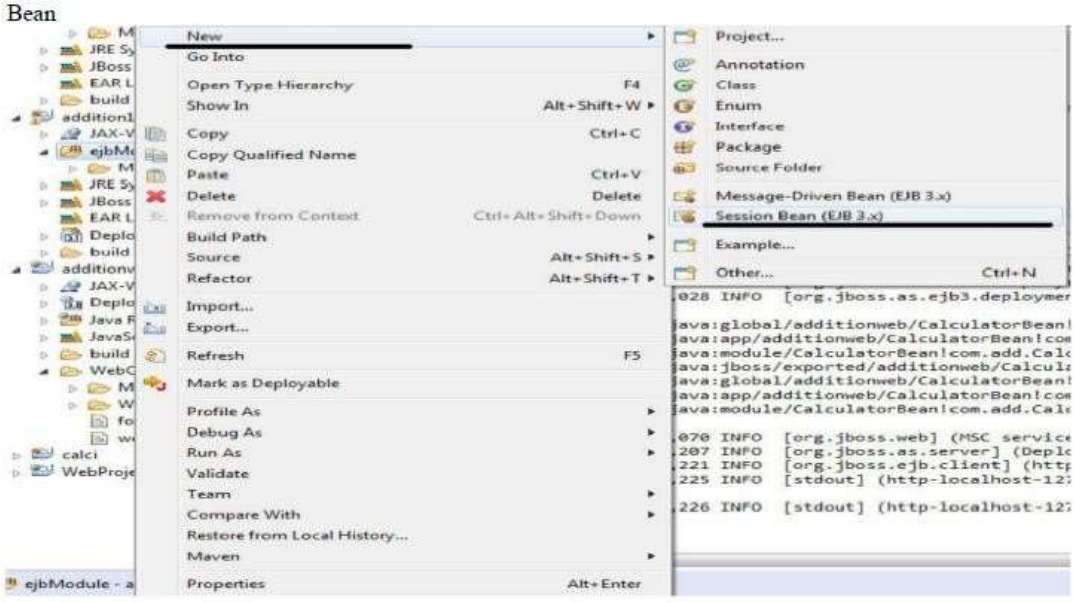
Step 2:

Now create Stateless session bean with its remote interface. Expand project –> expande ejbModule –> Right click Session Bean –> New –> Session


addition1
JAX-WS Web Services
ejbModule
META-INF
JRE System Library [jdk1.7.0_25]
JBoss 7.1 Runtime [JBoss 7.1 Runtime]
EAR Libraries
Deployment Descriptor: addition1
build


New EJB Project
Java
Configure project for building a Java application.
Source folders on build path:
ejbModule
Add Folder...
Edit...
Remove
Default output folder:
build\classes
< Back
Next >
Finish
Cancel

Bean


New
Go Into
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
Mark as Deployable
Profile As
Debug As
Run As
Validate
Team
Compare With
Restore from Local History...
Maven
Properties Alt+Enter
Project...
Annotation
Class
Enum
Interface
Package
Source Folder
Message-Driven Bean (EJB 3.x)
Session Bean (EJB 3.x)
Example...
Other... Ctrl+N

Write following code in CalculatorBean.java

CalculatorBean.java | CalculatorBeanRemote.java | webappadd.jsp

```
1  package com.add;
2
3  import javax.ejb.LocalBean;
5
6  /**
7   * Session Bean implementation class CalculatorBean
8   */
9  @Stateless
10 @LocalBean
11 public class CalculatorBean implements CalculatorBeanRemote {
12
13     /**
14      * Default constructor.
15      */
16     public CalculatorBean() {
17         // TODO Auto-generated constructor stub
18     }
19
20     public float add(float a, float b)
21     {
22         return a+b;

       }

   }
```

Write Following code in CalculatorBeanRemote .java

CalculatorBean.java | CalculatorBeanRemote.java

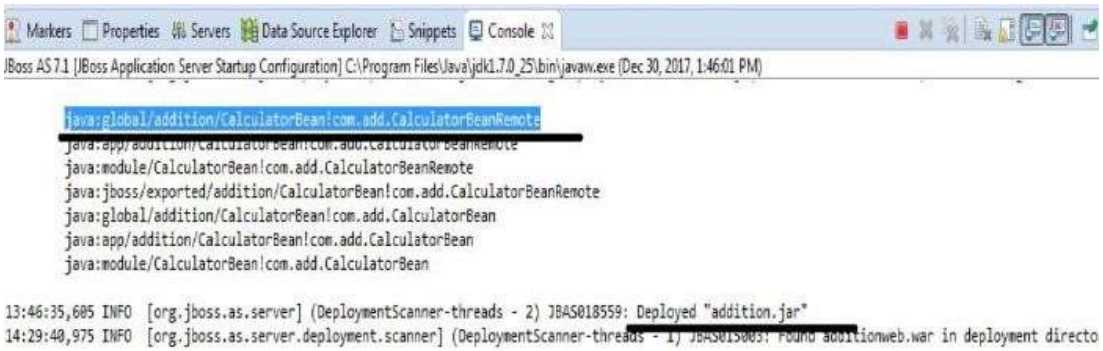
```
1  package com.add;
2
3  import javax.ejb.Remote;
4
5  @Remote
6  public interface CalculatorBeanRemote {
7
8      public float add(float a, float b);
9
10 }
11
```

Step 3:
Deploying the project :

Now we need to deploy the our EJB "addition" on server. Follow the steps mentioned bellow
to deploy this project on server.
Strat the server
Right click on "JBoss 7.1 Runtime Server" from Servers view and click on Start.

Step 4:
Now next step Go to Project-> addition -> right click -> run-> Run on server

Step 5:
After running the program you can see following message on console



Step 6:
Once this jar file is deployed to server now export EJB jar file save it in desktop -> Finish.

Step 7:
Now create another project

Write the following code in form.html

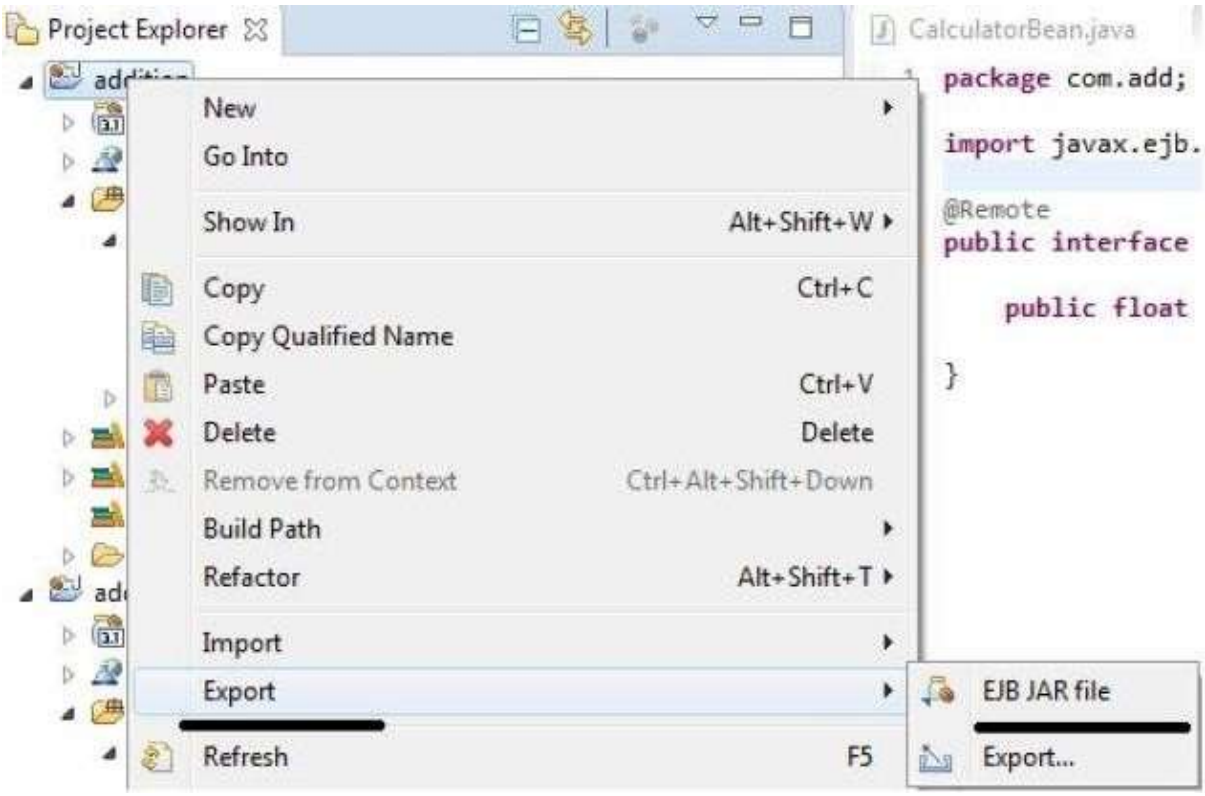
//form.html

```html
<html>
    <head>
        <title>Calculator</title>
    </head>

    <body bgcolor="blue">
        <h1>Calculator</h1>
        <hr>

        <form action="webappadd.jsp" method="POST">
    <p>Enter first value:
            <input type="text" name="num1" size="25"></p>
            <br>
        <p>Enter second value:
            <input type="text" name="num2" size="25"></p>
            <br>

            <b>Seclect your choice:</b><br>
    <input type="radio" name="group1" value ="add">Addition<br>

    <p>
            <input type="submit" value="Submit">
            <input type="reset" value="Reset"></p>

        </form>


    </body>
</html>
```

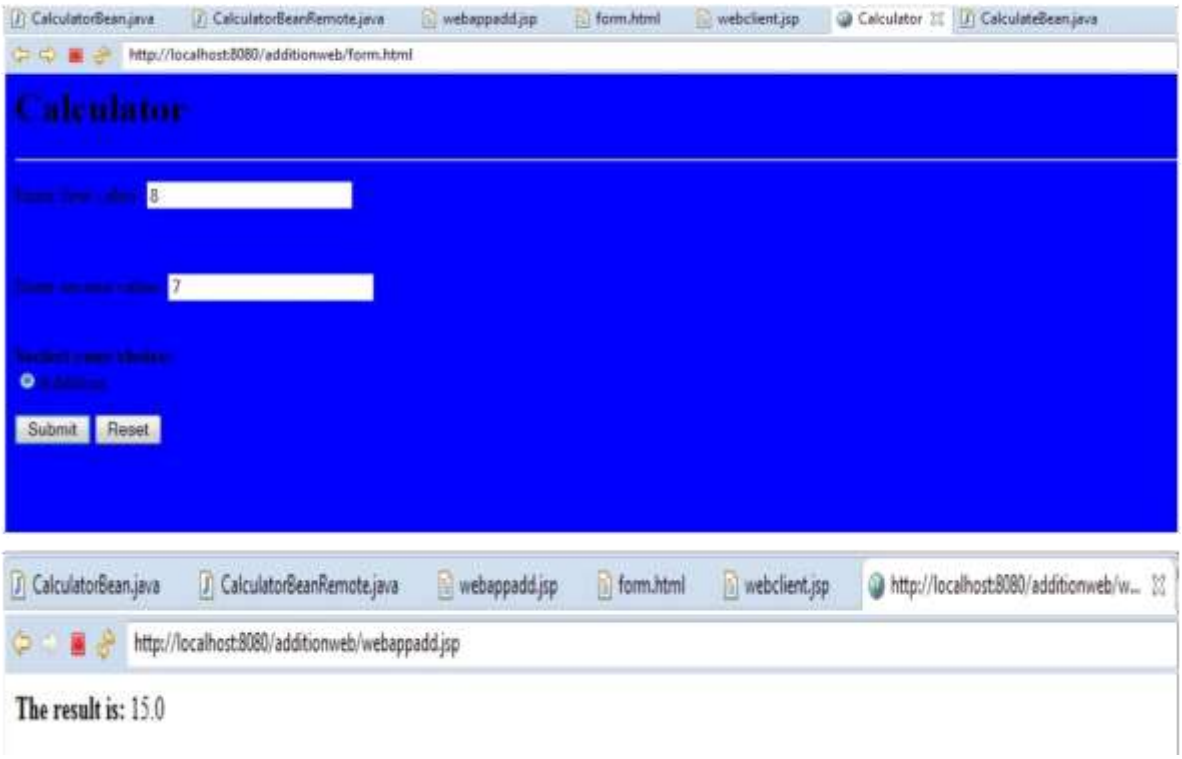
CalculatorBean.java | CalculatorBeanRemote.java | webappadd.jsp | form.html | webclient.jsp | http://localhost:8080/additionweb/w... | CalculateBean.java

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ page import="com.add.*, javax.naming.*, javax.ejb.EJB"%>

<%

float result=0;
//  CalculatorBeanRemote calculator=null;


    try {

        InitialContext ic = new InitialContext();


         CalculatorBeanRemote  calculator = (CalculatorBeanRemote) ic.lookup("java:global/addition/CalculatorBean!com.add.CalculatorBeanRemote");

         System.out.println("Loaded Calculator Bean");
//CalculatorBean



        String s1 = request.getParameter("num1");
          String s2 = request.getParameter("num2");
          String s3 = request.getParameter("group1");

System.out.println(s3);

        if ( s1 != null && s2 != null ) {
            Float num1  = new Float(s1);
             Float num2  = new Float(s2);

             if(s3.equals("add"))
                 result=calculator.add(num1.floatValue(),num2.floatValue());


    %>
    <p>

        <b>The result is:</b> <%= result %>
        <p>

        <%
            }
          }// end of try
          catch (Exception e) {
     e.printStackTrace ();
    //result = "Not valid";
   }

%>
```

Write following code inwebappadd.jsp


Project Explorer
CalculatorBean.java
package com.add;
import javax.ejb.
@Remote
public interface
public float
}
New
Go Into
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Refactor Alt+Shift+T
Import
Export
Refresh F5
EJB JAR file
Export...

OUTPUT:-


CalculatorBean.java
CalculatorBeanRemote.java
webappadd.jsp
form.html
webclient.jsp
Calculator
CalculateBean.java
http://localhost:8080/additionweb/form.html
Calculator
8
7
Submit
Reset
CalculatorBean.java
CalculatorBeanRemote.java
webappadd.jsp
form.html
webclient.jsp
http://localhost:8080/additionweb/w...
http://localhost:8080/additionweb/webappadd.jsp
The result is: 15.0