Lab Manual

of

Advanced JAVA Technology

Subject Code: 3IT41

Academic Year 2020-21

Name of Student : Jaimin M. Shimpi

Id. No. : 18IT441

Batch : B 10A

Faculty Name : Dr. Vatsal H. Shah

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

Birla Vishwakarma Mahavidhyalaya Engineering College,

Vallabh Vidhyanagar-388120

Birla Vishvakarma Mahavidyalaya Engineering College

Information Technology Department

2020-2021

CERTIFICATE

This is to certify that Mr. / ~~Mrs~~. **Jaimin M. Shimpi**

of ~~Class~~/Sem. **6th** Id Number: **18IT441** has satisfactorily completed his/~~her~~ term work in Month of **April** for the term ending in 2020/2021, Number of Practical certified **17** out of **17** in the subject of **3IT41: ADVANCED JAVA TECHNOLOGY**.

Date: 28 / 04 / 2021

Signature of Teacher

INDEX

Sr. No.	Date	Practical	Pg. No.	Sign
1.		W.A.P. to make Calculator Application using Swing.	1	
2.		Create a database college in MySQL. Inside this database create a table student with id and name as fields. Write a JDBC program to get all this data from student table in college database.	5	
3.		Do the above program for insert ID and NAME into student table in college database.	7	
4.		Write down java networking demo programs for two way communication (Simple chat)	9	
5.		Write down simple Program in servlet and show the web.xml configuration for the same.	11	
6.		Create simple html form which contain username and password and submit to servlet page and display the same.	13	
7.		Create html form which contains the following field. i. Name (Text field) ii. Surname (Text Field) iii. Gender (Radio Button) iv. Sports (Check Box...like cricket, tennis, soccer…) v. Feedback (Text area) vi. And submit this page to servlet page whichdisplay all of above.	15	
8.		Write the servlet program which can get the username and password from html and if username is ADMIN and password is BVM then send Redirect to valid user page, else show not valid user.	18	

9.		**Write the simple session program which display new or old session with the session ID, Creation time, Last Access Time and session time out.**	**20**	
10.		**Write down the simple JSP program which displays prime numbers between 1 to 100.**	**23**	
11.		**Declare addition and multiplication methods in declarative tag. Call addition method using scriptlet and multiplication method through expression.**	**24**	
12.		**Create html page which can get the multiple sports selection from the user and call jsp page to display the same.**	**25**	
13.		**Do the above program by using JSTL core library functions.**	**27**	
14.		**Write down a simple Program in JSF by using Managed Bean.**	**29**	
15.		**Write down a JSF Page Which Consist of Basic HTML Tags.**	**30**	
16.		**Write down a simple program in spring.**	**31**	
17.		**Create application in hibernate which will create a object of contact class and set the id, first name, last name and email fields. Finally save the whole object in My SQL Database. (Four types of files are required. One for core java class with setter and getter methods and one for main application. Remaining two are the configuration files)**	**32**	

Practical 1

Aim: W.A.P. to make Calculator Application using Swing.

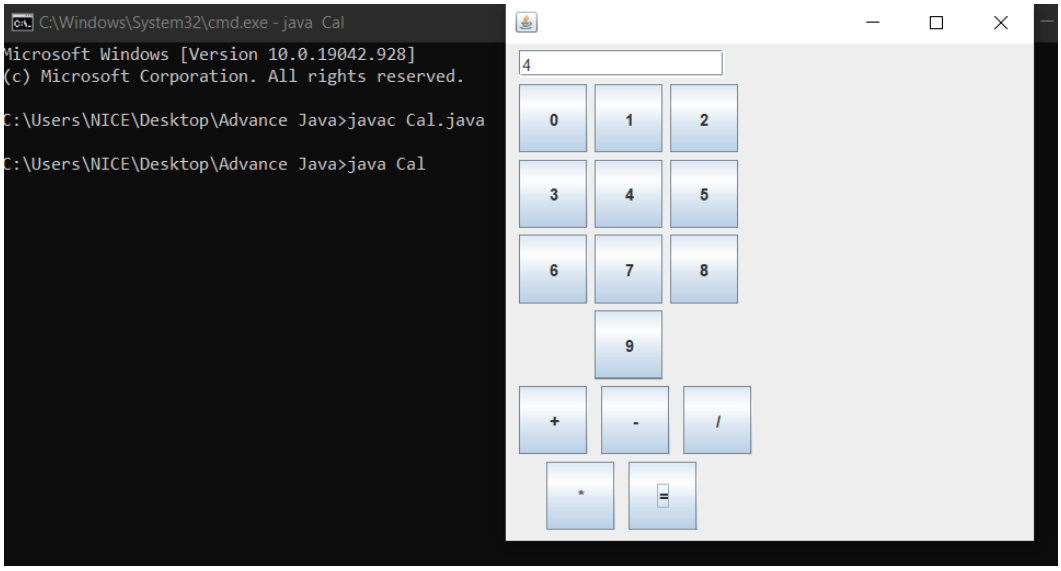
Code:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class Cal extends JFrame implements ActionListener
{
	JTextField t1;
	int a,b,c;
	String msg="",s,s1,s2,s3,s4;
	JButton eq,sum,sub,mul,div;
	JButton n[]=new JButton[10];
	Cal()
	{
		Container c=getContentPane();
		c.setLayout(new FlowLayout());
		this.setLayout(null);
		sum=new JButton("+");
		div=new JButton("/");
		mul=new JButton("*");
		eq=new JButton("=");
		sub=new JButton("-");
		t1=new JTextField(20);
		for(int i=0;i<10;i++)
		{
			n[i]=new JButton(i+"");
			n[i].setActionCommand(i+"");
			add(n[i]);
			n[i].addActionListener(this);
		}
		for(int i=0;i<3;i++)
			n[i].setBounds(10+55*i,30,50,50);
		for(int i=3;i<6;i++)
			n[i].setBounds(10+55*(i-3),85,50,50);
		for(int i=6;i<9;i++)
			n[i].setBounds(10+55*(i-6),140,50,50);
		n[9].setBounds(10+55,195,50,50);

		t1.setBounds(10,5,150,20);
		sum.setBounds(10,250,50,50);
		sub.setBounds(70,250,50,50);
```

```
                div.setBounds(130,250,50,50);
                mul.setBounds(30,305,50,50);
                eq.setBounds(90,305,50,50);
                add(eq);
                add(sum);
                add(sub);
                add(mul);
                add(div);
                add(t1);
                sum.addActionListener(this);
                sub.addActionListener(this);
                mul.addActionListener(this);
                div.addActionListener(this);
                eq.addActionListener(this);
        }

public void actionPerformed(ActionEvent e)
 {
         s=e.getActionCommand();
        if(s.equals("0")||s.equals("1")||s.equals("2")||s.equals("3")||s.equals("4")||s.equals("5")||s.equa
        ls("6")||s.equals("7")||s.equals("8")||s.equals("9")||s.equals("0"))
        {
                s1=t1.getText()+s;
                t1.setText(s1);
        }
        if(s.equals("+"))
        {
                s2=t1.getText();
                t1.setText("");
                s3="+";
        }
        if(s.equals("-"))
        {
        s2=t1.getText();
        t1.setText("");
        s3="-";
        }
        if(s.equals("*"))
        {
                s2=t1.getText();
                t1.setText("");
                s3="*";
        }
        if(s.equals("/"))
        {
```

```
                s2=t1.getText();
                t1.setText("");
                s3="/";
        }
        if(s.equals("="))
        {
                s4=t1.getText();
                a=Integer.parseInt(s2);
                b=Integer.parseInt(s4);
        if(s3.equals("+"))
                c=a+b;
        if(s3.equals("-"))
                c=a-b;
        if(s3.equals("*"))
                c=a*b;
        if(s3.equals("/"))
                c=a/b;
        t1.setText(String.valueOf(c));
        }
 if(s.equals("Clear"))
 {
  t1.setText("");
 }
}
        public static void main(String args[])
        {
                Cal a1=new Cal();
                a1.setSize(400,400);
                a1.setVisible(true);
        }
}
```

Output:


C:\Windows\System32\cmd.exe - java Cal
Microsoft Windows [Version 10.0.19042.928]
(c) Microsoft Corporation. All rights reserved.
C:\Users\NICE\Desktop\Advance Java>javac Cal.java
C:\Users\NICE\Desktop\Advance Java>java Cal
4
0
1
2
3
4
5
6
7
8
9
+
-
/
*
=

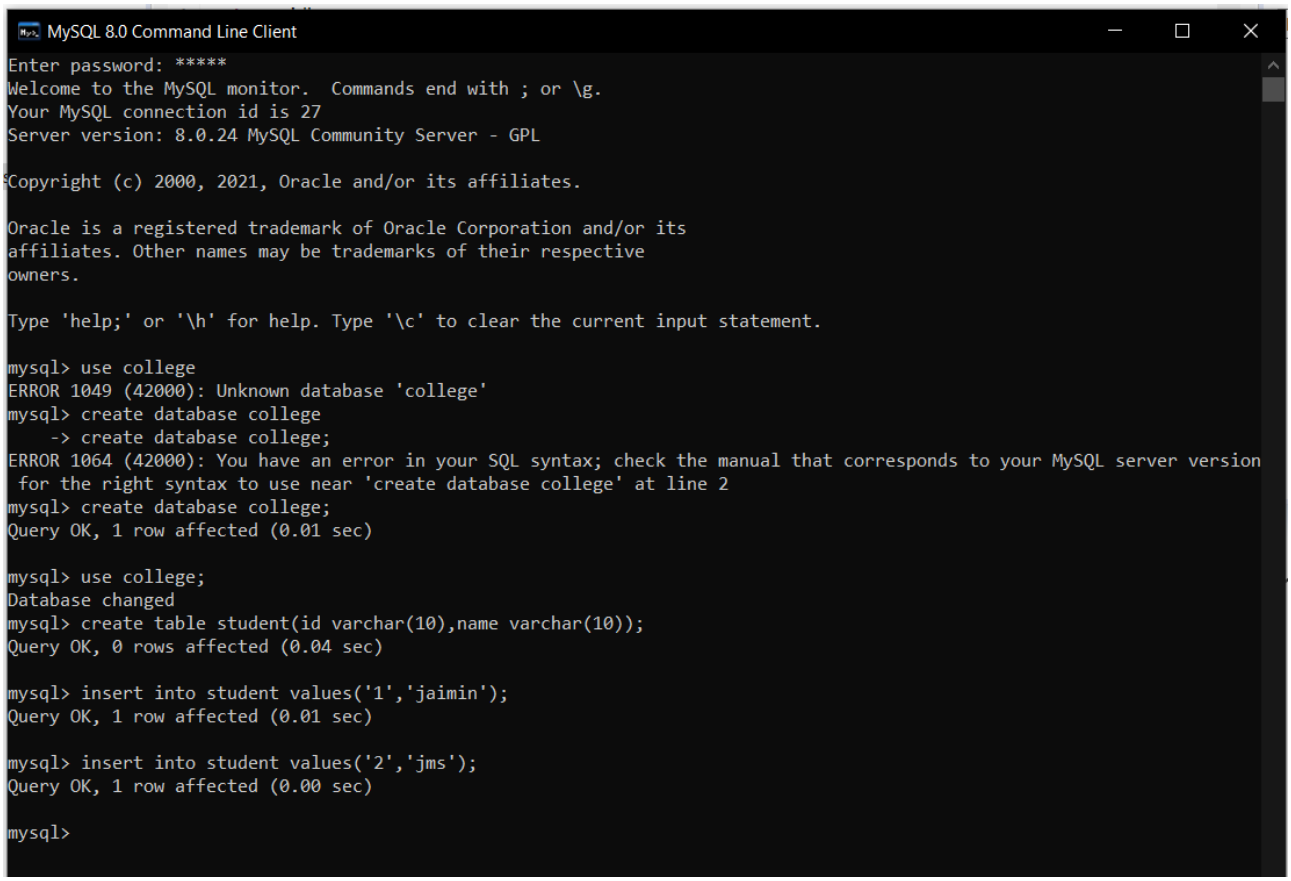
Practical 2

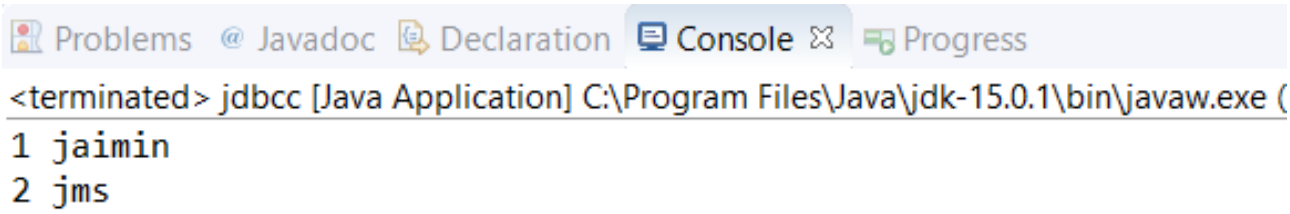
Aim: Create a database college in MySQL. Inside this database create a table student with id and name as fields. Write a JDBC program to get all this data from student table in college database.

Code:

```
package jdbc;
import java.sql.*;
public class jdbcc {
	public static void main(String[] args) throws Exception{
		Class.forName("com.mysql.cj.jdbc.Driver");
		Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/college","root","admin");
		Statement stmt=con.createStatement();
		String query="select * from student";
		ResultSet rs=stmt.executeQuery(query);
		while(rs.next())
		{
			System.out.println(rs.getString(1)+" "+rs.getString(2));
		}
		con.close();
	}

}
```

Output:


MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 27
Server version: 8.0.24 MySQL Community Server - GPL
Copyright (c) 2000, 2021, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> use college
ERROR 1049 (42000): Unknown database 'college'
mysql> create database college
-> create database college;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version
for the right syntax to use near 'create database college' at line 2
mysql> create database college;
Query OK, 1 row affected (0.01 sec)
mysql> use college;
Database changed
mysql> create table student(id varchar(10),name varchar(10));
Query OK, 0 rows affected (0.04 sec)
mysql> insert into student values('1','jaimin');
Query OK, 1 row affected (0.01 sec)
mysql> insert into student values('2','jms');
Query OK, 1 row affected (0.00 sec)
mysql>


Problems
Javadoc
Declaration
Console
Progress
<terminated> jdbcc [Java Application] C:\Program Files\Java\jdk-15.0.1\bin\javaw.exe (
1 jaimin
2 jms

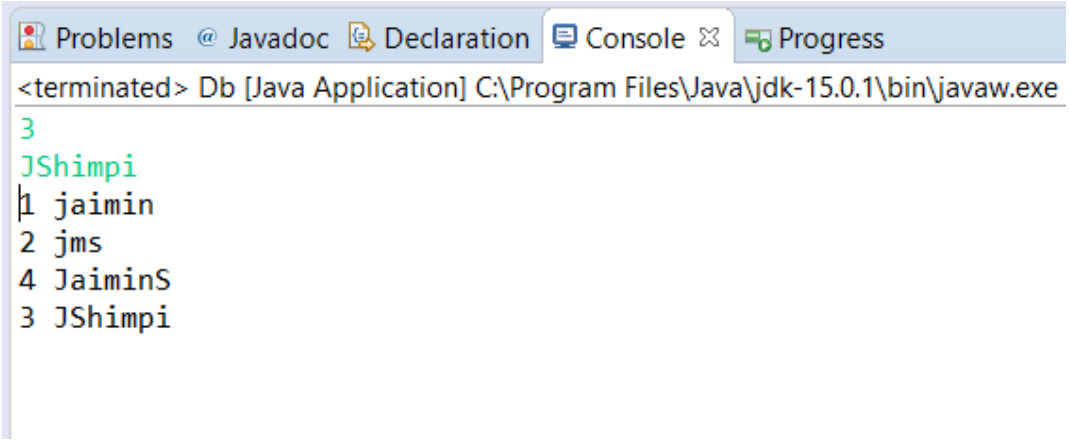
Practical 3

Aim: Do the above program for insert ID and NAME into student table in college database.

Code:

```
package jdbc;
import java.sql.*;
import java.io.*;

public class Db {
	public static void main(String[] args)throws Exception {
		Class.forName("com.mysql.cj.jdbc.Driver");
		Connection con = DriverManager.getConnection
		("jdbc:mysql://localhost:3306/college","root","admin");
		String query="insert into student values(?,?)";
		PreparedStatement stmt=con.prepareStatement(query);
		BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
		String id=br.readLine();
		String name=br.readLine();
		stmt.setString(1,id);
		stmt.setString(2,name);
		int i=stmt.executeUpdate();
		Statement st=con.createStatement();
		query="select * from student";
		ResultSet rs=st.executeQuery(query);
		while(rs.next())
		{
			System.out.println(rs.getString(1)+" "+rs.getString(2));
		}
		con.close();
	}
}
```

Output:


Problems
Javadoc
Declaration
Console
Progress
<terminated> Db [Java Application] C:\Program Files\Java\jdk-15.0.1\bin\javaw.exe
3
JShimpi
1 jaimin
2 jms
4 JaiminS
3 JShimpi

Practical 4

Aim: Write down java networking demo programs for two way communication (Simple chat).

Server.java

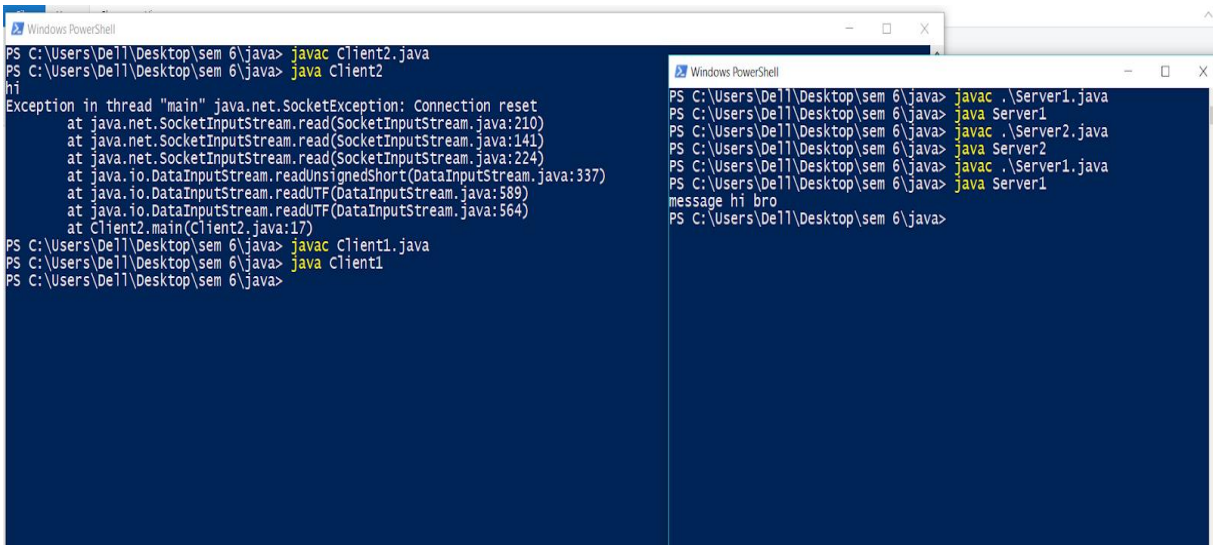
```
import java.net.*;
import java.util.*;
import java.io.*;
public class Server
{
     public static void main(String args[]) throws Exception
    {
         ServerSocket ss=new ServerSocket(1223);
         Socket s=ss.accept();
         BufferedReader in= BufferedReader(new InputStreamReader(s.getInputStream()));
         PrintStream out=new PrintStream(s.getOutputStream());
         BufferedReader kb= new BufferedReader (new InputStreamReader (System.in));
         String str="",str1="";
         while(!str.equals("stop"))
         {
                  str=in.readLine();
                  System.out.println("client says "+str);
                  Str1=kb.readLine();
                  out.println(str1);
         }
         out.close();
         in.close();
         kb.close();
         s.close();
         ss.close();
     }
}
```

Client.java

```
import java.net.*;
import java.io.*;
import java.util.*;
public class Client
{
    public static void main(String args[]) throws Exception
```

```
    {
        Socket s=new Socket("localhost",1223);
        BufferedReader din=new
BufferedReader(new              InputStreamReader(s.getInputStream()));
        DataOutputStream dout=new DataOutputStream(s.getOutputStream());
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        String str="", str2="";
        while(!str.equals("stop"))
        {
                str=br.readLine();
                dout.writeBytes(str);
                str2=din.readLine();
                System.out.println("server says "+str2);
        }
        dout.close();
        s.close();
    }
}
```

Output:



Practical 5

Aim: Write down simple program in servlet and show the web.xml configuration for the same.

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.Servlet;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.Servlet;
@WebServlet("/Demo")
public class Demo extends HttpServlet implements Servlet {
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter pw=response.getWriter();
        pw.println("welcome");
        pw.close();
    }
}
```

Index.html

```
 <html>
<body>
 <form action="testing" method="GET">
 <input type="submit" value="My name is khan">
 </body>
 </html>
```

web.xml

```
 <?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd" id="WebApp_ID" version="3.0">
        <display-name>Ser1</display-name>
    <servlet>
        <servlet-name>Demo</servlet-name>
```

```
        <servlet-class>Demo</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>Demo</servlet-name>
        <url-pattern>/welcome</url-pattern>
     </servlet-mapping>
  <welcome-file-list>
        <welcome-file>index.html</welcome-file>
        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.jsp</welcome-file>
        <welcome-file>default.html</welcome-file>
        <welcome-file>default.htm</welcome-file>
         <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
</web-app>
```

Output:

Practical 6

Aim: Create simple html form which contain username and password and submit to servlet page and display the same.

Ser6.java

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

@WebServlet(name = "Ser6")
public class Ser6 extends HttpServlet {
        protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
                response.setContentType("text/html");
                PrintWriter out=response.getWriter();
                String name=request.getParameter("user");
                out.print("welocme "+name);
   }
}
```

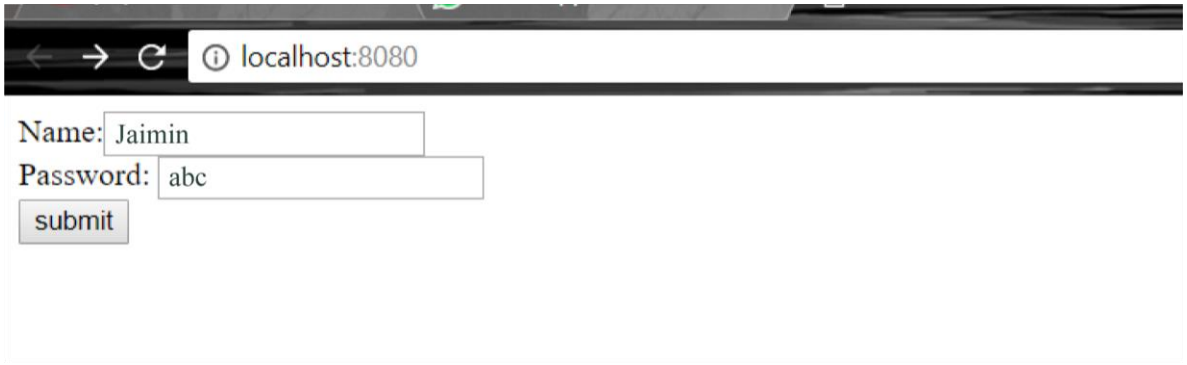
index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
</head>
<body>
<form action="welcome">
   Name:<input type="text" name="user"/><br>
   Password: <input type="test" name="pass"/><br>
   <input type="submit" value="submit"/>
</form>
</body>
</html>
```

web.xml

```
<web-app>
        <servlet>
        <servlet-name>Ser6</servlet-name>
        <servlet-class>Ser6</servlet-class>
        </servlet>

        <servlet-mapping>
        <servlet-name>Ser6</servlet-name>
        <url-pattern>/welcome </url-pattern>
        </servlet-mapping>

        <welcome-file-list>
        <welcome-file>index.html</welcome-file>
        </welcome-file-list>

</web-app>
```

Output:



Practical 7

Aim: Create html form which contains the following field.

- **Name (Text field)**
- **Surname (Text Field)**
- **Gender (Radio Button)**
- **Sports (Check Box...like cricket, tennis, soccer…)**
- **Feedback (Text area)**
- **And submit this page to servlet page which display all of above.**

Ser7.java

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


public class Ser7 extends HttpServlet {
        protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

          response.setContentType("text/html");
          PrintWriter out=response.getWriter();
          out.print("welocme
"+(String)request.getParameter("fname")+"
                  "+(String)request.getParameter("lname"));
          out.print("<br> Gender: "+request.getParameter("gender"));
          out.print("<br>");
          String a=request.getParameter("football");
          String b=request.getParameter("hockey");
          out.print("Hobbies: ");
          if(a=="null"){ }
          else
            out.print("Football ");
          if(b=="null"){ }
          else
          out.print("hockey");
```

```
            out.print("<br> Feedback: "+request.getParameter("fdbk"));
    }
}
```

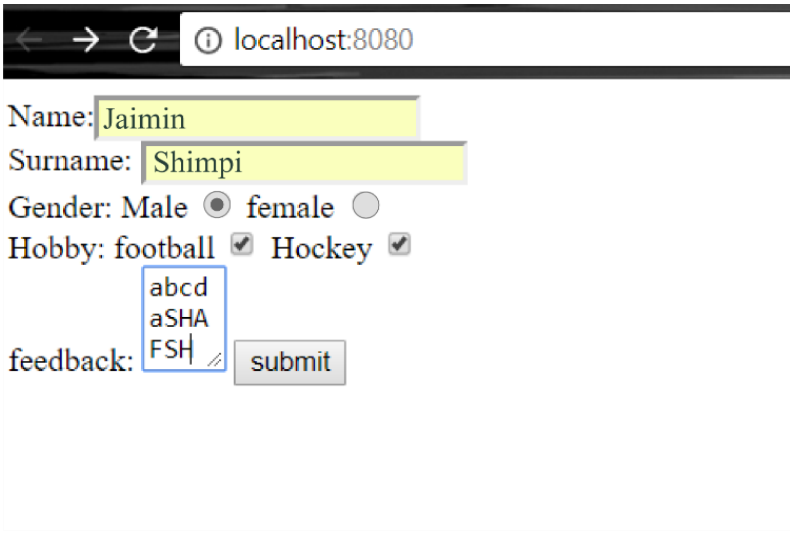
index.html

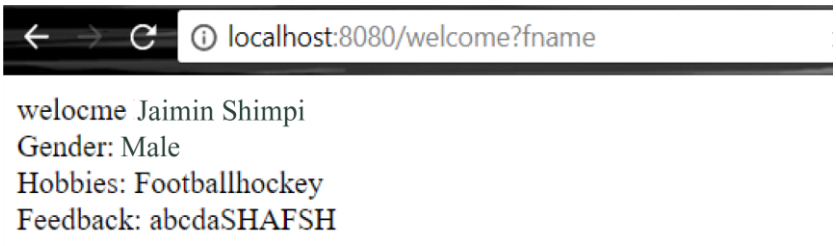
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
</head>
<body>
    <form action="welcome" method=""post">
            Name:<input type="text" name="fname"/><br>
            Surname: <input type="text" name="lname"/><br>
            Gender: Male  <input type="radio" name="gender" value="male"/>
            female
             <input type="radio" name="gender" value="female"/> <br>
            Hobby: football
            <input type="checkbox" name="football"/>
             Hockey <input type="checkbox" name="hockey"/> <br>
            feedback: <textarea name="fdbk" cols="3" rows="3"></textarea>
            <input type="submit" value="submit"/>
</form>
</body>
</html>
```

web.xml

```
<web-app>
                <servlet>
                <servlet-name>Ser7</servlet-name>
                <servlet-class>Ser7</servlet-class>
                </servlet>
                <servlet-mapping>
                <servlet-name>Ser7</servlet-name>
                <url-pattern>/welcome</url-pattern>
                </servlet-mapping>
                <welcome-file-list>
                <welcome-file>index.html</welcome-file>
                </welcome-file-list>
</web-app>
```

Output:


localhost:8080
Name: Jaimin
Surname: Shimpi
Gender: Male female
Hobby: football Hockey
abcd
aSHA
FSH
feedback:
submit


localhost:8080/welcome?fname
welocme Jaimin Shimpi
Gender: Male
Hobbies: Footballhockey
Feedback: abcdaSHAFSH

Practical 8

Aim: Write the servlet program which can get the username and password from html and if username is ADMIN and password is BVM then send Redirect to valid user page, else show not valid user.

web.xml

```
<web-app
        <servlet>
                <servlet-name>Ser8</servlet-name>
                <servlet-class>Ser8</servlet-class>
        </servlet>
        <servlet-mapping>
                <servlet-name>Ser8</servlet-name>
                <url-pattern>/login</url-pattern>
        </servlet-mapping>
</web-app>
```

Ser8.java

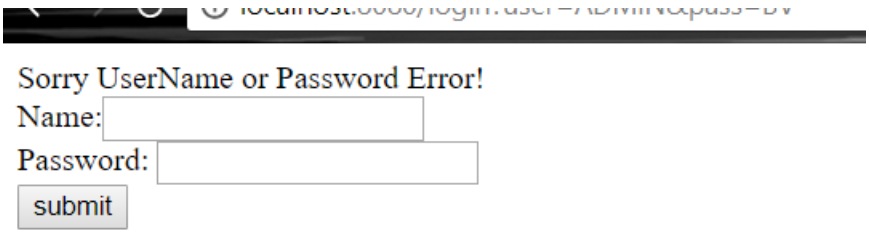
```
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.IOException;
import java.io.PrintWriter;
public class Ser8 extends HttpServlet {
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws       ServletException, IOException {

   response.setContentType("text/html");
   PrintWriter out=response.getWriter();
   String name = (String)request.getParameter("user");
   String pass = (String)request.getParameter("pass");
   if(pass.equals("BVM") && name.equals("ADMIN"))
   {
      response.sendRedirect("/welcome.html");
   }
   else
   {
      out.print("Sorry UserName or Password Error!");
      RequestDispatcher rd=request.getRequestDispatcher("/index.html");
      rd.include(request, response);
```

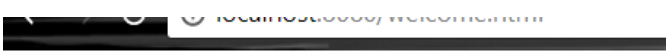
```
    }
  }
}
```

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
</head>
<body>
<form action="login">
    Name:<input type="text" name="user"/><br>
    Password: <input type="text" name="pass"/><br>
    <input type="submit" value="submit"/>
</form>
</body>
</html>
```

Output:





Practical 9

Aim: Write the simple session program which display new or old session with the Session ID, CreationTime, LastAccessTime and session Timeout .

Ser91.java

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

@WebServlet(name = "Ser91")
public class Ser91 extends HttpServlet {
   protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

   response.setContentType("text/html");
   PrintWriter out=response.getWriter();
   HttpSession session = request.getSession(false);
      String name=(String)session.getAttribute("uname");

   out.print("Hello "+ name);
   out.print("<br> Session id:  "+session.getId());
   out.print("<br> creation time" + session.getCreationTime());
   out.print("<br> Last modified time" + session.getLastAccessedTime());
   out.print("<br> Max inactive time" + session.getMaxInactiveInterval());
   out.close();
   }
}
```

Ser9.java

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

```
import java.io.PrintWriter;

@WebServlet(name = "Ser9")
public class Ser9 extends HttpServlet {


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

   response.setContentType("text/html");
   PrintWriter out=response.getWriter();
   String name=(String)request.getParameter("user");
   out.print("welcome "+name);
      HttpSession session = request.getSession();
      session.setAttribute("uname",name);
   out.print("<a href='login1'> Next</a>");
   out.close();
   }
}
```

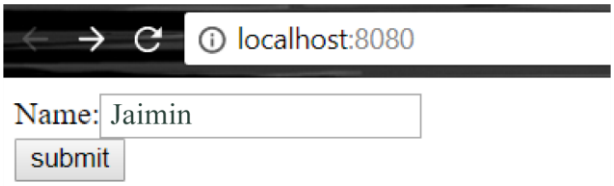
web.xml

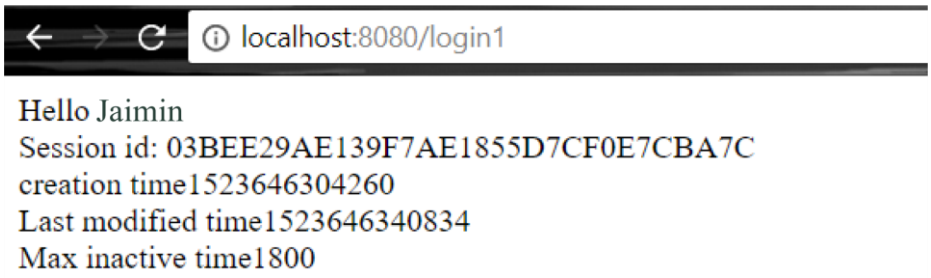
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
        version="4.0">

   <servlet>
      <servlet-name>Ser9</servlet-name>
      <servlet-class>Ser9</servlet-class>
   </servlet>
   <servlet-mapping>
      <servlet-name>Ser9</servlet-name>
      <url-pattern>/login</url-pattern>
   </servlet-mapping>
   <servlet>
      <servlet-name>Ser91</servlet-name>
      <servlet-class>Ser91</servlet-class>
   </servlet>
   <servlet-mapping>
      <servlet-name>Ser91</servlet-name>
      <url-pattern>/login1</url-pattern>
   </servlet-mapping>
</web-app>
```

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
</head>
<body>
<form action="login">
   Name:<input type="text" name="user"/><br>
   <input type="submit" value="submit"/>
</form>

</body>
</html>
```

Output:





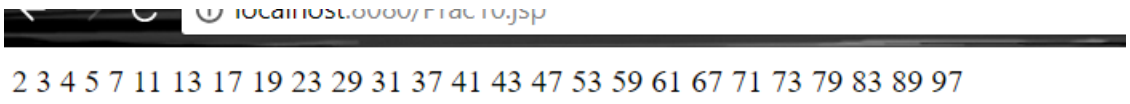
Practical 10

Aim: Write down the simple JSP program which displays prime numbers between 1 to 100.

Prac10.jsp

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
   <title>Practical 10</title>
</head>
<body>
<%
   int i,j,flag=0;
   out.print("2");
   for(i=3;i<=100;i++)
   {
      flag=0;
      for(j=2;j<(i/2);j++)
      {
        if((i%j)==0)
           flag=1;
      }
      if(flag==0)
         out.print(" "+i);
   }
%>
</body>
</html>
```

Output:



Practical 11

Aim: Declare addition and multiplication methods in declarative tag. Call addition method using scriptlet and multiplication method through expression.

Prac11.jsp

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
   <title>Practical 10</title>
</head>
<body>
<%!
   int sum(int a , int b)
   {
      int c=a+b;
      return c;
   }
   int mul(int a, int b)
   {
      int c=a*b;
      return c;
   }
%>
<%
   out.print("sum "+sum(4,5));
%>
<br>
Product is : <%= mul(4,5) %>
</body>
</html>
```

Output: -

Practical 12

Aim: Create html page which can get the multiple sports selection from the user and call jsp page to display the same.

Prac12.jsp

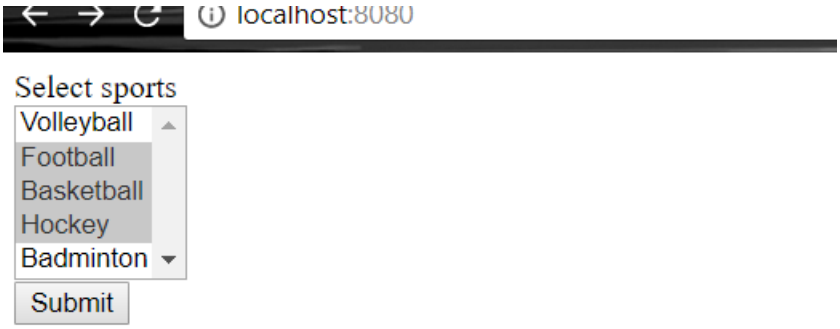
```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
   <title>Practical 12</title>
</head>
<body>
        You selected:
        <br>
        <% String[] items = request.getParameterValues("sports");
                for(int i= 0; i < items.length; i++){
                        out.println(items[i] + "<br>");
                }
        %>

</body>
</html>
```

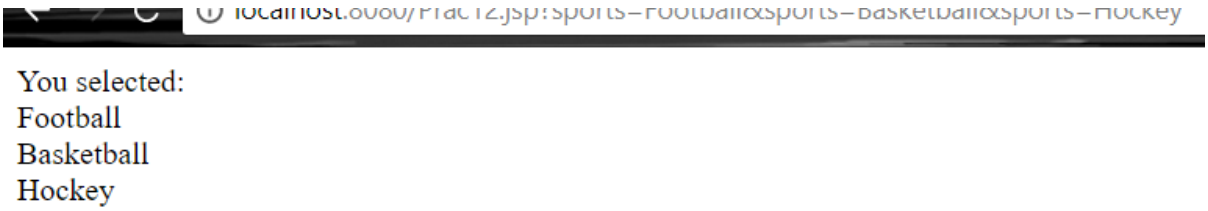
index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
</head>
<body>
<form action="Prac12.jsp" >
   Select sports <br>
   <select name="sports" size="5" multiple>
      <option>Volleyball</option>
      <option>Football</option>
      <option>Basketball</option>
      <option>Hockey</option>
      <option>Badminton</option>
   </select>
   <br>
   <input type="submit" value="Submit">
</form>
```

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

Output: -





Practical 13

Aim: Do the above program by using JSTL core library functions.

Practical13.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <title>Practical 13</title>
</head>
<body>
   <form action="Practical13.jsp" method="get">
      <h3>Sports</h3><br>
      <input type="checkbox" name="sports" value="Cricket">Cricket<br>
      <input type="checkbox" name="sports" value="Tennis">Tennis<br>
      <input type="checkbox" name="sports" value="Football">Football<br><br>
      <input type="submit" name="submit">
   </form>
</body>
</html>
```

Practical13.jsp

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
   <title>Practical13</title>
</head>
<body>
   <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
   <%
      String s[]=request.getParameterValues("sports");
      request.setAttribute("s",s);
   %>
   <c:forEach var="str" items="${s}">
      <c:out value="${str}"/>
   </c:forEach>
</body>
</html>
```

Output:


Sports
Cricket
Tennis
Football
Submit
localhost:8080/Practical13.jsp?sports=Cricket&sports=Tennis&submit=Sul
Cricket Tennis

Practical 14

Aim: Write down a simple Program in JSF by using Managed Bean.

index.xhtml

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:h="http://xmlns.jcp.org/jsf/html">
        <h:head>
                <title>Facelet Title</title>
         </h:head>

        <h:body>
                        #{second.world}
        </h:body>
</html>
```

second.java:

```
import javax.faces.bean.ManagedBean;
import javax.faces.bean.RequestScoped;
@ManagedBean
@RequestScoped
public class second
{
                String world ="Hello World---";
                 public String getWorld()
        {
                 return world;
         }

}
```

Practical 15

Aim: Write down a JSF Page Which Consist of Basic HTML Tags.

Code:

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

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
       "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
      xmlns:f="http://xmlns.jcp.org/jsf/core">
<f:view>
   <h:form>
       <h:panelGrid columns="2">
          <h:outputLabel value="Celsius"/>
          <h:inputText  value="#{TemperatureConvertor.celsius}"/>
       </h:panelGrid>
       <h:commandButton action="#{TemperatureConvertor.celsiusToFahrenheit}"
value="Calculate"/>
       <h:commandButton action="#{TemperatureConvertor.reset}" value="Reset"/>
       <h:messages layout="table"/>
   </h:form>
   <h:panelGroup rendered="#{TemperatureConvertor.initial!=true}">
       <h3> Result </h3>
       <h:outputLabel value="Fahrenheit "/>
       <h:outputLabel value="#{TemperatureConvertor.fahrenheit}"/>
   </h:panelGroup>
</f:view>
</html>
```

Practical 16

Aim: Write down a simple program in spring.

HelloWorld.java

```
public class HelloWorld {
   private String message;

   public void setMessage(String message){
      this.message  = message;
   }
   public void getMessage(){
      System.out.println("Your Message : " + message);
   }
}
```

MainApp.java

```
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class MainApp {
   public static void main(String[] args) {
      ApplicationContext context = new ClassPathXmlApplicationContext("Beans.xml");
      HelloWorld obj = (HelloWorld) context.getBean("helloWorld");
      obj.getMessage();
   }
}
```

Beans.xml

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

<beans xmlns = "http://www.springframework.org/schema/beans"
   xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation = "http://www.springframework.org/schema/beans
   http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">

<bean id = "helloWorld" class = "HelloWorld">
      <property name = "message" value = "Hello World!"/>
   </bean>
</beans>
```

Practical 17

Aim: Create application in hibernate which will create a object of contact class and set the id, first name, last name and email fields. Finally save the whole object in My SQL Database. (Four types of files are required. One for core java class with setter and getter methods and one for main application. Remaining two are the configuration files.

Contact.java:

```
public class Contact {
   String id,firstname,lastname,email;

   public String getId() {
      return id;
   }
   public void setId(String id) {
      this.id = id;
   }
   public String getFirstname() {
      return firstname;
   }
   public void setFirstname(String firstname) {
      this.firstname = firstname;
   }
   public String getLastname() {
      return lastname;
   }
   public void setLastname(String lastname) {
      this.lastname = lastname;
   }
   public String getEmail() {
      return email;
   }
   public void setEmail(String email) {
      this.email = email;
   }
}
```

Hibernate.hbm.xml:

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE hibernate-mapping PUBLIC
 "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
 "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
 <hibernate-mapping>
  <class name="Contact" table="contact">
    <id name="id">
     <generator class="assigned"></generator>
    </id>
    <property name="firstname"></property>
    <property name="lastname"></property>
   <property name="email"></property
  </class>
 </hibernate-mapping>
```

Hibernate.cfg.xml: -

```
<?xml version='1.0' encoding='utf-8'?>

<!DOCTYPE hibernate-configuration PUBLIC
        "-//Hibernate/Hibernate Configuration DTD//EN"
        "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
    <session-factory>

        <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
        <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/contact</property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password"></property>
        <property name="hibernate.connection.pool_size">1</property>
        <property name="hibernate.current_session_context_class">thread</property>
        <property name="hibernate.show_sql">true</property>
        <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

        <mapping resource="hibernate.hbm.xml"/>

    </session-factory>
</hibernate-configuration>
```

Main.java:

```
import org.hibernate.*;
import org.hibernate.cfg.Configuration;

public class Main {

   public static void main(final String[] args) throws Exception {
      Configuration cfg= new Configuration();
      cfg.configure("hibernate.cfg.xml");
      SessionFactory sessionFactory = cfg.buildSessionFactory();
      Session session = sessionFactory.openSession();
      Transaction t = session.beginTransaction();

      Contact c1 = new Contact();
      c1.setId("1234");
      c1.setFirstname("abc");
      c1.setLastname("xyz");
      c1.setEmail("abc@xyz.com");

      session.persist(c1);
      t.commit();
      session.close();
      System.out.println("Successfully saved");
   }
}
```