1a) Create a simple calculator application using servlet.

Input.html

```
<body>

 <form name=f1 method=get action="Cal" >
   Enter First Number <input type="text" name="t1" ><br>
   Enter Second Number<input type="text" name="t2" ><br>
   Select
   <input type="radio" name="r1" value="a">ADDTION
   <input type="radio" name="r1" value="s">SUBSTRACTION
   <input type="radio" name="r1" value="m">MULTIPLY
   <input type="radio" name="r1" value="d">DIVIDE
   <input type="submit" value="OK" >
  </form>
</body>
```

Cal.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Cal extends HttpServlet
{
      public void doGet(HttpServletRequest req,
HttpServletResponse res) throws ServletException,IOException
      {
            PrintWriter pw=res.getWriter();
            int a=Integer.parseInt(req.getParameter("t1"));
            int b=Integer.parseInt(req.getParameter("t2"));
            String c=req.getParameter("r1");

            if(c.equals("a"))
            {
             pw.println("The ans is "+(a+b));
            }
            else if(c.equals("s"))
            {
                pw.println("The ans is "+(a-b));
            }
           else if(c.equals("m"))
            {
                pw.println("The ans is "+(a*b));
            }
            else if(c.equals("d"))
            {
                  pw.println("The ans is "+(a/b));
            }
      }
}
```

1b) Create a servlet for a login page. If the username and password are correct then it says message "Hello <username>" else a message "login failed"

Input.html

```
<body>

    <form name=f1 method=get action="Login" >
       Enter UserId <input type="text" name="t1" ><br>
       Enter Password<input type="text" name="t2" ><br>
       <input type="submit" value="OK" >
    </form>
</body>
```

Login.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Login extends HttpServlet
{
      public void doGet(HttpServletRequest req,
HttpServletResponse res) throws ServletException,IOException
      {
            PrintWriter pw=res.getWriter();
            String u= req.getParameter("t1");
            String p= req.getParameter("t2");

            if(u.equals("tyit") && p.equals("java"))
            {
                  pw.println("Hello  "+u));
            }
            else
            {
                pw.println("Login failed");
             }
      }
}
```

1c)Create a registration servlet in Java using JDBC. Accept the details such as Username, Password, Email, and Country from the user using HTML Form and store the registration details in the database.

Input.html

```
<body>

   <form name=f1 method=get action="Register" >
      Enter UserId <input type="text" name="t1" ><br>
      Enter Password<input type="text" name="t2" ><br>
      Enter Emailid <input type="text" name="t3" ><br>
      Enter Country<input type="text" name="t4" ><br>

      <input type="submit" value="OK" >
   </form>
</body>
```

Register.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class Register extends HttpServlet
{
      public void doGet(HttpServletRequest req,
HttpServletResponse res) throws ServletException,IOException
      {
            PrintWriter pw=res.getWriter();
            String u= req.getParameter("t1");
            String p= req.getParameter("t2");
            String e= req.getParameter("t3");
            String c= req.getParameter("t4");

            try
            {
                  Class.forName("com.mysql.jdbc.Driver");
                  Connection
      con=DriverManager.getConnection("jdbc:mysql://localhost/d
      b1","root","abc");
            PreparedStatement    st=con.prepareStatement("insert
      into user values(?,?,?,?)");
            pst.setString(1,u);
            pst.setString(2,p);
            pst.setString(3,e);
            pst.setString(4,c);
            pst.execute();
            con.close();
            pw.println("Data inserted");
            }
            catch(Exception e1)
            {
                  pw.println(e1);
            }
      }
```

```
    }
```

On Mysql

```
Create database db1;

Use db1;

Create table user(username varchar(20), password varchar(20), email
varchar(20), country varchar(20));
```

2a) Using Request Dispatcher Interface create a Servlet which will validate the password entered by the user, if the user has entered "Servlet" as password, then he will be forwarded to Welcome Servlet else the user will stay on the index.html page and an error message will be displayed.

Input.html

```
    <body>

        <form name=f1 method=get action="Login" >
           Enter UserId <input type="text" name="t1" ><br>
           Enter Password<input type="text" name="t2" ><br>
           <input type="submit" value="OK" >
        </form>
    </body>
```

Login.java

```
      import java.io.*;
      import javax.servlet.*;
      import javax.servlet.http.*;
      public class Login extends HttpServlet
      {
            public void doGet(HttpServletRequest req,
      HttpServletResponse res) throws ServletException,IOException
            {
                  PrintWriter pw=res.getWriter();
                  String u= req.getParameter("t1");
                  String p= req.getParameter("t2");

                  if(u.equals("tyit") && p.equals("java"))
                  {
                        RequestDispatcher
rd1=req.getRequestDispatcher("Result");
                        rd1.forward(req,res);
                  }
                  else
                  {
                      RequestDispatcher
rd2=req.getRequestDispatcher("input.html”);
                        pw.println("Invalid user.Please try again");
                        rd2.include(req,res);
                   }
```

```
          }}
```

Result.java

```
      import java.io.*;
      import javax.servlet.*;
      import javax.servlet.http.*;
      public class Result extends HttpServlet
      {
            public void doGet(HttpServletRequest req,
      HttpServletResponse res) throws ServletException,IOException
            {
                  PrintWriter pw=res.getWriter();

                  pw.println("Welcome");


            }
    }
```

2b) Create a servlet that uses Cookies to store the number of times a user has visited servlet.

Input.html

```
    <body>

        <form name=f1 method=get action="Counter" >
           <input type="submit" value="OK" >
        </form>
    </body>
```

Counter.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Counter extends HttpServlet
{
     public void doGet(HttpServletRequest req, HttpServletResponse
res) throws ServletException,IOException
     {
           PrintWriter pw=res.getWriter();
           boolean f=false;
                Cookie c[]=req.getCookies();
                if(c!=null)
                {
                      for(int i=0;i<c.length;i++)
                      {
                            if(c[i].getName().equals("count"))
                            {
                                  int
n=Integer.parseInt(c[i].getValue());
                                  n=n+1;
                                  c[i].setValue(n+"");

                                  res.addCookie(c[i]);
```

```
                                    pw.println("Number od visits: "+n);
                                    f=true;
                                    break;
                              }
                        }
                  }
                  if(f==false)
                  {
                        Cookie c1=new Cookie("count","1");
                        res.addCookie(c1);
                        pw.println("Number of visits: 1");
                  }
}
    }
```

2c) Create a servlet demonstrating the use of session creation and destruction. Also check whether the user has visited this page first time or has visited earlier also using sessions.

Input.html

```
    <body>

        <form name=f1 method=get action="Session" >
           <input type="submit" value="OK" >
        </form>
    </body>
```

Session.java

```
      import java.io.*;
      import javax.servlet.*;
      import javax.servlet.http.*;
      public class Session extends HttpServlet
      {
            public void doGet(HttpServletRequest req,
      HttpServletResponse res) throws ServletException,IOException
            {
                  PrintWriter pw=res.getWriter();
                  HttpSession hs=req.getSession(true);

                  if(hs.isNew())
                  {
                        pw.println("First time visitor");
                  }
                  else
                  {
                        pw.println("Welcome back");
                  }
            }
      }
```

3a) Create a Servlet application to upload and download a file.

Upload file example

Input.html

```
<body>

<form action="Upload" method="post" enctype="multipart/form-data">

   Select File to Upload:<input type="file" name="t1" >

   <input type="submit" value="ok">

   </form>

</body>
```

Upload.java

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


@MultipartConfig
public class Upload extends HttpServlet
{
public void doPost(HttpServletRequest req,HttpServletResponse res) throws
ServletException, IOException
{

   PrintWriter pw = res.getWriter();

   Part f=req.getPart("t1");

   try
   {
      FileOutputStream os=new FileOutputStream(new File("d:/abc.txt"));
     is=f.getInputStream();
      int r;

      while ((r = is.read()) != -1) {
         os.write((char)r);
      }
      pw.println("file uploaded sucessfully...!!!");
   }
   catch(Exception e)
   {
      pw.print(e);
```

```
   }
}

}
```

In the above example the uploaded file will be stored in d drive with the name as abc.txt.

Download file example

Input.html

```
<body>

<form action="Download" >

   <input type="submit" value="ok">

   </form>

</body>
```

Download.java

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

import javax.servlet.http.*;

public class Download extends HttpServlet
{

      public void doGet(HttpServletRequest req, HttpServletResponse res)throws
ServletException, IOException
      {
            String filename="d:/abc.txt";
            FileInputStream is = new FileInputStream(filename);
            PrintWriter pw=res.getWriter();
            res.setHeader("Content-Disposition","attachment; filename=\"" + filename + "\"");

                  int i;

                    while ((i=is.read()) != -1) {
                      pw.write(i);
                    }

         }

}
```

The above program will download abc.txt file from d drive to the location that we specify.

3b) Develop Simple Servlet Question Answer Application using Database.

On MySQL

```
create database db1;
use db1;
create table quiz(question varchar(100),option1 varchar(20),option2 varchar(20),option3
varchar(20),option4 varchar(20),ans varchar(20));
insert into quiz values('capital of india','mumbai','delhi','chennai','banglore','delhi');
insert into quiz values('national animal','lion','tiger','elephant','cat','tiger');
insert into quiz values('national sports','cricket','hockey','football','tennis','hockey');

select * from quiz;
```

Input.html

```
<body>

<form action="mcq" >

   <input type="submit" value="ok">

   </form>

</body>
```

mcq.java

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

public class mcq extends HttpServlet {

   public void doGet(HttpServletRequest req,HttpServletResponse res) throws
ServletException, IOException
   {
      PrintWriter pw=res.getWriter();
      res.setContentType("text/html");

      try
      {
```

```
        Class.forName("com.mysql.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost/d1","root","abc");
        Statement st=con.createStatement();

        ResultSet rs=st.executeQuery("select * from quiz");

        pw.println("<form action=Result>");
        int i=1;
        while(rs.next())
        {
            pw.println(rs.getString("question"));
            pw.println("<br>");
            pw.println("<input type=radio name="+i+"
value="+rs.getString("option1")+">"+rs.getString("option1"));
            pw.println("<input type=radio name="+i+"
value="+rs.getString("option2")+">"+rs.getString("option2"));
            pw.println("<input type=radio name="+i+"
value="+rs.getString("option3")+">"+rs.getString("option3"));
            pw.println("<input type=radio name="+i+"
value="+rs.getString("option4")+">"+rs.getString("option4"));
            pw.println("<br>");
            i++;
        }
        pw.println("<input type=submit></form>");


      }
      catch(Exception e)
      {
           pw.println(e);
      }
   }
}
```

Result.java

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

public class Result extends HttpServlet {
```

```
   public void doGet(HttpServletRequest req,HttpServletResponse res) throws
ServletException, IOException
   {
      PrintWriter pw=res.getWriter();

      try
      {
         Class.forName("com.mysql.jdbc.Driver");
         Connection
con=DriverManager.getConnection("jdbc:mysql://localhost/d1","root","abc");
         Statement st=con.createStatement();

         ResultSet rs=st.executeQuery("select ans from quiz");


         int i=1,c=0;
         while(rs.next())
         {
           if(rs.getString("ans").equals(req.getParameter(i+"")))
           {
              c++;
           }
           i++;
         }
         pw.println(c);


      }
      catch(Exception e)
      {
           pw.println(e);
      }
   }
}
```

3c) Create simple Servlet application to demonstrate Non-Blocking Read Operation.

NonBlockingServlet.java

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

public class NonBlockingServlet extends HttpServlet
{
```

```
   AsyncContext ac=null;
   public void doGet(HttpServletRequest req, HttpServletResponse res)throws
ServletException, IOException
   {

      final PrintWriter pw = res.getWriter();
      try
      {

         pw.println("<h1>FileReader</h1>");


         ac.start(new Runnable()
         {
            String filename="d:/booklist.txt";
            FileInputStream is = new FileInputStream(filename);
            BufferedReader br = new BufferedReader(new InputStreamReader(is));
            String text;
            @Override
            public void run ()
            {
               try
               {
                  pw.println("Reading started....");
                  while((text=br.readLine())!=null)
                  {
                     pw.print("<br>"+text);


                  }
               }
               catch(Exception e)
               {
                  pw.println(e);
               }
               ac.complete();
            }
         });
      }
      catch(Exception e)
      {
         pw.println(e);
      }

   }

}
```

4a) Develop a simple JSP application to display values obtained from the use of intrinsic objects of various types.

input.html

```
<body>
    <form action="index.jsp">
      <input type="text" name="t1">
      <input type="submit">
    </form>
  </body>
```

index.jsp

```
<%
    String n=request.getParameter("t1");
    out.println(n);
    out.println(request.getMethod()+"<br/>");
    out.println(request.getRequestURL()+"<br/>");
    out.println(request.getQueryString()+"<br/>");

    response.setContentType("text/html");
    out.println(response.getCharacterEncoding()+"<br/>");
    out.println(response.getContentType()+"<br>");

    out.println(session.getId()+"<br/>");
    out.println(session.getCreationTime()+"<br/>");
    out.println(session.getLastAccessedTime()+"<br/>");

%>
```

4b) Develop a simple JSP application to pass values from one page to another with validations. (Name-txt, age-txt, email-txt, gender-radio button).

input.html

```
  <body>
    <form action="index.jsp">
      Enter Your Name<input type="text" name="name" ><br>
      Enter Your Age<input type="text" name="age" ><br>
      Enter E-mail<input type="text" name="email" ><br>
      Select Gender
      <input type="radio" name="gender" value="male">Male
      <input type="radio" name="gender" value="female">Female
      <input type="radio" name="gender" value="other">Other<br>

      <input type="submit" value="Submit Form">
     </form>
  </body>
```

validate.jsp

```
<body>

<jsp:useBean id="obj"  class="p1.Details"/ >

<jsp:setProperty name="obj" property="*"/>
<%
	if (obj.validate())
	{
%>
	<jsp:forward page="s.jsp">
		<jsp:param name="p1" value="<%= obj.getName() %>" />
	</jsp:forward>
<%
	}
	else
	{
%>
	<jsp:include page="input.html"/>
	<%= obj.getError() %>
<%
	}
%>
</body>
```

Details.java

```
package p1;

public class Details
{
	String name="", age="", email="", gender="", error="";

	public void setName(String n)
	{
		name=n;
	}
	public void setAge(String a)
	{
		age=a;
	}
	public void setEmail(String e)
	{
		email=e;
	}
	public void setGender(String g)
	{
		gender=g;
	}
```

```
	public void setError(String e)
	{
		error=e;
	}
	public String getName()
	{
		return name;
	}
	public String getAge()
	{
		return age;
	}
	public String getEmail()
	{
		return email;
	}
	public String getGender()
	{
		return gender;
	}
	public String getError()
	{
		return error;
	}
	public boolean validate()
	{
		boolean res=true;
		if(name.equals(""))
		{
			error=error+"<br>Enter First Name";
			res=false;
		}
		if(age.equals("") )
		{
			error=error+"<br>Age Invalid";
			res=false;
		}

		return res;
	}
}
```

s.jsp

```
<body>
Welcome, <%=request.getParameter(“p1”) %>
</body>
```

4c) Create a registration and login JSP application to register and authenticate the user based on username and password using JDBC.

On Mysql

```
create database db1;

use db1;

create table user(username varchar(20),password varchar(20),emailid varchar(100));

select * from user;
```

demo.html

```
<body>
        <a href="input.html">Register new user</a>
        <a href="input1.html">Login</a>
</body>
```

input.html

```
<body>
<form action="insert.jsp">

        User name<input type=text name=t1>
        Password<input type=password name=t2>
        Email ID<input type=text name=t3>

        <input type=submit value=ok>
</form>
</body>
```

insert.jsp

```
<body>

   <%@ page import="java.sql.*"%>
   <%! String u,p,e;%>

   <%

     u=request.getParameter("t1");
     p=request.getParameter("t2");
     e=request.getParameter("t3");

     try
         {
       Class.forName("com.mysql.jdbc.Driver");
       Connection con=DriverManager.getConnection("jdbc:mysql://localhost/db1","root","abc");
```

```
        PreparedStatement pst=con.prepareStatement("insert into user values(?,?,?)");
            pst.setString(1,u);
            pst.setString(2,p);
            pst.setString(3,e);
         pst.execute();
         con.close();
         out.println("Data inserted successfully");
        }
          catch(Exception e)
          {
        out.println(e);
          }
    %>
   </body>
```

input1.html

```
<body>
<form action="verify.jsp">

          User name<input type=text name=t1>
          Password<input type=password name=t2>


          <input type=submit value=Login>
</form>
</body>
```

verify.jsp

```
<%@ page import="java.sql.*"%>
    <%! String u,p;%>
   <%
          boolean f=false;
          u=request.getParameter("t1");
          p=request.getParameter("t2");
try
{
          Class.forName("com.mysql.jdbc.Driver");
          Connection con=DriverManager.getConnection("jdbc:mysql://localhost/db1","root","abc");
          Statement st=con.createStatement();
          ResultSet rs=st.executeQuery("select * from user");
          If(rs.next()==true)
          {
                    out.println("Valid user");
          }
          else
          {
                    out.println("Invalid user");

          }
```

```
}
catch(Exception e)
{
        out.println(e);
}

%>
```

5a) Create an html page with fields, eno, name, age, desg, salary. Now on submit this data to a JSP page which will update the employee table of database with matching eno.

On Mysql

```
create database db2;

use db2;

create table emp(eno int,name varchar(20),age int,designation varchar(20),salary int);

insert into emp values(1,'abc',35,'manager',30000);

insert into emp values(2,'xyz',28,'TL',23000);

insert into emp values(3,'pqr',40,'director',60000);

select * from emp;
```

input.html

```
<body>
<form action="update.jsp">

        Emp no<input type=text name=t1>
        Name<input type=text name=t2>
        Age<input type=text name=t3>
        Designation<input type=text name=t4>
        Salary<input type=text name=t5>

        <input type=submit value=ok>
</form>
</body>
```

update.jsp

```
<body>
   <%@ page import="java.sql.*"%>
   <%! int no,age,salary;String name,designation;%>
<%
     no=Integer.parseInt(request.getParameter("t1"));
     name=request.getParameter("t2");
        age=Integer.parseInt(request.getParameter("t3"));
        designation=request.getParameter("t4");
```

```
        salary=Integer.parseInt(request.getParameter("t5"));

      try
          {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost/db2","root","abc");
        PreparedStatement pst=con.prepareStatement("update emp set
name=?,age=?,designation=?,salary=? where eno=?");
            pst.setString(1,name);
            pst.setInt(2,age);
            pst.setString(3,designation);
            pst.setInt(4,salary);
            pst.setInt(5,no);
        pst.execute();
        con.close();
        out.println("Data updated ");
       }
         catch(Exception e)
         {
        out.println(e);
         }
   %>
  </body>
```

5b) Create a JSP page to demonstrate the use of Expression language.

input.html

```
<body>
      <form action="demo.jsp">
        Enter name <input type="text" name="t1">
         <input type="submit">
      </form>
</body>
```

demo.jsp

```
<body>

          Welcome, ${param.t1}

          The sum is ${1+2}

          <jsp:setProperty name="obj" property="first" value="${param.t1}" />
</body>
```

5b) Create a JSP application to demonstrate the use of JSTL.

```
<body>
      <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

       <c:out value="${'Welcome to javaTpoint'}"/>

       <c:set var="a" value="11"/>
       <c:out value="${a}"/>

        <c:remove var = "a"/>

       After remove <c:out value="${a}"/>

       <c:set var = "b"  value = "5"/>

        <c:if test = "${b%2==0}">
          <p>Even</p>
        </c:if>

      <c:choose>

       <c:when test = "${b%2==0}">
         Even
       </c:when>

       <c:otherwise>
         Odd
       </c:otherwise>
```

```
    </c:choose>

     <%--  <c:redirect url = "index.jsp"/>--%>
  </body>
```

6a) Create a Currency Converter application using EJB.

Input.html

```
<body>
<form action="convert" >

 Enter Amount <input type="text" name="t1">
 Select conversion
<input type="radio" name="r1" value="r2d" >Rupees to Dollar
<input type="radio" name="r1" value="d2r" >Dollar to Rupees
      <input type="submit" value="Convert" >

</form>
</body>
```

convert.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.ejb.*;
import p1.*;

public class convert extends HttpServlet
{

@EJB
Cbean obj;

public void doGet(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException
{

      PrintWriter pw = res.getWriter();
      double a = Double.parseDouble(req.getParameter("t1"));
      String r=req.getParameter("r1");

      if(r.equals("r2d"))
      {
            pw.println("The converted amt is: "+obj.r_d(a));
      }
      else if(r.equals("d2r"))
      {
            pw.println("The converted amt is: "+obj.d_r(a));
      }
}
}
```

Cbean.java

```
package p1;
import javax.ejb.*;

@Stateless
@LocalBean
public class Cbean
{

      public double r_d(double r)
      {
            return r/65.65;
      }
      public double d_r(double d)
      {
            return d*65.65;
      }
}
```

6b) Develop a Simple Room Reservation System Application Using EJB.

On MySQL

```
create database db1;

use db1;

create table room(rno varchar(10) , type varchar(20), charges int,
custname varchar(20), mob varchar(20) , status varchar(10));

insert into room values('1001','Delux',5000,'','','Not Booked');
insert into room values('1002','Super Delux',7000,'','','Not
Booked');
insert into room values('1003','Executive',9500,'','','Not Booked');
insert into room values('2001','Delux',5000,'','','Not Booked');
insert into room values('2002','Super Delux',7000,'','','Not
Booked');
insert into room values('2003','Executive',9500,'','','Not Booked');
```

input.html

```
<body>
<form action="roomservlet" >

 Name<input type="text" name="t1" ><br>
 Mobile No.<input type="text" name="t2" ><br>

 Select a room Type
<input type="radio" name="r1" value="Delux">Delux
<input type="radio" name="r1" value="Super Delux">Super Delux
<input type="radio" name="r1" value="Executive">Executive<br>
```

```
<input type="submit" value="Book">
</form>
</body>
```

roomservlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.ejb.*;
import p1.*;

public class roomservlet extends HttpServlet
{

@EJB
roombean obj;

public void doGet(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException
{

      PrintWriter pw = response.getWriter();

      String n=request.getParameter("t1");
      String m=request.getParameter("t2");
      String t=request.getParameter("r1");

      String msg = obj.book(t,n,m);
      pw.println(msg);
}
}
```

roombean.java

```
package p1;
import javax.ejb.*;
import java.sql.*;

@Stateless
public class roombean
{

      public String book(String t, String n, String m)
      {
      String msg="";

      try
      {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/db1","root","abc
");
            PreparedStatement pst = con.prepareStatement("select *
from room where Type=? and status=?");
            pst.setString(1,t);
            pst.setString(2,"Not Booked");
            ResultSet rs= pst.executeQuery();
            if(rs.next())
```

```
            {
                  String rno=rs.getString("rno");
                  PreparedStatement pst1 =
con.prepareStatement("update room set custname=?,mob=?,status=?
where rno=? ");
                  pst1.setString(1,n);
                  pst1.setString(2,m);
                  pst1.setString(3,"Booked");
                  pst1.setString(4,rno);

                  pst1.execute();

                  msg = "Room "+rno+ " Booked <br> Charges =
"+rs.getString("charges");
            }
            else
            {
                  msg = " room currently not available";
            }
      }
      catch(Exception e)
      {
            System.out.println(e);
      }
      return msg;
      }
}
```

6c) Develop simple shopping cart application using EJB [Stateful Session Bean].

On Netbeans

File→ new → project →Java EE→Enterprise Application→Next→next→finish

This will create 3 projects. (Assuming the name of the project is cart)

1) cart
2) cart-ejb
3) cart-war

Right click on cart-ejb→new→others→Enterprise java beans→session bean →enter name as CartBean and package as p1, check stateful and local→finish

This will add 2 files:

CartBean.java: this contains class that implements the interface.
CartBeanLocal.java: this contains interface

Add some methods that is to be used in bean

Add a servlet in cart-war project named CartAccess.java

Inside doget method right click and click insert code and then call enterprise beans.

Select cart-ejb and select the Cart bean and click ok.

CartBeanLocal.java (interface)

```
package p1;

import javax.ejb.Local;
import java.util.*;

@Local
public interface CartBeanLocal
{

 public void addBook(String title);
 public void removeBook(String title) throws Exception;
 public ArrayList<String> getBooks();


}
```

CartBean.java

```
package p1;

import javax.ejb.Stateful;
import java.util.*;

@Stateful
public class CartBean implements CartBeanLocal
{
    ArrayList <String> b=new ArrayList();

    public void addBook(String title)
    {
        b.add(title);
    }
    public void removeBook(String title) throws Exception
    {
        boolean result = b.remove(title);
        if (result == false)
        {
            throw new Exception(title + " not in cart.");
        }
    }
    public ArrayList<String> getBooks()
    {
        return b;
    }

}
```

CartAccess.java (Servlet)

```
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
import p1.CartBeanLocal;
import java.util.*;

public class CartAccess extends HttpServlet {
    CartBeanLocal cartBean = lookupCartBeanLocal();

    public void doGet(HttpServletRequest req,HttpServletResponse
res) throws ServletException, IOException
    {
        PrintWriter pw=res.getWriter();
        cartBean.addBook("java");
        cartBean.addBook("php");
        cartBean.addBook("sql");
        pw.println("book added");
        ArrayList<String> b=cartBean.getBooks();
        for(String s : b)
                pw.println(s +"<br />");
    }

    private CartBeanLocal lookupCartBeanLocal() {
        try {
            Context c = new InitialContext();
            return (CartBeanLocal) c.lookup("java:global/cart/cart-
ejb/CartBean!p1.CartBeanLocal");
        } catch (NamingException ne) {
            Logger.getLogger(getClass().getName()).log(Level.SEVERE,
"exception caught", ne);
            throw new RuntimeException(ne);
        }
    }
}
```

7a) Develop simple EJB application to demonstrate Servlet Hit count using Singleton Session Beans.

input.html

```
<form action="CountServlet">
      <input type=submit value=ok>
</form>
```

CountServlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.ejb.*;
import p1.*;

public class CountServlet extends HttpServlet {

   @EJB
   CountBean obj;

   @Override
   protected void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException,
IOException
   {

         PrintWriter pw = res.getWriter();

         pw.println("<h1>Number of times this servlet is accessed: " + obj.count() + "</h1>");


   }
}
```

CountBean.java

```
package p1;

import javax.ejb.Singleton;

@Singleton
public class CountBean {
   private int i=0;

   public int count() {
          i=i+1;
      return i;
   }
}
```

7c) Create a marks entry application using EJB

on MySQL

```
Create database db1;

use db1;

create table marks(name varchar(20), m1 int, m2 int,m3 int);
```

input.html

```
<body>
<form action="markservlet" >

 Name<input type="text" name="t1" ><br>
 Marks 1<input type="text" name="t2" ><br>
 Marks 2<input type="text" name="t3" ><br>
 Marks 3<input type="text" name="t4" ><br>

<input type="submit" value="ok">
</form>
</body>
```

markservlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.ejb.*;
import p1.*;

public class markservlet extends HttpServlet
{

@EJB
markbean obj;

public void doGet(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException
{

      PrintWriter pw = res.getWriter();

      String n=request.getParameter("t1");
      int m1=Integer.parseInt(req.getParameter("t2"));
      int m2=Integer.parseInt(req.getParameter("t3"));
      int m3=Integer.parseInt(req.getParameter("t4"));

      obj.storemarks(n,m1,m2,m3);
      pw.println("marks entered");
}
}
```

markbean.java

```
package p1;

import javax.ejb.*;

@Stateless
public class markbean
{
      public String  storemarks(String n,int m1,int m2,int m3)
      {
      try
      {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/db1","root","roo
t");
            PreparedStatement pst1 = con.prepareStatement("insert
into marks values(?,?,?,?) ");
                  pst1.setString(1,n);
                  pst1.setInt(2,m1);
                  pst1.setInt(3,m2);
                  pst1.setInt(4,m3);

                  pst1.execute();

                  con.close();

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

      }
}
```

8a) Develop a simple Inventory Application Using JPA.
8b) Develop a Guestbook Application Using JPA.
8c) Create simple JPA application to store and retrieve Book details.

On Mysql

```
Create database j1;

use j1;

create table books(bno int primary key auto_increment ,name
varchar(20), author varchar(20));
```

input.html

```
<body>

<form action="index.jsp" >

 Book Name<input type="text" name="t1" >
 Author<input type="text" name="t2" >

<input type="submit" value="ok">
</form>
</body>
```

index.jsp

```
<body>

<%@page import="java.util.*,javax.persistence.*,p1.*" %>

<%
EntityManagerFactory f =
Persistence.createEntityManagerFactory("jpa2PU");

EntityManager m =f.createEntityManager();

String n = request.getParameter("t1");
String a = request.getParameter("t2");
Books b = new Books();

b.setName(guest);
b.setAuthor(msg);

EntityTransaction t = m.getTransaction();
t.begin();
m.persist(b);
t.commit();

try {
        List<Books>v = m.createQuery("SELECT b from Books
b").getResultList();
         Iterator i = v.iterator();
```

```
          while (i.hasNext()) {
              Books obj1 = (Books) i.next();
              out.println(obj1.getBno());
              out.println(obj1.getName());
              out.println(obj1.getAuthor());
              out.println("<br>");
          }

 } catch (RuntimeException e)
 {
      out.println(e);
 }
%>
```

Books.java

```
package p1;

import java.io.*;
import javax.persistence.*;
import javax.validation.constraints.*;



@Entity
@Table(name = "books")

public class Books implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "bno")
    private Integer vno;
    @Size(max = 20)
    @Column(name = "name")
    private String name;
    @Size(max = 300)
    @Column(name = "author")
    private String author;

    public Books() {
    }

    public Books(Integer bno) {
        this.bno = bno;
    }

    public Integer getBno() {
        return bno;
    }

    public void setBno(Integer bno) {
        this.bno = bno;
    }

    public String getName() {
```

```
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAuthor() {
        return author;
    }

    public void setAuthor(String author) {
        this.author = author;
    }

}
```

The above file is created automatically by performing the following steps:

Project→right click→New→Entity classes from database

In Data source→ select new database source→ specify any name in JNDI name text box and in Database connection select New Database connection.

In the new window select Driver as MySql (Connector/JDriver) as we are using MySQL database.

In the new window specify database name created above as **j1 and** password for mysql as abc (depends what is there in your machine). Click next and finish.

Once finish is clicked you will come back to the previous window, it will show the list of tables present inside database in **available tables list.**

Select the table you want to connect to and click on add. This will add the table to **selected tables list. Click next.**

In the next window **provide package name** as something (example p1) and **uncheck Generate named query annotation for persistent fields and uncheck Generate JAXB annotation.** Click next and finish.

9a) Develop a JPA Application to demonstrate use of ORM associations.
9b) Develop a Hibernate application to store Feedback of Website Visitor in MySQL Database.

Input.html

```
<body>
    <form action=" connect.jsp">
       Visitor no<input type="text" name="t1">
       Visitor name<input type="text" name="t2">
       Visitor feedback<input type="text" name="t3">
       <input type="submit" value="ok">
    </form>
  </body>
```

connect.jsp

```
<body>
<%@page  import="org.hibernate.* , org.hibernate.cfg.*,java.util.*, p1.*" %>
<%
        AnnotationConfiguration cf=new AnnotationConfiguration();
        SessionFactory sf=cf.configure().buildSessionFactory();
        Session s=sf.openSession();
        Transaction t=s.beginTransaction();
        int x=Integer.parseInt(request.getParameter("t1"));
        String y=request.getParameter("t2");
        String z=request.getParameter("t3");
        visitor obj=new visitor();
        obj.setVno(x);
        obj.setVname(y);
        obj.setFeedback(z);
        s.save(obj);
```

```
        t.commit();

        out.println("data inserted");

try {
Query q= s.createQuery("from visitor");
   List<visitor>v=q.list();
         Iterator i = v.iterator();
         while (i.hasNext()) {
             visitor obj1 = (visitor) i.next();
             out.println(obj1.getVno());
             out.println(obj1.getVname());
             out.println(obj1.getFeedback());
         }

 } catch (RuntimeException e)
 {
     out.println(e);
 }


    %>

 </body>
```

Visitor.java

```
package p1;
import javax.persistence.*;
@Entity
@Table(name="visitor")
public class visitor
{
  int no;
  String n,f;
  public void setVno(int a)
  {
    no=a;
  }
  @Id
  @Column(name="vno")
  public int getVno()
  {
    return no;
  }
   public void setVname(String b)
  {
    n=b;
  }
   @Column(name="vname")
  public String getVname()
```

```
   {
      return n;
   }
   public void setFeedback(String c)
   {
      f=c;
   }
   @Column(name="feedback")
   public String getFeedback()
   {
      return f;
   }
}
```

hibernate.cfg.xml (This file is created automatically)

```
<hibernate-configuration>

  <session-factory>

    <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

    <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>

    <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/dbr2</property>

    <property name="hibernate.connection.username">root</property>

    <property name="hibernate.connection.password">abc</property>


    <mapping class="p1.visitor"/>                    //this line is required if mapping is done in
.java file
```

OR

```
    <mapping resource="visitor.hbm.xml"/>        //this line is required if mapping is done in
.hbm.xml file


  </session-factory>

</hibernate-configuration>
```

visitor.hbm.xml

```
<class name="p1.visitor" table="visitor">

     <id name="vno" column="vno">

       <generator class="assigned"/>
```

```
        </id>
        <property name="vname" column="vname"/>
        <property name="feedback" column="feedback"/>
    </class>
```

Q) Develop a Three page web application site using any two or three Java EE Technologies.

Any program of EJB and JDBC or any program of Hibernate.

Q) Write a JSP code to accept an employee id from user and delete the records of that employee. If the employee id does not exist, it should redirect to page with message “No such employee ID” else it should redirect to another page with the message “Employee record deleted successfully”.

input.html

```
<body>
<form action="delete.jsp">
        Emp no<input type=text name=t1>
        <input type=submit value=ok>
</form>
</body>
```

update.jsp

```
<body>
    <%@ page import="java.sql.*"%>
    <%! int no,age,salary;String name,designation;%>
    <%
      no=Integer.parseInt(request.getParameter("t1");
 try
        {
        Class.forName("com.mysql.jdbc.Driver");
```

```
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost/db1","root","abc");

        PreparedStatement pst=con.prepareStatement("delete from emp where eno=?");
          pst.setInt(1,no);

       int a= pst.executeUpdate();

        con.close();

        if(a==0)

%>

        <jsp:forward page=”a.jsp”/>

<%

}

else

{

%>

<jsp:forward page=”s.jsp”/>

<%

          }

        catch(Exception e)

        {

        out.println(e);

        }

    %>

  </body>
```

s.jsp

```
<body>

Employee deleted successfully

</body>
```

a.jsp

```
<body>

No such employee  </body>
```