Programme Name: B.Sc (Hons) Computer Science
Semester: V
Paper Title: Internet Technologies Practical

Submitted by:
DIYA GARG
Examination Roll No: 18 044 570 007
College Roll No: CSC/18/15

College Name:
Mata Sundri College for Women, University of Delhi
Address: Mata Sundari Lane, New Delhi- 110002

JAVA

1. Implement a BankAccount class and a class Bank.

//BA2.java

```
import java.util.*;

class BankAccount
{
int accnum;
double bal;
int taxrate=5;

public BankAccount(int x, double y)
{
        accnum= x;
        bal= y;
}

BankAccount(double z)
{
        accnum= accnum+1;
        bal= z;
}

public void deposit(double x)
{
        double newbal= bal+x;
        bal= newbal;
}

public void withdraw(double x)
{
```

```
        double newbal= bal-x;
        bal= newbal;
}

public int getAccNo()
{
        return accnum;
}

public double getBalance()
{
        return bal;
}

public void taxdeduction()
{
        double tax= (bal*taxrate)/100.0;
        withdraw(tax);
}
}
```

class Bank

```
{
ArrayList<BankAccount> accounts;

public Bank()
{
        accounts= new ArrayList<BankAccount>();
}

public void addAcc(BankAccount a)
{
        accounts.add(a);
}

public double total()
{
```

```
        double sum=0;
        for(BankAccount b: accounts)
                sum+=b.getBalance();
        return sum;
}

public int max()
{
        if(accounts.size()==0)
        {
                System.out.println("The bank has no accounts.");
                return -1;
        }
        else
        {
                BankAccount m= accounts.get(0);
                for(BankAccount b: accounts)
                        if(b.getBalance()>m.getBalance())
                                m=b;
                return m.getAccNo();
        }
}

public int min()
{
        if(accounts.size()==0)
        {
                System.out.println("The bank has no accounts.");
                return -1;
        }
        else
        {
                BankAccount m= accounts.get(0);
                for(BankAccount b: accounts)
                        if(b.getBalance()<m.getBalance())
                                m=b;
                return m.getAccNo();
```

```
        }
}

public BankAccount findAcc(int accno)
{
        for(BankAccount b: accounts)
                if(b.getAccNo() == accno)
                        return b;
        return null;
}

public int countAcc(double specBal)
{
        int c=0;
        for(BankAccount b: accounts)
                if(b.getBalance() >= specBal)
                        c++;
        return c;
}
}

public class BA2
{
public static void main(String args[])
{
        Bank b=new Bank();

        Scanner sc= new Scanner(System.in);
        System.out.println("");
        System.out.println("BANK ACCOUNT OPERATIONS");
        System.out.println("1. Add account with account number");
        System.out.println("2. Add account without account number");
        System.out.println("3. Get total balance in the bank");
        System.out.println("4. Account number with maximum balance");
        System.out.println("5. Account number with minimum balance");
        System.out.println("6. Find an account given a bank account number");
```

```
            System.out.println("7. Count no. of accounts having at least specific
balance");
            System.out.println("");

            char ch= 'n';

            do{

            System.out.print("Enter your choice: ");
            String choice=sc.next();
            switch(choice)
            {
                  case "1":
                        System.out.print("Enter the account number: ");
                        int a= sc.nextInt();
                        System.out.print("Enter the starting balance: ");
                        double d=sc.nextDouble();
                        b.addAcc(new BankAccount(a,d));
                        System.out.println("Account succesfully created.");
                        break;

                  case "2":
                        System.out.print("Enter the starting balance: ");
                        double d1=sc.nextDouble();
                        BankAccount myac=new BankAccount(d1);
                        int num=myac.getAccNo();
                        System.out.println("Account succesfully created with
account number- "+ num);
                        break;

                  case "3":
                        System.out.println("Total Balance in the bank= "+
b.total());
                        break;
```

```
                        case "4":
                                System.out.println("AccountNo having maximum Balance
in the bank= "+ b.max());
                                break;

                        case "5":
                                System.out.println("AccountNo having minimum Balance
in the bank= "+ b.min());
                                break;

                        case "6":
                                System.out.print("Enter the account number you want to
search: ");
                                int ac= sc.nextInt();
                                BankAccount aa= b.findAcc(ac);
                                if(aa==null)
                                        System.out.println("No such account exists.");
                                else
                                        System.out.println("Account exists with balance: " +
aa.getBalance());
                                break;

                        case "7":
                                System.out.print("Enter the amount: ");
                                double dd= sc.nextDouble();
                                System.out.println("No. of accounts having at least Rs.
500 as balance in the bank= "+ b.countAcc(dd));
                                break;
                        default:
                                System.out.println("Enter a valid choice.");
                }
                System.out.print("Do you wish to continue: ");
                ch= sc.next().charAt(0);
                }while(ch=='y' || ch== 'Y');
        }
}
```

//OUTPUT

```
C:\Users\aaa\Desktop\Semester 5\IT_Practicals>javac BA2.java

C:\Users\aaa\Desktop\Semester 5\IT_Practicals>java BA2

BANK ACCOUNT OPERATIONS
1. Add account with account number
2. Add account without account number
3. Get total balance in the bank
4. Account number with maximum balance
5. Account number with minimum balance
6. Find an account given a bank account number
7. Count no. of accounts having atleast specific balance

Enter your choice: 1
Enter the account number: 001
Enter the starting balance: 1000
Account succesfully created.
Do you wish to continue: y
Enter your choice: 1
Enter the account number: 002
Enter the starting balance: 2000
Account succesfully created.
Do you wish to continue: y
Enter your choice: 3
Total Balance in the bank= 3000.0
Do you wish to continue: y
Enter your choice: 4
AccountNo having maximum Balance in the bank= 2
Do you wish to continue: y
Enter your choice: 5
AccountNo having minimum Balance in the bank= 1
Do you wish to continue: y
Enter your choice: 6
Enter the account number you want to search: 001
Account exists with balance: 1000.0
Do you wish to continue: y
Enter your choice: 6
Enter the account number you want to search: 3
No such account exists.
Do you wish to continue: y
Enter your choice: 7
Enter the amount: 500
No. of accounts having atleast Rs. 500 as balance in the bank= 2
Do you wish to continue: y
Enter your choice: 7
Enter the amount: 6000
No. of accounts having atleast Rs. 500 as balance in the bank= 0
Do you wish to continue: n
```

2. Implement an Abstract class Stack and its two Subclasses - StaticStack and DynamicStack.

//stk.java

```
import java.util.*;
import java.util.Scanner;

abstract class stackk
{
	public int top;

	public stackk()
	{
		top= -1;
	}

	public abstract void push(int el);
	public abstract int pop();
	public abstract void display();
}

class s_stack extends stackk
{
	int arr[];

	public s_stack(int size)
	{
		super();
		arr= new int[size];
	}

	public void push(int el)
	{
		if(super.top==arr.length)
```

```
            {
                System.out.println("Stack Overflow.");
            }
            else
            {
                super.top++;
                arr[super.top]= el;
            }
        }

        public int pop()
        {
            if(super.top==-1)
            {
                System.out.println("Stack Underflow.");
                return -1;
            }
            else
            {
                return arr[super.top--];
            }
        }

        public void display()
        {
            System.out.print("[");
            for(int i=0; i<=super.top; i++)
            {
                System.out.print(arr[i]+" ");
            }
            System.out.println("]");

        }
}
```

```
class d_stack extends stackk
{
	ArrayList<Integer> al;

	public d_stack()
	{
		super();
		al= new ArrayList<Integer>();
	}

	public void push(int el)
	{
		super.top++;
		al.add(super.top, el);
	}

	public int pop()
	{
		if(super.top== -1)
		{
			System.out.println("Stack Underflow");
			return -1;
		}
		else
		{
			int el= al.get(super.top);
			al.remove(super.top);
			super.top--;
			return el;
		}
	}

	public void display()
	{
		System.out.print("[");
```

```
                for(int i=0; i<=super.top; i++)
                {
                        System.out.print(al.get(i)+" ");
                }
                System.out.println("]");
        }
}

public class stk
{
        public static void main(String args[])
        {
                d_stack ds= new d_stack();
                s_stack s= new s_stack(10);

                Scanner sc= new Scanner(System.in);

                System.out.println("Stack Menu");
                System.out.println("1. Push into static stack");
                System.out.println("2. Pop from static stack");
                System.out.println("3. Push into dynamic stack");
                System.out.println("4. Pop from dynamic stack");

                char ch='n';

                do{

                System.out.print("Enter your choice: ");
                String choice= sc.next();
                switch(choice)
                {
                        case "1":
                                System.out.print("Enter the number of elements to
be pushed: ");
                                int a= sc.nextInt();
                                for(int j=0; j<a; j++)
                                {
```

```
                                        System.out.print("Enter the element to be
pushed: ");
                                        int b=sc.nextInt();
                                        s.push(b);
                                        s.display();
                                }
                                break;

                        case "2":
                                s.pop();
                                s.display();
                                break;

                        case "3":
                                System.out.print("Enter the number of elements to
be pushed: ");
                                int c= sc.nextInt();
                                for(int j=0; j<c; j++)
                                {
                                        System.out.print("Enter the element to be
pushed: ");
                                        int d= sc.nextInt();
                                        ds.push(d);
                                        ds.display();
                                }
                                break;

                        case "4":
                                ds.pop();
                                ds.display();
                                break;
                        default:
                                System.out.println("Enter a valid choice.");
                }
                System.out.print("Do you wish to continue: ");
                ch= sc.next().charAt(0);
                }while(ch=='y' || ch== 'Y');
```

```
    }
}
```

//OUTPUT

```
C:\Users\aaa\Desktop\Semester 5\IT_Practicals>javac stk.java

C:\Users\aaa\Desktop\Semester 5\IT_Practicals>java stk
Stack Menu
1. Push into static stack
2. Pop from static stack
3. Push into dynamic stack
4. Pop from dynamic stack
Enter your choice: 1
Enter the number of elements to be pushed: 4
Enter the element to be pushed: 4
[4 ]
Enter the element to be pushed: 6
[4 6 ]
Enter the element to be pushed: 1
[4 6 1 ]
Enter the element to be pushed: 8
[4 6 1 8 ]
Do you wish to continue: y
Enter your choice: 2
[4 6 1 ]
Do you wish to continue: y
Enter your choice: 3
Enter the number of elements to be pushed: 3
Enter the element to be pushed: 9
[9 ]
Enter the element to be pushed: 1
[9 1 ]
Enter the element to be pushed: 7
[9 1 7 ]
Do you wish to continue: y
Enter your choice: 4
[9 1 ]
Do you wish to continue: y
Enter your choice: 4
[9 ]
Do you wish to continue: y
Enter your choice: 4
[]
Do you wish to continue: y
Enter your choice: 4
Stack Underflow
[]
Do you wish to continue: n
```

JAVABEANS

1. Implement Employee JavaBean using Serializability interface.

//EmployeeBean.java

```
package Employees;

import java.io.Serializable;

public class EmployeeBean implements Serializable
{
   private String name;
   private String deptt;
   private String empID;
   private String address;
   private String gender;
   private String birthdate;
   private String email;
   private float salary;

   public EmployeeBean()
   {
      name= "";
      deptt = "";
      empID = "";
      address = "";
     gender = "";
      birthdate = "";
      email = "";
   }

   public String getName() {
      return name;
```

```
}

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

public String getDeptt() {
   return deptt;
}

public void setDeptt(String deptt) {
   this.deptt = deptt;
}

public String getEmpID() {
   return empID;
}

public void setEmpID(String empID) {
   this.empID = empID;
}

public String getAddress() {
   return address;
}

public void setAddress(String address) {
   this.address = address;
}

public String getGender() {
   return gender;
}

public void setGender(String gender) {
   this.gender = gender;
}

public String getBirthdate() {
```

```
        return birthdate;
    }

    public void setBirthdate(String birthdate) {
        this.birthdate = birthdate;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public float getSalary() {
        return salary;
    }

    public void setSalary(float salary) {
        this.salary = salary;
    }


}
```

//Details_EmpBean.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Employee Details</title>
<style>
input[type= "text"]
{
  width: 50%;
  padding: 10px 15px;
```

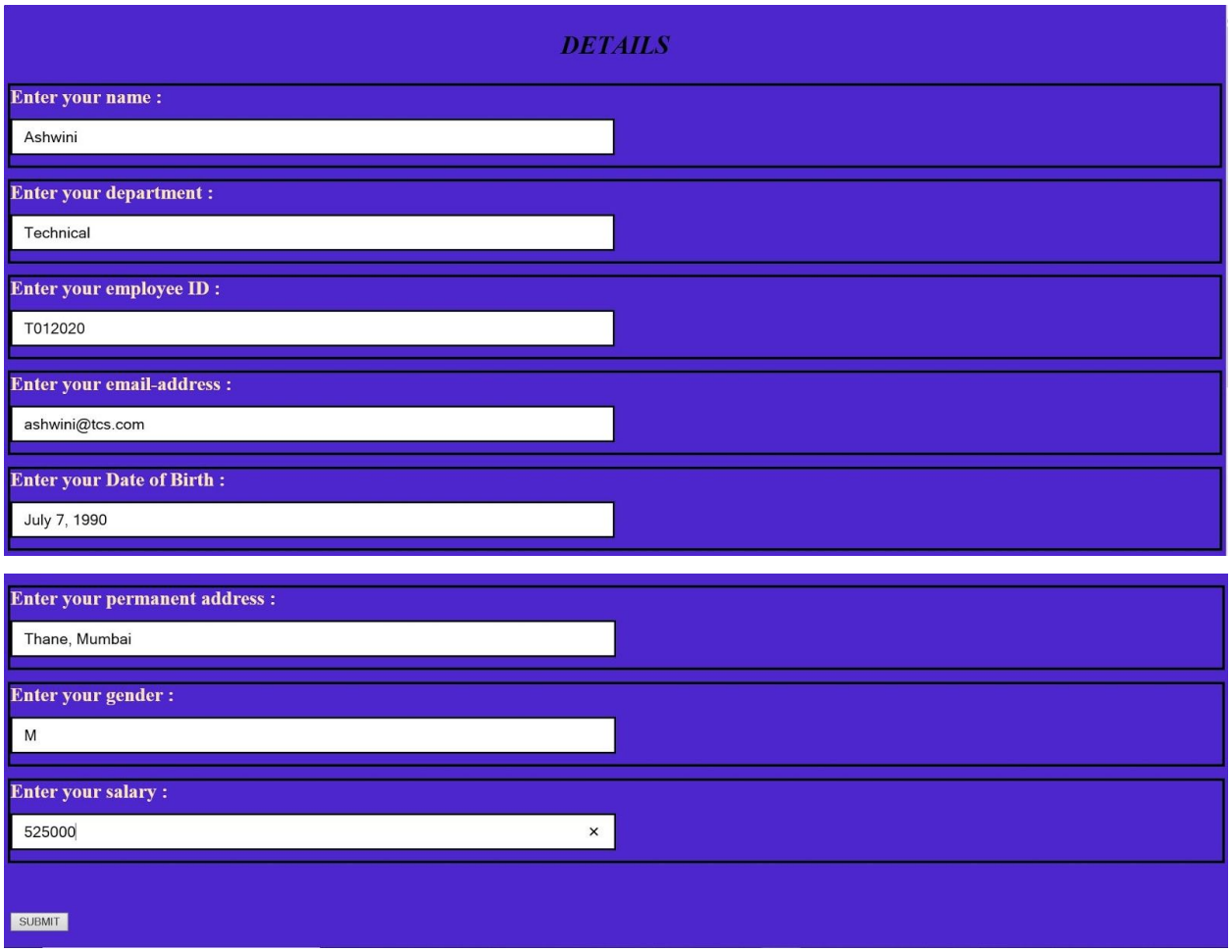
```
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}

body
{
  background: #f3f3f3;
}
</style>
</head>
<body style="background-color:#4d27cd;">
        <h1><i><p style= "color: #D6FFB6;"><center>DETAILS</center></p></i></h1>
                <form action="Print_EmpBean.jsp" method="get">
                        <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter your name :</label>
                        <br><input type="text" name="name" id="name"  value="" ></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your department :</label>
                        <br><input type="text" name="deptt" id="deptt"  value="" ></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your employee ID :</label>
                        <br><input type="text" name="empID" id="empIDl"  value="" ></h2>
                        <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter your email-address :</label>
                        <br><input type="text" name="email" id="email"  value="" ></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your Date of Birth :</label>
                        <br><input type="text" name="birthdate" id="birthdate"  value=""
></h2>
                                <h2 style= "outline-style: solid; outline-color: black;"><label
style= "color: #FFE8B6;" > Enter your permanent address :</label>
                        <br><input type="text" name="address" id="address"  value=""
></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your gender :</label>
                        <br><input type="text" name="gender" id="gender"  value="" ></h2>
```

```
                        <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter your salary :</label>
                        <br><input type="text" name="salary" id="salary"  value="" ></h2>


                <br><h2><input type="submit" value="SUBMIT"></h2>
                                </h1>
                </form>
        </body>
</html>
```

//Print_EmpBean.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>EmployeeBean</title>
  </head>
  <body>
    <jsp:useBean id="u1" scope="session"
class="Employees.EmployeeBean"></jsp:useBean>
    <jsp:setProperty name="u1" property="*" />
    You have entered the below details:<br></b></h2><h3>
    NAME: <jsp:getProperty name="u1" property= "name" /><br>
    DEPARTMENT NAME: <jsp:getProperty name="u1" property= "deptt" /><br>
    EMPLOYEE ID: <jsp:getProperty name="u1" property= "empID" /><br>
    BIRTH DATE: <jsp:getProperty name="u1" property= "birthdate" /><br>
    GENDER: <jsp:getProperty name="u1" property= "gender" /><br>
    EMAIL: <jsp:getProperty name="u1" property= "email" /><br>
    PERMANENT ADDRESS: <jsp:getProperty name="u1" property= "address" /><br>
    SALARY: <jsp:getProperty name="u1" property= "salary" /><br>
  </body>
</html>
```

//OUTPUT



You have entered the below details:

NAME: Ashwini
DEPARTMENT NAME: Technical
EMPLOYEE ID: T012020
BIRTH DATE: July 7, 1990
GENDER: M
EMAIL: ashwini@tcs.com
PERMANENT ADDRESS: Thane, Mumbai
SALARY: 52500.0

2. Implement Student JavaBean using Serializability interface.

//StudentBean.java

```
package Students;

import java.io.Serializable;

public class StudentBean implements Serializable
{
   private String name;
   private String rollno;
   private String course;
   private String address;
   private String gender;
   private String birthdate;
   private String email;
   private double marks[];

   public StudentBean()
   {
      name= "";
      rollno = "";
      course = "";
      address = "";
     gender = "";
      birthdate = "";
      email = "";
   }

   public String getName() {
      return name;
   }

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

```
public String getRollno() {
    return rollno;
}

public void setRollno(String rollno) {
    this.rollno = rollno;
}

public String getCourse() {
    return course;
}

public void setCourse(String course) {
    this.course = course;
}

public String getAddress() {
    return address;
}

public void setAddress(String address) {
    this.address = address;
}

public String getGender() {
    return gender;
}

public void setGender(String gender) {
    this.gender = gender;
}

public String getBirthdate() {
    return birthdate;
}

public void setBirthdate(String birthdate) {
    this.birthdate = birthdate;
```

```
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public double[] getMarks() {
        return marks;
    }

    public void setMarks(double[] marks) {
        this.marks = marks;
    }
}
```

//Details_StudBean.html

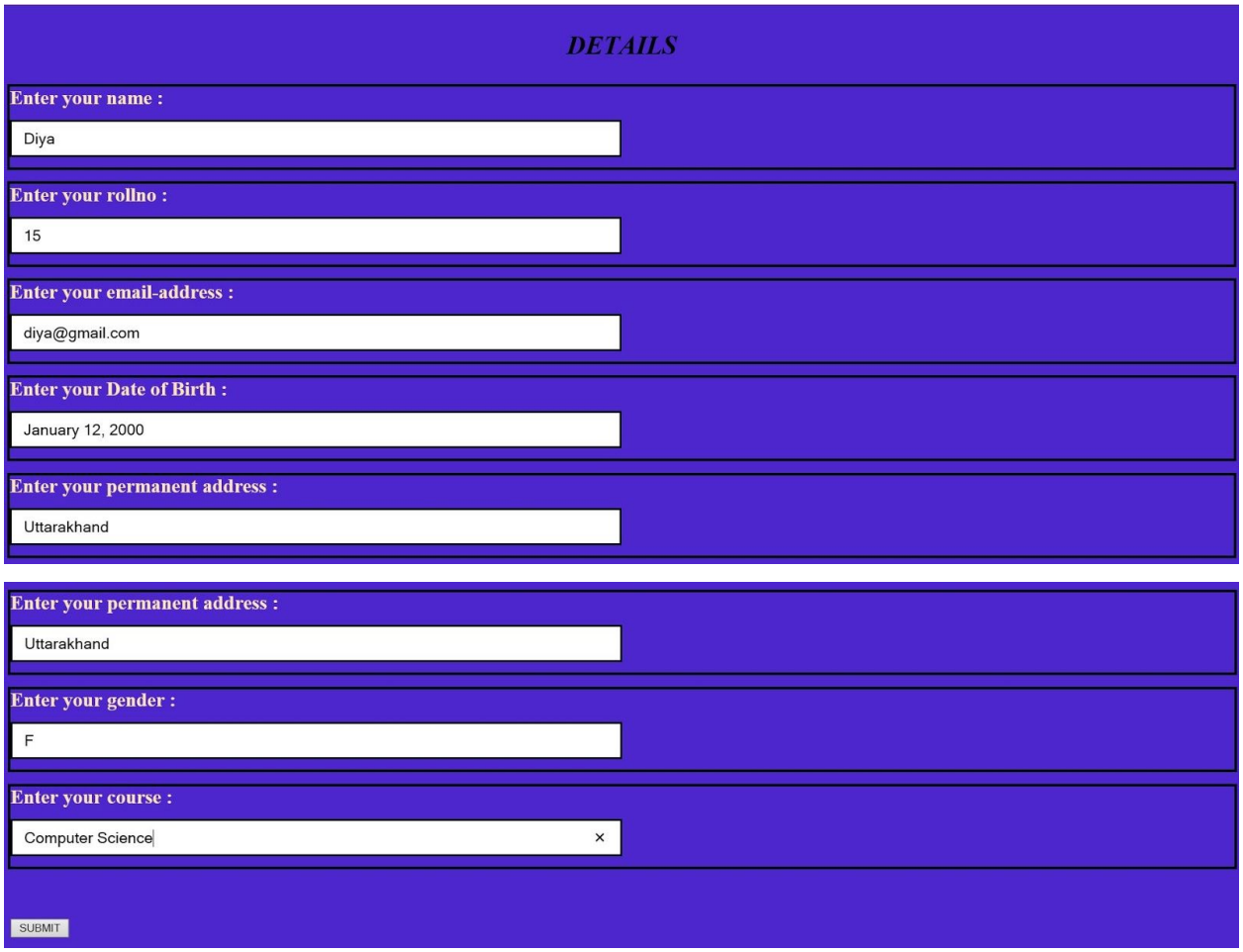
```
<!DOCTYPE html>
<html>
<head>
    <title>Student Details</title>
<style>
input[type= "text"]
{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}

body
{
  background: #f3f3f3;
```

```
}
</style>
</head>
<body style="background-color:#4d27cd;">
        <h1><i><p style= "color: #D6FFB6;"><center>DETAILS</center></p></i></h1>
                <form action="Print_StudBean.jsp" method="get">
                        <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter your name :</label>
                        <br><input type="text" name="name" id="name"  value="" ></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your rollno :</label>
                        <br><input type="text" name="rollno" id="rollno"  value="" ></h2>
                        <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter your email-address :</label>
                        <br><input type="text" name="email" id="email"  value="" ></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your Date of Birth :</label>
                        <br><input type="text" name="birthdate" id="birthdate"  value=""
></h2>
                                <h2 style= "outline-style: solid; outline-color: black;"><label
style= "color: #FFE8B6;" > Enter your permanent address :</label>
                        <br><input type="text" name="address" id="address"  value=""
></h2>
                                        <h2 style= "outline-style: solid; outline-color:
black;"><label style= "color: #FFE8B6;" > Enter your gender :</label>
                        <br><input type="text" name="gender" id="gender"  value="" ></h2>
                        <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter your course :</label>
                        <br><input type="text" name="course" id="course"  value="" ></h2>


                <br><h2><input type="submit" value="SUBMIT"></h2>
                                </h1>
                </form>
        </body>
</html>
```

//Print_StudBean.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>StudentBean</title>
  </head>
  <body>
    <jsp:useBean id="u" scope="session"
class="Students.StudentBean"></jsp:useBean>
    <jsp:setProperty name="u" property="*" />
    You have entered the below details:<br></b></h2><h3>
    NAME: <jsp:getProperty name="u" property= "name" /><br>
    ROLL NO: <jsp:getProperty name="u" property= "rollno" /><br>
    BIRTH DATE: <jsp:getProperty name="u" property= "birthdate" /><br>
    GENDER: <jsp:getProperty name="u" property= "gender" /><br>
    EMAIL: <jsp:getProperty name="u" property= "email" /><br>
    PERMANENT ADDRESS: <jsp:getProperty name="u" property= "address" /><br>
    COURSE: <jsp:getProperty name="u" property= "course" /><br>
  </body>
</html>
```

//OUTPUT


DETAILS
Enter your name :
Diya
Enter your rollno :
15
Enter your email-address :
diya@gmail.com
Enter your Date of Birth :
January 12, 2000
Enter your permanent address :
Uttarakhand
Enter your permanent address :
Uttarakhand
Enter your gender :
F
Enter your course :
Computer Science
SUBMIT

You have entered the below details:

NAME: Diya
ROLL NO: 15
BIRTH DATE: January 12, 2000
GENDER: F
EMAIL: diya@gmail.com
PERMANENT ADDRESS: Uttarakhand
COURSE: Computer Science

JDBC

1. Create a Student and Results Database and perform the following operations using JDBC programs.

a. Find total number of students.
b. Print average marks for each subject input by user.
c. Find the name of the student getting the highest marks.
d. Find no of students getting first, second and third division.
e. Find subject wise toppers.
f. Find the average marks.
g. Find the student getting second highest marks.

//CODE

```
package studentsdatabase;

import java.io.*;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
```

```
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;

public class StudentsDatabase
{
   public static void main(String args[])
   {
         Connection con;
          Statement state;
          ResultSet rs;
          int ch, max;

          try
          {
          Class.forName("com.mysql.cj.jdbc.Driver");
          con
=DriverManager.getConnection("jdbc:mysql://localhost:3306/IT_STUDENTS","roo
t", "root" );

          do
          {
             System.out.println("");
             System.out.println("Menu:");
             System.out.println("1. Display total number of students.");
             System.out.println("2. Display average marks for each subject.");
             System.out.println("3. Display the name of the student getting highest
marks.");
             System.out.println("4. Display the number of students getting first,
second and third division.");
             System.out.println("5. Display subject wise toppers.");
             System.out.println("6. Display the average marks.");
             System.out.println("7. Display the name of the student getting second
highest marks..");
             System.out.println("8. Display all the records from the table.");
             System.out.println("9. Exit");
             System.out.println("Enter your choice: ");
```

```
        BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
        ch=Integer.parseInt(br.readLine());

        switch(ch)
        {
           case 1:
           state=con.createStatement();
           String sql="select count(*) as total_stud from student";
           rs=state.executeQuery(sql);
           while(rs.next())
           {
             System.out.println("The total number of students
is:"+rs.getString("total_stud"));
           }

           break;

           case 2:

           System.out.println("Enter the subject for which you wish to find
average marks: ");
           String subject=br.readLine();

           if("ENGLISH".equals(subject))
           {
           state=con.createStatement();
           sql="SELECT avg(ENGLISH) as avg_mark from results";
           rs=state.executeQuery(sql);
           while(rs.next())
           {

       System.out.println("The average marks of all students in English:
"+rs.getString("avg_mark"));
           }
           }

           else if("HINDI".equals(subject))
           {
```

```
                state=con.createStatement();
                sql="SELECT avg(HINDI) as avg_mark from results";
                rs=state.executeQuery(sql);
                while(rs.next())
                {
                  System.out.println("The average marks of all students in Hindi:
"+rs.getString("avg_mark"));
                }
                }

                else if("MATHS".equals(subject))
                {
                state=con.createStatement();
                sql="SELECT avg(MATHS) as avg_mark from results";
                rs=state.executeQuery(sql);
                while(rs.next())
                {
                  System.out.println("The average marks of all students in Maths:
"+rs.getString("avg_mark"));
                }
                }

                else if("JAVA".equals(subject))
                {
                state=con.createStatement();
                sql="SELECT avg(JAVA) as avg_mark from results";
                rs=state.executeQuery(sql);
                while(rs.next())
                {
                  System.out.println("The average marks of all students in JAVA:
"+rs.getString("avg_mark"));
                }
                }

                else
                {
                state=con.createStatement();
                sql="SELECT avg(PHP) as avg_mark from results";
                rs=state.executeQuery(sql);
```

```
                while(rs.next())
                {
                  System.out.println("The average marks of all students in PHP:
        "+rs.getString("avg_mark"));
                }
                }

                break;

                case 3:
                 state=con.createStatement();
                 sql="select name,rollno,(sum(ENGLISH+HINDI+MATHS+JAVA+PHP))
        as total from results inner join student on results.rollno=student.roll group by
        rollno order by total desc";

            rs=state.executeQuery(sql);
            rs.next();
            String name=rs.getString(1);
            int no=rs.getInt(2);
            int marks= rs.getInt(3);
            System.out.println("The student with highest marks is: "+name + " , Roll
Number: " + no + " and Marks: " + marks);
            break;

            case 4:
            state=con.createStatement();
            sql="select  count(division) from results where division='I' ";
            ResultSet rs1=state.executeQuery(sql);
            rs1.next();
            int c1=rs1.getInt(1);
            System.out.println("The number of students getting first division is: "+c1);

            sql="select  count(division) from results where division='II' ";
            rs1=state.executeQuery(sql);
            rs1.next();
            int c2=rs1.getInt(1);
            System.out.println("The number of students getting second division is: "+c2);

            sql="select  count(division) from results where division='III' ";
```

```
rs1=state.executeQuery(sql);
rs1.next();
int c3=rs1.getInt(1);
System.out.println("The number of students getting third division is: "+c3);

 break;

case 5:
System.out.println("Enter Subject for which you wish to find the topper: ");
subject=br.readLine();

if("ENGLISH".equals(subject))
{
state=con.createStatement();
sql="select rollno from results group by rollno order by ENGLISH desc";

rs=state.executeQuery(sql);
rs.next();
max=rs.getInt(1);

   System.out.println("The ENGLISH subject topper's RollNo is: "+max);
}

else if("HINDI".equals(subject))
{
state=con.createStatement();
sql="select rollno from results group by rollno order by HINDI desc";

rs=state.executeQuery(sql);
rs.next();
 max=rs.getInt(1);
   System.out.println("The HINDI Subject Topper's RollNo is: "+max);
}

else if("MATHS".equals(subject))
{
    state=con.createStatement();
    sql="select rollno from results group by rollno order by MATHS desc";
```

```
   rs=state.executeQuery(sql);
   rs.next();
   max=rs.getInt(1);
   System.out.println("The MATHS subject topper's RollNo is: "+max);
}

else if("JAVA".equals(subject))
{
   state=con.createStatement();
   sql="select rollno from results group by rollno order by JAVA desc";
   rs=state.executeQuery(sql);
   rs.next();
   max=rs.getInt(1);
  System.out.println("The JAVA subject topper's RollNo is: "+max);
}

 else
{
   state=con.createStatement();
   sql="select rollno from results group by rollno order by PHP desc";

   rs=state.executeQuery(sql);
   rs.next();
   max=rs.getInt(1);
   System.out.println("The PHP subject topper's RollNo is: "+max);
}
break;

case 6:

state=con.createStatement();
String sql1="SELECT avg(ENGLISH) as avg_mark from results";
rs1=state.executeQuery(sql1);
rs1.next();
float a1=rs1.getFloat(1);
String sql2="SELECT avg(HINDI) as avg_mark from results";
ResultSet rs2=state.executeQuery(sql2);
rs2.next();
float a2=rs2.getFloat(1);
```

```
String sql3="SELECT avg(MATHS) as avg_mark from results";
ResultSet rs3=state.executeQuery(sql3);
rs3.next();
float a3=rs3.getFloat(1);
String sql4="SELECT avg(JAVA) as avg_mark from results";
ResultSet rs4=state.executeQuery(sql4);
rs4.next();
float a4=rs4.getFloat(1);
String sql5="SELECT avg(PHP) as avg_mark from results";
ResultSet rs5=state.executeQuery(sql5);
rs5.next();
float a5=rs5.getFloat(1);
float avg=(a1+a2+a3+a4+a5)/5;
System.out.println("The average marks of all subjects is:"+avg);

break;

case 7:

state=con.createStatement();
sql="select name,rollno, max(ENGLISH+HINDI+MATHS+JAVA+PHP) as total
from results inner join student on results.rollno=student.roll where
(ENGLISH+HINDI+MATHS+JAVA+PHP)< (select
max(ENGLISH+HINDI+MATHS+JAVA+PHP) from results) group by
(ENGLISH+HINDI+MATHS+JAVA+PHP) order by total desc";

rs=state.executeQuery(sql);
rs.next();
name=rs.getString(1);
max=rs.getInt(2);
no=rs.getInt(3);
System.out.println("The student with second highest marks is: "+name+ " ,
Roll no: "+max + " and Marks: "+no);
break;

case 8:
state=con.createStatement();
sql="select * from student";
rs=state.executeQuery(sql);
```

```
            System.out.println("\t ROLLNO"+"\t"+ "NAME"+"\t"+ "AGE"+"\t"+"GENDER")
            while(rs.next())
            {
               System.out.println("\n");

               System.out.print("\t" +rs.getInt(1));
               System.out.print("\t" +rs.getString(2));
               System.out.print("\t" +rs.getInt(3));
               System.out.print("\t" +rs.getString(4));

            }
            System.out.println("\n");
             sql="select * from results";
            rs=state.executeQuery(sql);
            System.out.println("\t ROLLNO"+"\t"+ "ENGLISH"+"\t"+
"HINDI"+"\t"+"MATHS"+"\t"+ "JAVA" +"\t"+"PHP"+"\t"+ "DIVISION")
            while(rs.next())
            {
               System.out.println("\n");

               System.out.print("\t" +rs.getInt(1));
               System.out.print("\t" +rs.getFloat(2));
               System.out.print("\t" +rs.getFloat(3));
               System.out.print("\t" +rs.getFloat(4));
               System.out.print("\t" +rs.getFloat(5));
               System.out.print("\t" +rs.getFloat(6));
               System.out.print("\t" +rs.getString(7));

            }

            break;

            case 9:
            System.exit(0);

            default:
            System.out.println("Invalid Choice");
            break;
         }
```

```
        }while(ch!=9);

      }
        catch(ClassNotFoundException e)
        {
           System.out.println(e.getMessage());
        }
        catch (SQLException ex)
        {
           System.out.println(ex.getMessage());
        }
        catch (IOException ex)
        {
           Logger.getLogger(PrepStmnt.class.getName()).log(Level.SEVERE, null, ex);
        }
   }
}
```

//OUTPUT

```
Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
1
The total number of students is:5

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
```

5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
2
Enter the subject for which you wish to find average marks:
MATHS
The average marks of all students in Maths: 76.500000

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
3
The student with highest marks is: DIYA , Roll Number: 1 and Marks: 471

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
4
The number of students getting first division is: 3
The number of students getting second division is: 1

The number of students getting third division is: 1

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
5
Enter Subject for which you wish to find the topper:
HINDI
The HINDI Subject Topper's RollNo is: 10

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
6
The average marks of all subjects is:83.58

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.

7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
7
The student with second highest marks is: ANJALI , Roll no: 10 and Marks: 454

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit
Enter your choice:
8

ROLLNO	NAME	AGE	GENDER
1	DIYA	20	F
2	MISHIKA	20	F
3	CHAKSHITA	20	F
8	PREET	21	M
10	ANJALI	22	F

ROLLNO	ENGLISH	HINDI	MATHS	JAVA	PHP	DIVISION
1	95.0	95.0	90.0	93.0	98.0	I
2	85.0	85.0	80.0	83.0	88.0	II
3	75.0	75.0	70.0	73.0	78.0	III

8	50.0	80.0	70.0	83.5	89.0	I
10	100.0	98.5	72.5	93.5	89.5	I

Menu:
1. Display total number of students.
2. Display average marks for each subject.
3. Display the name of the student getting highest marks.
4. Display the number of students getting first, second and third division.
5. Display subject wise toppers.
6. Display the average marks.
7. Display the name of the student getting second highest marks..
8. Display all the records from the table.
9. Exit

Enter your choice:
9
BUILD SUCCESSFUL (total time: 23 seconds)

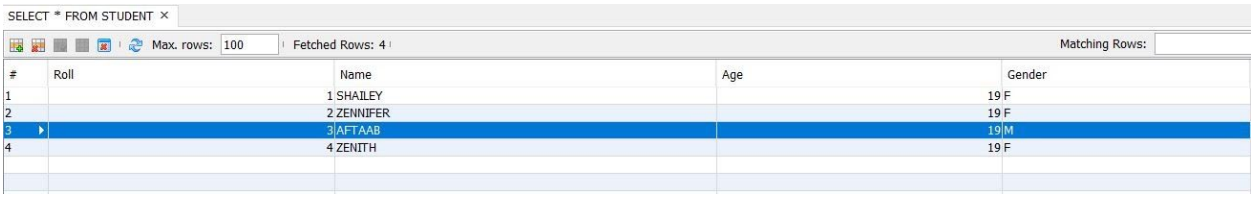
2. Create a procedure in MySQL to count the total number of rows using CallableStatement.

//CODE

```
/*
use stud_emp;
DELIMITER //
CREATE PROCEDURE getTotal (OUT total_num INT)
BEGIN
SELECT COUNT(*)
INTO total_num
FROM student;
END
//
DELIMITER;
*/

package jdbc_assignment2;

import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Types;

public class Ques1 {
```

```
public static void main(String[] args)
{
   try
   {
     Class.forName("com.mysql.cj.jdbc.Driver");
      Connection conn= DriverManager.getConnection
      ("jdbc:mysql://localhost:3306/stud_emp","root", "root" );
      CallableStatement stmt=conn.prepareCall("{call countTotal(?)}");
      stmt.registerOutParameter(1,Types.INTEGER);
      stmt.execute();
      System.out.println("No of rows in the table Student= "+stmt.getInt(1));
      stmt.close();
       }
            catch(Exception e)
            {
               System.out.println(e);
            }
        }

      }
```

//STUDENT TABLE



//OUTPUT

No of rows in the table Student= 4

JAVASCRIPT

1. Create a Student and Results Database and perform the following operations using JDBC programs.

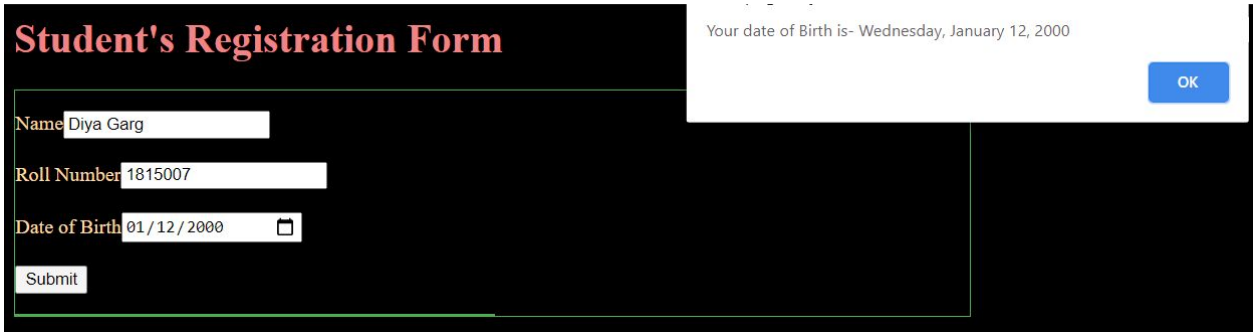
//studRegForm.html

```
<html>
<head>
<style>
body {
  background: black;
  align:center;
}
.content {
  max-width: 300px;
  margin: auto;
  background: white;
  padding: 10px;
}
div {
  width: 50%;
  border: 1px solid #4CAF50;
text-align="center";
}
```

```
</style>
<h1 style="color: F18484;">Student's Registration Form</h1>
</head>

<div>
<body>

<p id="p1" style="color: F18484;"></p>

<form name="RegForm" onsubmit="return validation()" method="get">
<label style="color: FFDAA7;">Name</label><input  name= "text" id="fname" name="fname" placeholder= "Enter your name" pattern="[a-zA-Z ]*" title="Enter alphabets only"></input>
<br><br>
<label style="color: FFDAA7;">Roll Number</label><input name= "text" id="rollno" name="rollno" placeholder= "Enter your roll number" pattern="[0-9]*" title="Enter digits only" minlength="7" maxlength="7"></input>
<br><br>
<label style="color: FFDAA7;">Date of Birth</label><input type= "date" id="dob" name="date" placeholder= "dd/mm/yy" ></input>
<br><br>
<input type="submit" value="Submit" name="submit">


</form>
<div id="Result"></div>

<script>
function validation()
{
        var x=document.getElementsByTagName("input");
        var z=document.forms["RegForm"]["dob"].value;
        var d=new Date(z);
        var d1=d.getDate();
        var y=d.getFullYear();
```

```
		var months = ["January","February","March","April","May","June","July","August","September","October","November","December"];
		var m = months[d.getMonth()];

		var days=["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"];
		var day=days[d.getDay()];

		for(var i=0; i<x.length; i++)
		if (x [i].value == "" )
		{
			 alert("Details must be filled out.");
			document.forms["RegForm"]["fname"].focus();
			return false;
		}
		else
		{
			alert("Your date of Birth is- " + day + ", "+ m + " " + d1 + ", "+ y);
			return true;
		}
	}
	</script>

	</body>
	</html>
```

//OUTPUT



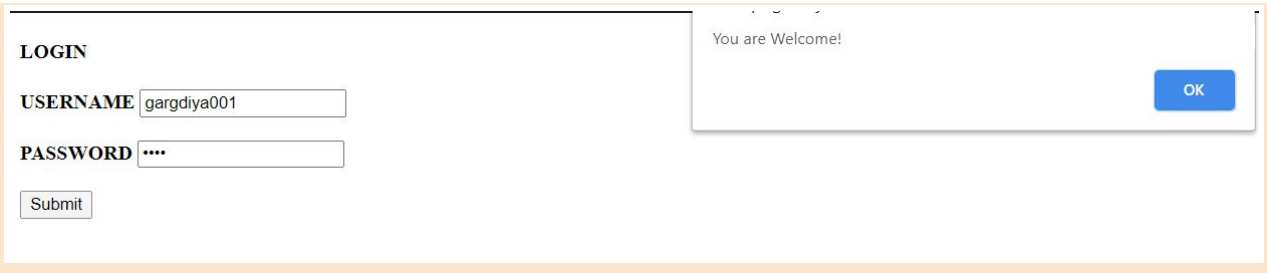
2. Implement a Static Password Protection.

//staticPassValid.html

```
<!DOCTYPE html>
<html>
<head>
<script>
function validation()
{
	var x = document.forms["SampleForm2"]["username"].value;
	var y = document.forms["SampleForm2"]["password"].value;
	if (x == "gargdiya001" && y=="g*rg")
	{
		alert("You are Welcome!");

		return false;
	}
	else
	{
		alert("The username or password is incorrect.");s
```

```
        }
}
</script>
</head>

<body>

<form name="SampleForm2"  onsubmit="return validation()" >
<h4>LOGIN</h4>

<label><b>USERNAME  </b></label>
<input type="text" placeholder="Enter username" id= "username"
name="username" required>
<br><br>
<label><b>PASSWORD  </b></label>
<input type="password" placeholder="Enter Password" id= "password"
name="password" required>
<br><br>
 <input type="submit" value="Submit" name="submit">
</form>

</body>
</html>
```

//OUTPUT



3. Implement a program to sort an array using Bubble Sort.

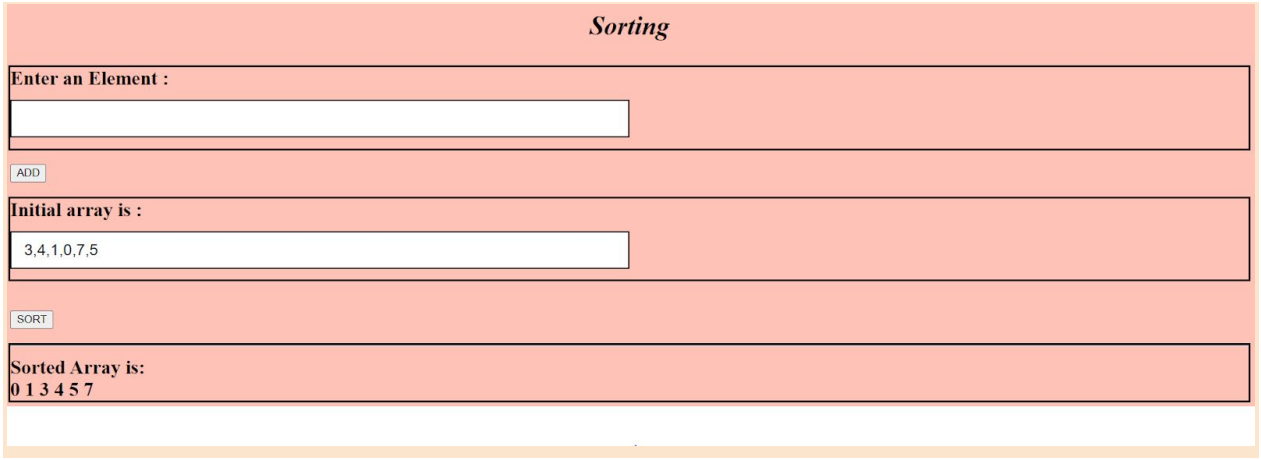
//sorting.html

```
<html>
<head>
<style>
input[type= "text"]{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}
</style>

<script>
var arr=[];
function add()
{
```

```
        var a=document.getElementById('push');
        var b=document.getElementById('pop');
        arr.push(a.value);
        b.value=arr;
        a.value="";
}

function display()
{
  var e = "<hr/>Sorted Array is:  </br>";

  for (var y=0; y<arr.length; y++)
  {
    e += arr[y] + " ";


  }
  document.getElementById("final").innerHTML = e;
}

function swap(arr, first_Index, second_Index){
   var temp = arr[first_Index];
   arr[first_Index] = arr[second_Index];
   arr[second_Index] = temp;
}
function sorting()
{

        var len = arr.length,
      i, j, stop;

   for (i=0; i < len; i++){
      for (j=0, stop=len-i; j < stop; j++){
         if (arr[j] > arr[j+1]){
            swap(arr, j, j+1);
         }
      }
```

```
   }
display();
}

</script>
        </head>
        <body style="background-color:#FFC2B6;">
        <h1><i><center><p >Sorting</p></center></i></h1>
                <form>
                        <h2 style= "outline-style: solid; outline-color: black;">Enter
an Element :
                        <br><input type="text" name="push" id="push" ></h2>
                        <input type="button" value="ADD" onClick="add()">

                        <h2 style= "outline-style: solid; outline-color: black;">Initial
array is :
                        <br><input type="text" name="pop" id="pop" ></h2><br>
                        <input type="button" value="SORT" onClick="sorting()">
                        <h2 style= "outline-style: solid; outline-color: black;">
                </form>

<div id="final"></div>
        </body>
</html>
```

//OUTPUT



4. Write a JavaScript program to implement stack methods- push and pop.

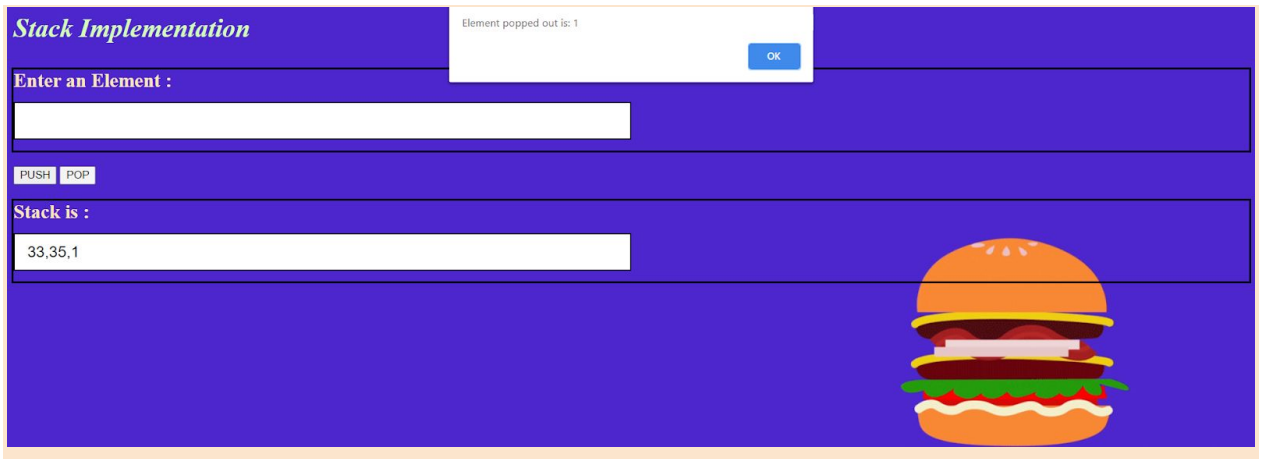
//UserStack.html

```
<html>
<head>
<style>
input[type= "text"]
{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
```

```
  font-size: 80%;
}

body {
  background: #f3f3f3 url('giphy_burger.gif') no-repeat right bottom;
}
</style>

<script>
var arr=[];
function stackpush()
{
	var a=document.getElementById('push');
	var b=document.getElementById('pop');
	arr.push(a.value);
	b.value=arr;
	a.value="";
}

function stackpop()
{
	var len=arr.length;
	if(len==0)
	{
		alert("Stack is already Empty!");
	}
	else
	{
		alert("Element popped out is: "+ arr.pop(len));
		var c=document.getElementById('pop');
		c.value=arr;
	}
}

</script>
	</head>
	<body style="background-color:#4d27cd;">
```

```
        <h1><i><p style= "color: #D6FFB6;">Stack
        Implementation</p></i></h1>
                <form>
                        <h2 style= "outline-style: solid; outline-color: black;"><label
                        style= "color: #FFE8B6;" > Enter an Element :</label>
                        <br><input type="text" name="push" id="push" ></h2>
                        <input type="button" value="PUSH" onClick="stackpush()">
                        <input type="button" value="POP" onClick="stackpop()">
                        <h2 style= "outline-style: solid; outline-color: black;"><label
                        style= "color: #FFE8B6;" > Stack is :</label>
                        <br><input type="text" name="pop" id="pop" ></h2>
                        </h1>
                </form>
        </body>
</html>
```

//OUTPUT



5. Write a JavaScript program to use setTimeOut and setInterval.

//setTimeOut&setInterval.html

```
<html>
<head><h1>SURPRISE!</h1></head>
<body>
<script>

function changeColor1( )
{
  document.getElementById("p1").style.color = "#FF5E9B";
  setTimeout ( "changeColor2( )", 1000 );
}
```

```
function changeColor2( )
{
  document.getElementById("p1").style.color = "#28B0ED";
  setTimeout ( "changeColor3( )", 1000 );
}

function changeColor3( )
{
  document.getElementById("p1").style.color = "#14D665";
  setTimeout ( "changeColor4( )", 1000 );
}

function changeColor4( )
{
document.getElementById("p1").style.color = "#000000";
}

var imgObj = null;
var animate ;

function init()
{
        imgObj = document.getElementById('myImage');
                imgObj.style.position= 'relative';
                imgObj.style.left = '0px'
}

function moveRight()
{
                imgObj.style.left = parseInt(imgObj.style.left) + 1 + 'px';
                animate = setInterval(moveRight, 1000);    // call moveRight in
                1000milliseconds
}

window.onload = init;
</script>
```

```
<p id="p1" style="font-size:20px;"><b>Click the button below to see the color change and enjoy.</b></p>
<input type="button" name="clickMe" id="colourButton" value="Click me!" onclick="changeColor1()"/>
<br><br>

<img id = "myImage" src = "im1.jpg" alt="Tracks" width="500" height="200"/>
 <p style="font-size:20px; color:blue;"><i>Click the button below to start animation!</i></p>
<input type = "button" value = "Start" onclick = "moveRight();" />

</body>
</html>
```

//OUTPUT

SURPRISE!

Click the button below to see the color change and enjoy.

Click me!

Click the button below to start animation!

Start

6. Implement the ques1 of chapter 10.

//InfonetServices.html

```
<html>
<head>
<style>
body {
  background: black;

    align:center;
}
.content {
  max-width: 300px;
  margin: auto;
  background: white;
  padding: 10px;
}
div {
  width: 50%;
  border: 1px solid #4CAF50;
text-align="center";
}
</style>
<h1 style="color: F18484;">INFONET SERVICES</h1>
</head>

<div>
<body>

<form name="InfonetServices" onsubmit="return validation()" method="get">
<input  name= "text" id="fname" name="fname" placeholder= "Enter your first
name" pattern="[a-zA-Z]*" title="Enter alphabets only"></input><label
style="color: FFDAA7;">First Name</label>
<br><br>
<input name= "text" id="lname" name="lname" placeholder= "Enter your last
name" pattern="[a-zA-Z]*" title="Enter alphabets only"></input><label
style="color: FFDAA7;">Last Name</label>
<br><br>
```

```
<input name= "text" id="email" name="email" placeholder= "Enter your e-mail address" ></input><label style="color: FFDAA7;">Email address</label>
<br><br>
<input name= "text" id="address" name="address" placeholder= "Enter your address" pattern="[a-zA-Z]*" title="Enter alphabets only"></input><label style="color: FFDAA7;">Address</label>
<br><br>
<input name= "text" id="city" name="city" placeholder= "Enter your city's name" pattern="[a-zA-Z]*" title="Enter alphabets only"></input><label style="color: FFDAA7;">City</label>
<br><br>
<input name= "text" id="state" name="state" placeholder= "Enter your state's name" pattern="[a-zA-Z]*" title="Enter alphabets only"></input><label style="color: FFDAA7;">State</label>
<br><br>
<input name= "text" id="code" name="code" placeholder= "Enter the postal code" pattern="[0-9]*" title="Enter digits only"></input><label style="color: FFDAA7;">Postal Code</label>
<br><br>
<input name= "text" id="country" name="country" placeholder= "Enter your country's name" pattern="[a-zA-Z]*" title="Enter alphabets only"></input><label style="color: FFDAA7;">Country</label>
<br><br>

<label style="color: FFDAA7;"><i><b>Please choose the most appropriate statement</b></i></label><br>
<input type="radio" id="r1" name="stmnt" value="I regularly purchase items online">
 <label for="r1" style="color: EAFFE8;">I regularly purchase items online</label><br>

 <input type="radio" id="r2" name="stmnt" value="I purchase items online on ocassions">
 <label for="r2" style="color: EAFFE8;">I purchase items online on ocassions</label><br>
```

```
<input type="radio" id="r3" name="stmnt" value="I have not purchased anything online, but I would consider it">
 <label for="r3" style="color: EAFFE8;">I have not purchased anything online, but I would consider it</label><br>

 <input type="radio" id="r4" name="stmnt" value="I prefer to shop in real stores">
 <label for="r4" style="color: EAFFE8;">I prefer to shop in real stores</label><br>
<br>

<label style="color: FFDAA7;"><i><b>I'm interested in(choose all that apply)</b></i></label><br>
<input type="checkbox" id="c1" name="stmnt1" value="Hiking">
 <label for="c1" style="color: EAFFE8;">Hiking</label><br>

 <input type="checkbox" id="c2" name="stmnt2" value="Mountain Biking">
 <label for="c2" style="color: EAFFE8;">Mountain Biking</label><br>

<input type="checkbox" id="c3" name="stmnt3" value="Camping">
 <label for="c3" style="color: EAFFE8;">Camping</label><br>

 <input type="checkbox" id="c4" name="stmnt4" value="Rock Climbing">
 <label for="c4" style="color: EAFFE8;">Rock Climbing</label><br>

 <input type="checkbox" id="c4" name="stmnt5" value="Off-Road 4WD">
 <label for="c4" style="color: EAFFE8;">Off-Road 4WD</label><br>

<input type="checkbox" id="c4" name="stmnt6" value="c">
 <label for="c4" style="color: EAFFE8;">Cross-country Skiing</label><br>
<br>

<label style="color: FFDAA7;" for= "learnt"><i><b>I learnt about this site from</b></i></label><br>
<select name="leanrt" id="cars">
   <option value="PrintAds">PrintAds</option>
   <option value="W3Schools">W3Schools</option>
```

```
   <option value="JavaTutorials">JavaTutorials</option>
   <option value="Self">Self</option>
  </select>
<br><br>

<label style="color: FFDAA7;" ><i><b>Comments</b></i></label><br>
<textarea rows= "4" cols= "30" placeholder="Please type any comments
here"></textarea>
<br><br>

<input type="submit" value="Submit" name="submit">
<input type="reset" value="Start Over" >

</form>

<script>
function validation()
{
        var x=document.getElementsByTagName("input");
        for(var i=0; i<x.length; i++)
        if (x [i].value == "" )
        {
                 alert("Details must be filled out.");
                document.forms["InfonetServices"]["fname"].focus();
                return false;
        }
        return true;
}
</script>

</body>
</html>
```

//OUTPUT

INFONET SERVICES

Diya First Name

Garg Last Name

diya@gmail.com Email address

Vijay Block Address

Laxmi Nagar City

New Delhi State

110092 Postal Code

India Country

Please choose the most appropriate statement

- ○ I regularly purchase items online
- ◉ I purchase items online on ocassions
- ○ I have not purchased anything online, but I would consider it
- ○ I prefer to shop in real stores

I'm interested in(choose all that apply)

- ☑ Hiking
- ☐ Mountain Biking
- ☑ Camping
- ☐ Rock Climbing
- ☐ Off-Road 4WD
- ☐ Cross-country Skiing

I learnt about this site from

Self

Comments

Please type any comments here

Submit Start Over

JSP

1.Display the pattern using scriplets and forEach loop.

//Pattern.html

```
<!DOCTYPE html>

<html>
   <head>
      <title>PATTERN</title>
      <meta charset="UTF-8">
      <meta name="viewport" content="width=device-width, initial-scale=1.0">
   </head>
 <style>

   body {
      background: #8D1849 }
   section {
      background: black;
      color: white;
      border-radius: 1em;
      padding: 1em;
      position: absolute;
      top: 50%;
      left: 50%;
      margin-right: -50%;
      transform: translate(-50%, -50%) }

      .button {
         background-color: #28686B;
         border: none;
         color: white;
         padding: 15px 32px;
         text-align: center;
         text-decoration: none;
         display: inline-block;
         font-size: 16px;
```

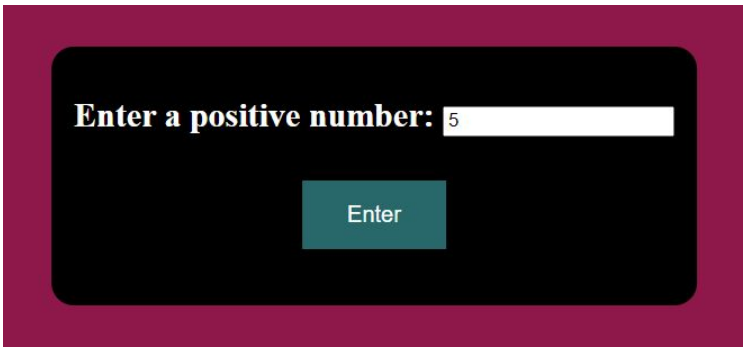
```
        margin: 4px 2px;
        cursor: pointer;
      }
 </style>

 <section>
  <body>
    <div>
       <form action="Pattern.jsp" method="get">
          <center><h1>
               Enter a positive number: <input type ="text" name="num"
               pattern="[0-9]+" title="Should be a number only" required><br><br>
          <input type ="submit" class= "button" value="Enter">
          </center>
       </form>
    </div>
  </body>
 </section>
</html>
```

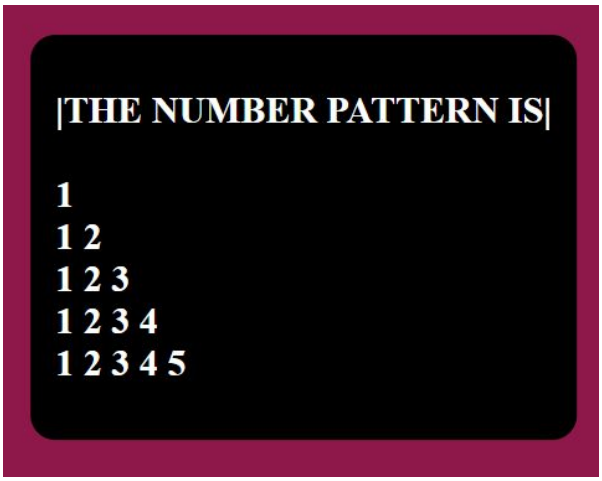
//Pattern.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <title>PATTERN</title>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

   </head>
   <style>

   body {
      background: #8D1849 }
   section {
      background: black;
      color: white;
      border-radius: 1em;
```

```
        padding: 1em;
        position: absolute;
        top: 50%;
        left: 50%;
        margin-right: -50%;
        transform: translate(-50%, -50%) }

    </style>
    <section>
    <body>
        <h2>
        <%
            out.println("|THE NUMBER PATTERN IS| <br>");
              out.print(" <br> ");
            int n= Integer.parseInt(request.getParameter("num"));
            int val;
            for(int i=0;i<n;i++)
            {
               val=1;
            for(int j=0;j<=i;j++)
            {

                  out.print(val+"        ");
                  val++;
             }

            out.print("<br>");

}
            %>
    </body>
    </section>
</html>
```

//OUTPUT





2. Make two files as follows:

a. main.html: shows 2 text boxes and 3 radio buttons with values "addition", "subtraction" and "multiplication"
b. operate.jsp: depending on what the user selects perform the corresponding function (Give two implementations: using request.getParameter() and using expression language)

//main.html

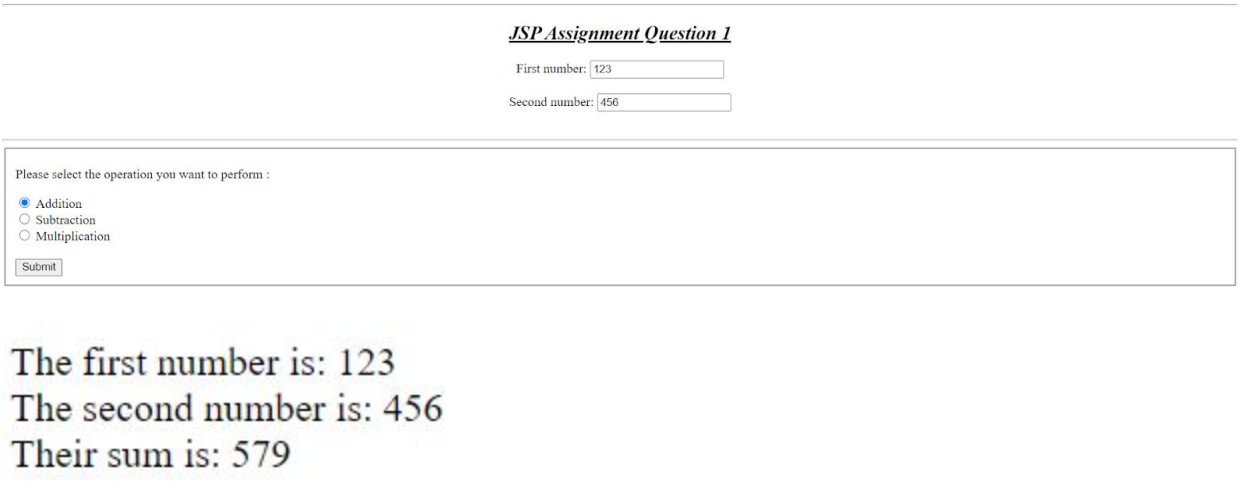
```
<html>
<body>
<hr>
<center>
   <h2><i><u>JSP Assignment Question 1</u></i></h2>
<form action="operate.jsp" method = "POST">
<label for="fn">First number:</label>
<input type="text" id="fn" name="fn"><br><br>
<label for="sn">Second number:</label>
<input type="text" id="sn" name="sn"><br><br>
</center>
<hr>
<fieldset>
<p>Please select the operation you want to perform :</p>
 <input type="radio" id="add" name="calc" value="Addition">
 <label for="add">Addition</label><br>
 <input type="radio" id="sub" name="calc" value="Subtraction">
 <label for="sub">Subtraction</label><br>
 <input type="radio" id="mul" name="calc" value="Multiplication">
 <label for="mul">Multiplication</label>
<br><br>
<input type="submit" value="Submit">
</fieldset>
</form>
</body>
</html>
```

//operate.jsp

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

<%! String[] selected; %>
<% String no1 = request.getParameter("fn");%>
<% String no2 = request.getParameter("sn");%>

<%
selected = request.getParameterValues("calc");

for(int i=0;i<selected.length;i++)
{
   if (selected[i].equals("Addition"))
   {
      int res =Integer.parseInt(no1)+ Integer.parseInt(no2);
      out.println("The first number is: "+no1+"<br>");
      out.println("The second number is: "+no2+"<br>");
      out.println("Their sum is: " + res);
   }
   if (selected[i].equals("Subtraction"))
   {
      int res =Integer.parseInt(no1)-Integer.parseInt(no2);
      out.println("The first number is: "+no1+"<br>");
      out.println("The second number is: "+no2+"<br>");
      out.println("Their difference is: " + res);
   }
   if (selected[i].equals("Multiplication"))
   {
      int res =Integer.parseInt(no1)*Integer.parseInt(no2);
      out.println("The first number is: "+no1+"<br>");
      out.println("The second number is: "+no2+"<br>");
      out.println("Their product is: " + res);
   }
}
%>
```

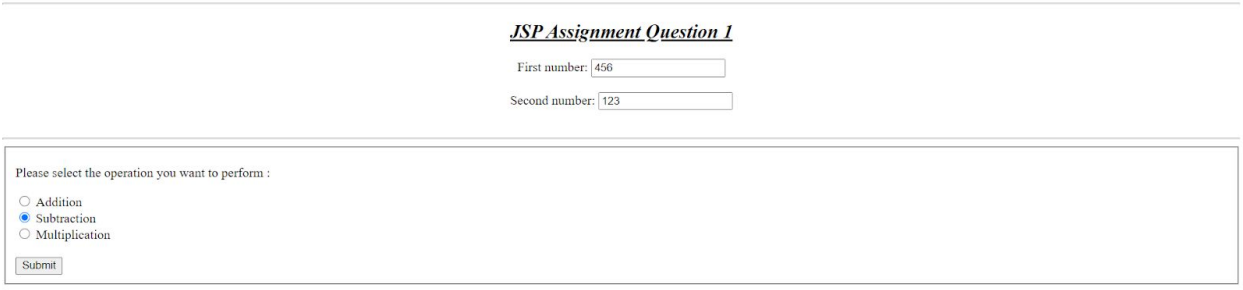
```
</body>
</html>
```

//Output : Addition

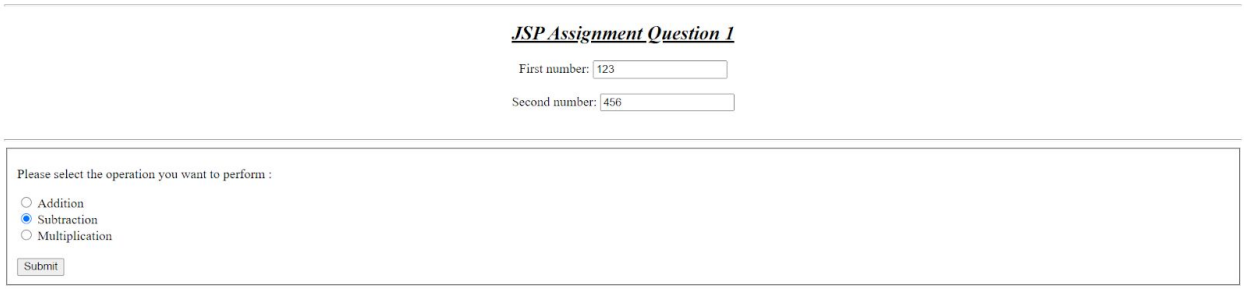


The first number is: 123
The second number is: 456
Their sum is: 579

//Output : Subtraction

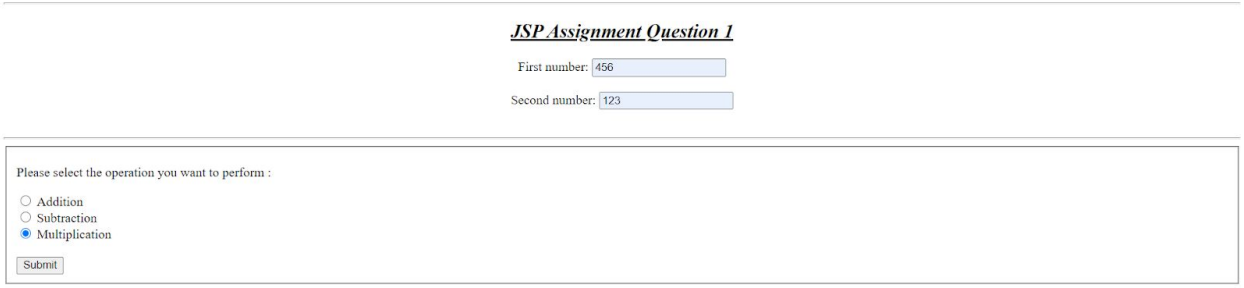


The first number is: 456
The second number is: 123
Their difference is: 333



The first number is: 123
The second number is: 456
Their difference is: -333

//Output : Multiplication



The first number is: 456
The second number is: 123
Their product is: 56088

3. Validation using two ways:

Validate User input entered in a form. The input must include Name, DOB, Email ID, Lucky Number, Favorite food etc. (Refer Chapter 8).

//METHOD 1- USING JSTL ACTIONS

//validate_jsp.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<title>User Info Entry Form</title>
</head>
<body bgcolor="white">
<form action="validate_jsp.jsp" method="post">
<input type="hidden" name="submitted" value="true">
<table>
<c:if test="${param.submitted && empty param.userName}">
<tr><td></td>
<td colspan="2"><font color="red">
Please enter your Name
</font></td></tr>
</c:if>
<tr>
<td>Name:</td>
<td>
<input type="text" name="userName"
value="<c:out value="${param.userName}" />">
</td>
</tr>
<c:if test="${param.submitted && empty param.birthDate}">
<tr><td></td>
<td colspan="2"><font color="red">
```

```
Please enter your Birth Date
</font></td></tr>
</c:if>
<tr>
<td>Birth Date:</td>
<td>
<input type="text" name="birthDate"
value="<c:out value="${param.birthDate}" />">
</td>
<td>(Use format yyyy-mm-dd)</td>
</tr>

<c:if test="${param.submitted && empty param.emailAddr}">
<tr><td></td>
<td colspan="2"><font color="red">
Please enter your Email Address
</font></td></tr>
</c:if>
<tr>
<td>Email Address:</td>
<td>
<input type="text" name="emailAddr"
value="<c:out value="${param.emailAddr}" />">
</td>
<td>(Use format name@company.com)</td>
</tr>
<c:if test="${param.submitted &&
param.gender != 'm' && param.gender != 'f'}">
<tr><td></td>
<td colspan="2"><font color="red">
Please select a valid Gender
</font></td></tr>
</c:if>
<tr>
<td>Gender:</td>
<td>
```

```
<c:choose>
<c:when test="${param.gender == 'f'}">
<input type="radio" name="gender" value="m">
Male<br>
<input type="radio" name="gender" value="f" checked>
Female
</c:when>
<c:when test="${param.gender == 'm'}">
<input type="radio" name="gender" value="m" checked>
Male<br>
<input type="radio" name="gender" value="f">
Female
</c:when>
<c:otherwise>
<input type="radio" name="gender" value="m">
Male<br>
<input type="radio" name="gender" value="f">
Female

</c:otherwise>
</c:choose>
</td>
</tr>
<c:if test="${param.submitted &&
(param.luckyNumber < 1 || param.luckyNumber > 100)}">
<tr><td></td>
<td colspan="2"><font color="red">
Please enter a Lucky Number between 1 and 100
</font></td></tr>
</c:if>
<tr>
<td>Lucky number:</td>
<td>
<input type="text" name="luckyNumber"
value="<c:out value="${param.luckyNumber}" />">
</td>
```

```
<td>(A number between 1 and 100)</td>
</tr>
<c:forEach items="${paramValues.food}" var="current">
<c:choose>
<c:when test="${current == 'z'}">
<c:set var="pizzaSelected" value="true" />
<c:set var="foodValid" value="true" />
</c:when>
<c:when test="${current == 'p'}">
<c:set var="pastaSelected" value="true" />
<c:set var="foodValid" value="true" />
</c:when>
<c:when test="${current == 'c'}">
<c:set var="chineseSelected" value="true" />
<c:set var="foodValid" value="true" />
</c:when>
<c:otherwise>
<c:set var="foodValid" value="false" />
</c:otherwise>
</c:choose>
</c:forEach>
<c:if test="${param.submitted && !foodValid}">
<tr><td></td>
<td colspan="2"><font color="red">
Please select only valid Favorite Foods

</font></td></tr>
</c:if>
<tr>
<td>Favorite Foods:</td>
<td>
<input type="checkbox" name="food" value="z"
${pizzaSelected ? 'checked' : ''}>Pizza<br>
<input type="checkbox" name="food" value="p"
${pastaSelected ? 'checked' : ''}>Pasta<br>
<input type="checkbox" name="food" value="c"
```

```
${chineseSelected ? 'checked' : ''}>Chinese
</td>
</tr>
<tr>
<td colspan="3">
<input type="submit" value="Send Data">
</td>
</tr>
</table>
</form>
</body>
</html>
```

//OUTPUT

Name:

Birth Date: (Use format yyyy-mm-dd)

Email Address: (Use format name@company.com)

Gender: ○ Male ○ Female

Lucky number: (A number between 1 and 100)

Favorite Foods: ☐ Pizza ☐ Pasta ☐ Chinese

Send Data

Please enter your Name

Name:

Please enter your Birth Date

Birth Date: (Use format yyyy-mm-dd)

Please enter your Email Address

Email Address: (Use format name@company.com)

Please select a valid Gender

Gender:
- ○ Male
- ○ Female

Please enter a Lucky Number between 1 and 100

Lucky number: (A number between 1 and 100)

Please select only valid Favorite Foods

Favorite Foods:
- ☐ Pizza
- ☐ Pasta
- ☐ Chinese

Send Data

Name: Diya

Birth Date: 2000-02-12 (Use format yyyy-mm-dd)

Email Address: diya@gmail.com (Use format name@company.com)

Gender:
- ○ Male
- ◉ Female

Lucky number: 7 (A number between 1 and 100)

Favorite Foods:
- ☑ Pizza
- ☐ Pasta
- ☐ Chinese

Send Data

//METHOD 2- USING BEANS

//UserInfoBean.java

```
package com;
import java.beans.*;
import java.io.Serializable;
public class UserInfoBean implements Serializable {
//properties
private String userName;
private String birthDate;
private String emailAddr;
private String gender;
private String luckyNumber;
private String[] food;
boolean birthDateValid = false;
boolean genderValid = false;
boolean userNameValid = false;
boolean foodValid = false;
boolean emailAddrValid = false;
boolean valid = false;
boolean luckyNumberValid = false;
boolean pizzaSelected=false;
boolean pastaSelected=false;
boolean chineseSelected=false;
public boolean isPizzaSelected() {
return pizzaSelected;
}
public void setPizzaSelected(boolean pizzaSelected) {
this.pizzaSelected = pizzaSelected;
}
public boolean isPastaSelected() {
return pastaSelected;
}
public void setPastaSelected(boolean pastaSelected) {
this.pastaSelected = pastaSelected;
```

```
}
public boolean isChineseSelected() {

return chineseSelected;
}
public void setChineseSelected(boolean chineseSelected) {
this.chineseSelected = chineseSelected;
}
public String getUserName() {
return (userName == null ? "" : userName);
}
public void setUserName(String userName) {
this.userName = userName;
}
public String getBirthDate() {
return (birthDate == null ? "" : birthDate);
}
public void setBirthDate(String birthDate) {
this.birthDate = birthDate;
}
public String getEmailAddr() {
return emailAddr;
}
public void setEmailAddr(String emailAddr) {
this.emailAddr = emailAddr;
}
public String getGender() {
return gender;
}
public void setGender(String gender) {
this.gender = gender;
}
public String getLuckyNumber() {
return luckyNumber;
}
public void setLuckyNumber(String luckyNumber) {
```

```
this.luckyNumber = luckyNumber;
}
public String[] getFood() {
return food;

}
public void setFood(String[] food) {
this.food = food;
}
public boolean isBirthDateValid() {
if (birthDate != null )
{
birthDateValid = true;
}
return birthDateValid;
}
public boolean isGenderValid() {
if(gender!=null)
{
if((gender.equals("f")||gender.equals("m")))
{
genderValid = true;
}
}
return genderValid;
}
public boolean isUserNameValid() {
if (userName!=null)
{
userNameValid = true;
}
return userNameValid;
}
public boolean isFoodValid() {
if (food == null) {
foodValid = false;
```

```
return foodValid;
}
for(int i=0;i<getFood().length;i++)
{
if (food[i].equals("c"))
{
chineseSelected=true;

}
else if (food[i].equals("p"))
{
pastaSelected=true;
}
else //if (food[i].equals("z"))
{
pizzaSelected=true;
}
}
foodValid = pastaSelected || pizzaSelected || chineseSelected;
return foodValid;
}
public boolean isEmailAddrValid() {
if (emailAddr.contains("@")&&emailAddr!=null ||emailAddr!="")
{
emailAddrValid = true;
}
return emailAddrValid;
}
public boolean isLuckyNumberValid() {
if (Integer.parseInt(luckyNumber) >=1 && Integer.parseInt(luckyNumber) <=100)
{
luckyNumberValid = true;
}
return luckyNumberValid;
}
public boolean isValid() {
```

```
valid = isBirthDateValid() && isEmailAddrValid() &&
isFoodValid() && isLuckyNumberValid() &&
isGenderValid() && isUserNameValid();
return valid;
}

}
```

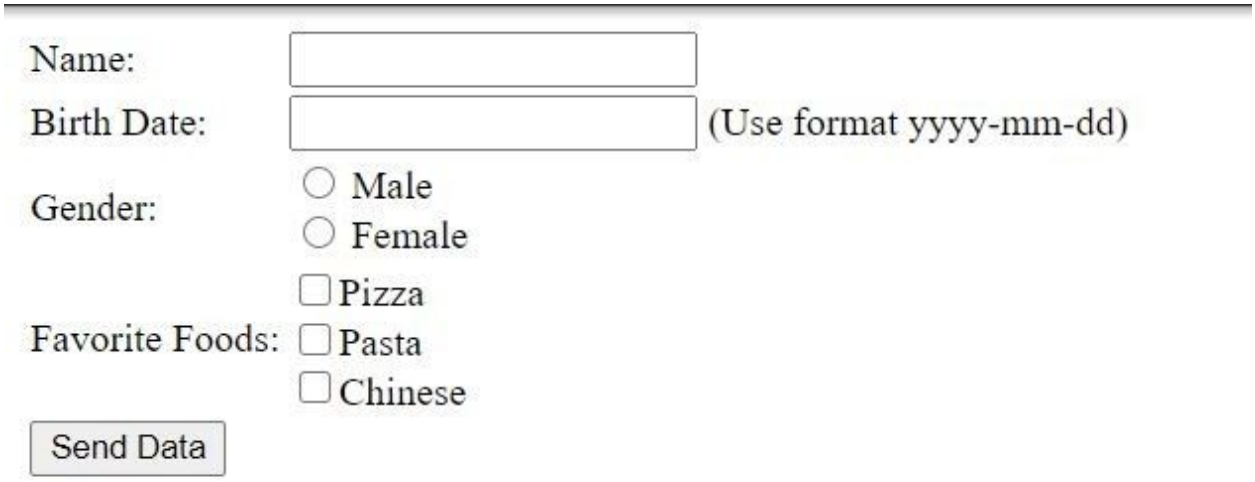
//validatebean_jsp.java

```
<%@ page contentType="text/html" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
<title>User Info Entry Form</title>
</head>
<body bgcolor="white">
<jsp:useBean id="userInfo"
class="com.UserInfoBean">
<jsp:setProperty name="userInfo" property="*" />
</jsp:useBean>
<form action="validatebean_jsp.jsp" method="post">
<input type="hidden" name="submitted" value="true">
<table>
<c:if test="${param.submitted && userInfo.userNameValid == false}">
<tr><td></td>
<td colspan="2"><font color="red">
Please enter your Name
</font></td></tr>
</c:if>
<tr>
<td>Name:</td>
<td>
<input type="text" name="userName"
value="<c:out value="${userInfo.userName}"/>">
</td>
```

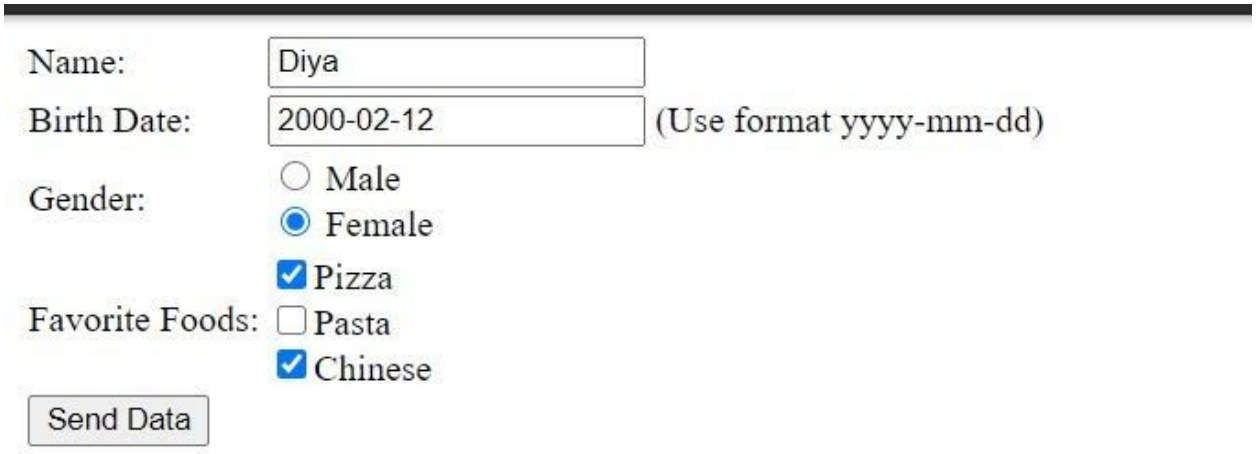
```
</tr>
<c:if test="${param.submitted && !userInfo.birthDateValid}">
<tr><td></td>
<td colspan="2"><font color="red">
Please enter a valid Birth Date
</font></td></tr>
</c:if>
<tr>
<td>Birth Date:</td>
<td>
<input type="text" name="birthDate"
value="<c:out value="${userInfo.birthDate}" />">
</td>
<td>(Use format yyyy-mm-dd)</td>
</tr>
<c:if test="${param.submitted && !userInfo.genderValid}">
<tr><td></td>
<td colspan="2"><font color="red">
Please select a valid Gender
</font></td></tr>
</c:if>

<tr>
<td>Gender:</td>
<td>
<c:choose>
<c:when test="${userInfo.gender == 'f'}">
<input type="radio" name="gender" value="m">
Male<br>
<input type="radio" name="gender" value="f" checked>
Female
</c:when>
<c:when test="${userInfo.gender == 'm'}">
<input type="radio" name="gender" value="m" checked>
Male<br>
<input type="radio" name="gender" value="f">
```

```
Female
</c:when>
<c:otherwise>
<input type="radio" name="gender" value="m">
Male<br>
<input type="radio" name="gender" value="f">
Female
</c:otherwise>
</c:choose>
</td>
</tr>
<c:if test="${param.submitted && !userInfo.foodValid}">
<tr><td></td>
<td colspan="2"><font color="red">
Please select only valid Favorite Foods
</font></td></tr>
</c:if>
<tr>
<td>Favorite Foods:</td>
<td>
<input type="checkbox" name="food" value="z"
${userInfo.pizzaSelected ? 'checked' : ''}>Pizza<br>
<input type="checkbox" name="food" value="p"
${userInfo.pastaSelected ? 'checked' : ''}>Pasta<br>
<input type="checkbox" name="food" value="c"
${userInfo.chineseSelected ? 'checked' : ''}>Chinese
</td>
</tr>
<tr>
<td colspan="3"> <input type="submit" value="Send Data"> </td>
</tr>
</table>
</form>
</body>
</html>
```

//OUTPUT


Name:
Birth Date:
(Use format yyyy-mm-dd)
Gender:
Male
Female
Favorite Foods:
Pizza
Pasta
Chinese
Send Data


Please enter your Name
Name:
Please enter a valid Birth Date
Birth Date:
(Use format yyyy-mm-dd)
Please select a valid Gender
Gender:
Male
Female
Please select only valid Favorite Foods
Favorite Foods:
Pizza
Pasta
Chinese
Send Data


Name:
Diya
Birth Date:
2000-02-12
(Use format yyyy-mm-dd)
Gender:
Male
Female
Favorite Foods:
Pizza
Pasta
Chinese
Send Data

4. Display :

Good Morning <uname>, Good Afternoon <uname> or Good Evening <uname> based on the current time of the day.

//Greetings.html

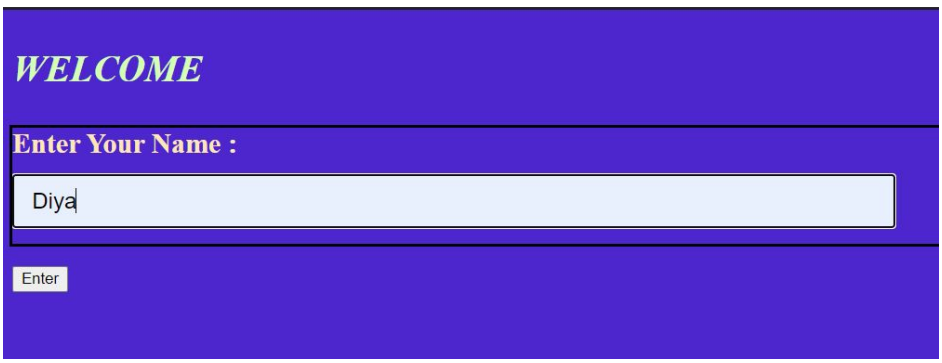
```
<!DOCTYPE html>
<html>
   <head>
      <title>GREETINGS</title>
      <meta charset="UTF-8">
      <meta name="viewport" content="width=device-width, initial-scale=1.0">
   </head>
   <style>
input[type= "text"]
{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}
body {
  background: #f3f3f3 ;
}
</style>
   <body style="background-color:#4d27cd;">
         <h1><i><p style= "color: #D6FFB6;">WELCOME</p></i></h1>
                  <form action="Greetings.jsp" method="get">
                           <h2 style= "outline-style: solid; outline-color: black;"><label style=
"color: #FFE8B6;" > Enter Your Name :</label>
                           <br><input type="text" name="uname" id="uname" ></h2>
                           <input type="submit" value="Enter" >
                           </h1>
                  </form>
         </body>
```

```
</html>
```

//Greetings.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"  import =
"java.io.*,java.util.Date , javax.servlet.*" %>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>GREETINGS</title>
   </head>
   <body>
      <%
         String nam= request.getParameter("uname");
      Date day = new Date();
     int hr= day.getHours();
      if (hr >= 0 && hr < 12) {
         out.print("Good Morning, " + nam);
      } else if (hr == 12) {
          out.print("Good Noon, " + nam);
      } else if (hr >= 12 && hr <= 16) {
          out.print("Good Afternoon, " + nam);
      } else {
          out.print("Good Evening, " + nam);
      }

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

//OUTPUT



5. Display the message You Win / You Lose based on the conditions specified.

Enter the string and select one of the even/odd buttons. If odd is selected, check the odd positions in the string and if it contains vowels, then display 'You Win' else display 'You Lose'. Similarly, for even button.

//StringVowel.html

```
<html>
<body>
<hr>
<center>
   <h2><i><u>WORKING WITH STRING</u></i></h2>
<form action="StringVowel.jsp" method = "POST">
<label for="str">ENTER THE STRING:</label>
<input type="text" id="str" name="str"><br><br>

</center>
<hr>
<fieldset>
<p>Please select a option to check the position of vowels :</p>
 <input type="radio" id="odd" name="vowel" value="odd">
 <label for="add">ODD</label><br>
 <input type="radio" id="even" name="vowel" value="even">
 <label for="sub">EVEN</label><br>

<br><br>
<input type="submit" value="Submit">
</fieldset>
</form>
</body>
</html>
```

//StringVowel.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8" %>
<%@page import="java.lang.Math" %>

<!DOCTYPE html>
<html>
<body>

<%! String[] selected;
String res="";
String chk= "aeiou";
int ilen, ileng, icounter =0;

double len; double leng;
double counter=0.0; %>
<% String s = request.getParameter("str");%>

<%
selected = request.getParameterValues("vowel");

for(int k=0; k<selected.length; k++)
{
   if (selected[k].equals("even"))
   {
      ilen= s.length();
      ileng=(ilen/2);
     for(int i=1; i<ilen; i=i+2)
     {
        if(i%2!=0)

                if(s.charAt(i) == 'a'|| s.charAt(i) == 'e'|| s.charAt(i) == 'i' || s.charAt(i) == 'o' ||
                s.charAt(i) == 'u' || s.charAt(i) == 'A'|| s.charAt(i) == 'E'|| s.charAt(i) == 'I' ||
                s.charAt(i) == 'O' || s.charAt(i) == 'U')

                        icounter++;
        }
```

```
    if(icounter==ileng)
     out.println("YOU WIN");
    else
       out.println("YOU LOSE");
    icounter=0;
     }

      if (selected[k].equals("odd"))
  {
     len= s.length();
     leng=Math.ceil(len/2);
    for(int i=0; i<len; i=i+2)
    {
       if(i%2==0)
                if(s.charAt(i) == 'a'|| s.charAt(i) == 'e'|| s.charAt(i) == 'i' || s.charAt(i) == 'o' ||
                s.charAt(i) == 'u' || s.charAt(i) == 'A'|| s.charAt(i) == 'E'|| s.charAt(i) == 'I' ||
                s.charAt(i) == 'O' || s.charAt(i) == 'U')
                        Counter++;
        }
    if(counter==leng)
     out.println("YOU WIN");
    else
       out.println("YOU LOSE");
    counter=0.0;
     }
}
%>

</body>
</html>
```

//OUTPUT


WORKING WITH STRING
ENTER THE STRING: malayalam
Please select a option to check the position of vowels :
ODD
EVEN
Submit


YOU LOSE


WORKING WITH STRING
ENTER THE STRING: malayalam
Please select a option to check the position of vowels :
ODD
EVEN
Submit


YOU WIN

6. Create your custom library which contains two tags: <hello>, <choco>.
Usage of the tags:
<hello name=”Ajay”>: Output should be Hello Ajay. It contains a mandatory attribute ‘name’ which can accept Dynamic value.
<choco texture=”Chewy”>: Output should be FiveStar, BarOne.
<choco texture=”Crunchy”>: Output should be Munch. KitKat.
That means the mandatory attribute must accept a value, and based on the attributes value, it should give output. You must use a bean ChocoBean for this purpose.

//HelloChoco package

//ChocoBean.java

```
package HelloChoco;

import java.io.IOException;
import java.io.Serializable;
import java.beans.*;
import javax.servlet.jsp.JspException;
import javax.servlet.jsp.JspWriter;
import javax.servlet.jsp.tagext.SimpleTagSupport;


public class ChocoBean extends SimpleTagSupport implements Serializable
{
   private String texture;

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

    public void doTag() throws JspException, IOException
    {
        JspWriter out = getJspContext().getOut();
        if(texture.equalsIgnoreCase("Chewy"))
```

```
        out.println("FIveStar, BarOne");
        else if(texture.equalsIgnoreCase("Crunchy"))
        out.println("Munch, KitKat");
    }
}
```

//Hello.java

```
package HelloChoco;

import java.io.IOException;
import javax.servlet.jsp.JspException;
import javax.servlet.jsp.JspWriter;
import javax.servlet.jsp.tagext.SimpleTagSupport;


public class Hello extends SimpleTagSupport
{
   private String name;

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

   @Override
    public void doTag() throws JspException, IOException
    {
        JspWriter out = getJspContext().getOut();
        out.println("Hello "+name);
    }
}
```

//HelloChoc.tld

```
<?xml version="1.0" encoding="UTF-8"?>
<taglib version="2.1" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-jsptaglibrary_2_1.xsd">
  <tlib-version>1.0</tlib-version>
  <jsp-version> 2.0</jsp-version>
  <short-name>HelloChoco</short-name>
  <uri>/WEB-INF/tlds/HelloChoc</uri>

  <tag>
     <name>HelloTag</name>
     <tag-class>HelloChoco.Hello</tag-class>
     <body-content>empty</body-content>

     <attribute>
        <name>name</name>
        <required>true</required>
        <type>String</type>
     </attribute>
  </tag>
<tag>
     <name>Chocolate</name>
     <tag-class>HelloChoco.ChocoBean</tag-class>
     <body-content>scriptless</body-content>
     <attribute>
        <name>texture</name>
        <required>true</required>
        <type>String</type>
     </attribute>
  </tag>
</taglib>
```

//HelloChocoTexture.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib prefix = "hc" uri = "/WEB-INF/tlds/HelloChoc.tld"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>TEXTURE</title>
      <style>
      section
      {
            border-radius: 1em;
             padding: 1em;
             position: absolute;
             top: 50%;
             left: 50%;
             margin-right: -50%;
             transform: translate(-50%, -50%)
      }
      </style>
   </head>
   <body style="background-color:#9997F0;" >
      <section>
   <fieldset>
   <center>
      <h2>

         <hc:HelloTag name= "Diya" />
         <br/>
         You like: <hc:Chocolate texture="Crunchy"/>

      </h2></center>
```

```
        </fieldset>
      </section>
   </body>
</html>
```

//OUTPUT

7. Create a custom tag “substring” with 3 mandatory attributes “input”, “start” and “end”.

//SubStr.java

```
package Strings;

import javax.servlet.jsp.tagext.*;
import javax.servlet.jsp.*;
import java.io.*;
import java.lang.*;
import java.util.*;

public class SubStr extends SimpleTagSupport
{
   private String input;
   private int start, end;

   public void setInput(String input) {
      this.input = input;
   }

   public void setStart(int start) {
      this.start = start;
   }

   public void setEnd(int end) {
      this.end = end;
   }

   StringWriter sw= new StringWriter();

   public void doTag() throws JspException, IOException {
         if (input != null) {
            JspWriter out = getJspContext().getOut();
            out.println("The string is: "+input+"<br>");
            StringBuffer sbf = new StringBuffer(input);
            String str=sbf.substring(start, end);
```

```
            out.println(" The substring is: "+str);
          }
           else {
       /* use message from the body */
                getJspBody().invoke(sw);
                getJspContext().getOut().println(sw.toString());
              }
          }
    }
```

//Ques7.tld

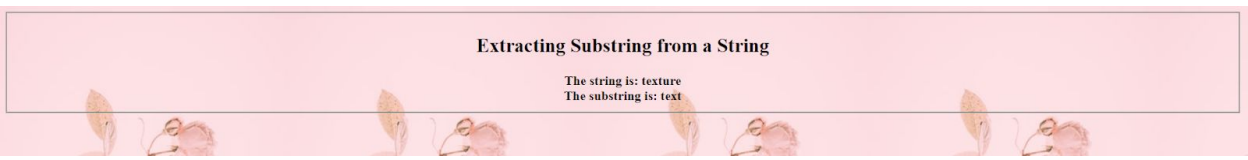
```
<?xml version="1.0" encoding="UTF-8"?>
<taglib version="2.1" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-jsptaglibrary_2_1.xsd">
  <tlib-version>1.0</tlib-version>
  <jsp-version> 2.0</jsp-version>
  <short-name>ques7</short-name>
  <uri>/WEB-INF/tlds/Ques7</uri>

  <tag>
     <name>Substring</name>
     <tag-class>Strings.SubStr</tag-class>
     <body-content>scriptless</body-content>

     <attribute>
        <name>input</name>
        <required>true</required>
        <type>String</type>
     </attribute>

    <attribute>
        <name>start</name>
          <required>true</required>
        <type>int</type>
     </attribute>
```

```
  <attribute>
       <name>end</name>
        <required>true</required>
       <type>int</type>
    </attribute>

 </tag>
</taglib>
```

//Ques7.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib prefix = "substr" uri = "/WEB-INF/tlds/Ques7.tld"%>
<!DOCTYPE html>
<html>
  <head>
     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
     <title>SUBSTRINGS</title>
  </head>
  <body background="pink.jpg">
  <fieldset>
  <center>
     <h2>Extracting Substring from a String</h2>

     <b><substr:Substring input= "texture" start="0" end="4" />  </b>
       </fieldset>

  </body>
</html>
```

//OUTPUT



8. Create a custom tag “reverse” with a mandatory attribute “input” to reverse a string.

//Reverse.java

```
package Strings;

import javax.servlet.jsp.tagext.*;
import javax.servlet.jsp.*;
import java.io.*;
import java.lang.*;
import java.util.*;

public class Reverse extends SimpleTagSupport
{
   private String input;

   public void setInput(String input) {
      this.input = input;
   }

   StringWriter sw= new StringWriter();

   @Override
   public void doTag() throws JspException, IOException {
         if (input != null) {
            JspWriter out = getJspContext().getOut();
            out.println("The string is: "+input+"<br>");
            StringBuffer sbf = new StringBuffer(input);
            sbf=sbf.reverse();

            out.println(" The reversed string is: "+sbf);
         }
          else {
      /* use message from the body */
               getJspBody().invoke(sw);
               getJspContext().getOut().println(sw.toString());
            }
```

```
        }
    }
```

//Reverse.tld

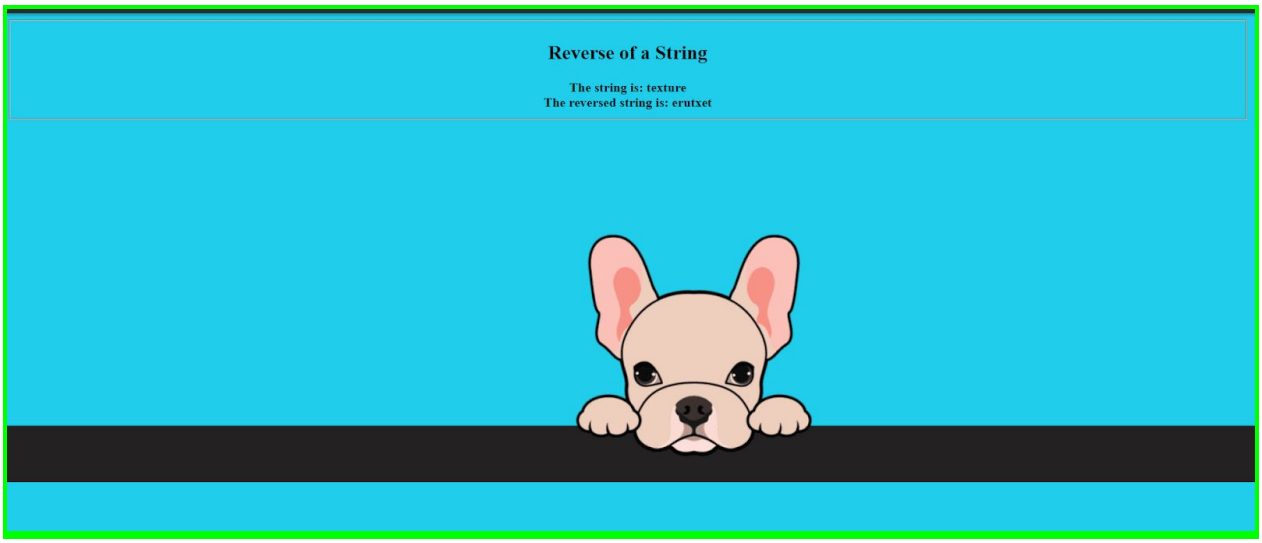
```
<?xml version="1.0" encoding="UTF-8"?>
<taglib version="2.1" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-jsptaglibrary_2_1.xsd">
  <tlib-version>1.0</tlib-version>
  <jsp-version> 2.0</jsp-version>
  <short-name>Reverse</short-name>
  <uri>/WEB-INF/tlds/Reverse</uri>

  <tag>
     <name>Reverse</name>
     <tag-class>Strings.Reverse</tag-class>
     <body-content>scriptless</body-content>

     <attribute>
        <name>input</name>
        <required>true</required>
        <type>String</type>
     </attribute>

  </tag>
</taglib>
```

//Reverse.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib prefix = "rev" uri = "/WEB-INF/tlds/Reverse.tld"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>REVERSE</title>
   </head>
```

```
   <body background="dog.png">
   <fieldset>
   <center>
      <h2>Reverse of a String</h2>

      <b><rev:Reverse input= "texture" />   </b>
         </fieldset>

   </body>
</html>
```

//OUTPUT



9. Create a custom tag “today” that displays today’s date and time.

//Dated.java

```
package DateTime;

import javax.servlet.jsp.tagext.*;
import javax.servlet.jsp.*;
import java.io.*;

public class dated extends SimpleTagSupport
{

   @Override
   public void doTag() throws JspException, IOException
   {
      JspWriter out = getJspContext().getOut();
      out.println("Today's Date and Time: "+new java.util.Date());
   }

}
```

//Date.tld

```
<?xml version="1.0" encoding="UTF-8"?>
<taglib version="2.1" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-jsptaglibrary_2_1.xsd">
  <tlib-version>1.0</tlib-version>
  <jsp-version> 2.0</jsp-version>
  <short-name>Date</short-name>
  <uri>/WEB-INF/tlds/Date</uri>

  <tag>
     <name>DateT</name>
     <tag-class>DateTime.dated</tag-class>
```

```
    <body-content>empty</body-content>
   <attribute></attribute>
 </tag>
</taglib>
```

//Date.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib prefix = "dat" uri = "/WEB-INF/tlds/Date.tld"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>DATE - TIME</title>
    <style>
    section
    {
          border-radius: 1em;
           padding: 1em;
           position: absolute;
           top: 50%;
           left: 50%;
           margin-right: -50%;
           transform: translate(-50%, -50%)
    }
    </style>
  </head>
  <body style="background-color:#9997F0;" >
    <section>
  <fieldset>
  <center>
    <h2>

       <dat:DateT /></h2></center>
       </fieldset>
    </section>
  </body>
</html>
```

//OUPUT

10. Ask a user's name and age on a HTML form.

Then display Hello <uname> on a JSP. On that same JSP page ask the product the user would like to buy. Then redirect to another JSP which would display: Hello <uname>, You have ordered <product>.

//User.java

```
package stud;

import java.io.*;

public class User implements Serializable
{
   private String uname;
   private int age;
   private String prod;


   public User()
   {
      uname="";
      age=0;
   }

   public String getUname() {
      return uname;
   }

   public void setUname(String uname) {
      this.uname = uname;
   }

   public int getAge() {
      return age;
   }

   public void setAge(int age) {
      this.age = age;
```

```
    }

    public String getProd() {
        return prod;
    }

    public void setProd(String prod) {
        this.prod = prod;
    }

}
```

//q2.html

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
input[type= "text"]{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}

</style>
    </head>
    <body>
    <center>
        <h1>PERSONAL DETAILS</h1>
        <form action="q2.jsp" method="post">
            <h2><b>
            Username: <input type ="text" name="uname" placeholder="Enter your name"
style:color="F9E6B0"><br>
             Age: <input type ="text" name="age" placeholder="Enter your age">
             <br>
```

```
            <input type ="submit" value="Enter">
              </b></h2>
            </form>
    </center>
    </body>
</html>
```

//q2.jsp

```
<%--
    Document   : q2
    Created on : Oct 8, 2020, 8:57:49 PM
    Author     : aaa
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
input[type= "text"]{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}

</style>
        <title>Q2</title>
    </head>
    <body>
    <center> <h2><b>
        <jsp:useBean id="u" scope="session" class="stud.User"/>
        <jsp:setProperty name="u" property="*" />
```

```
        Welcome
        <jsp:getProperty name="u" property= "uname" /><br>
        <form action="q2_1.jsp" method="post">
           Product:<input type ="text" name="prod"><br>
           <input type ="submit" value="Enter"><br>
            </form>
           </b></h2></center>
    </body>
</html>
```

//q2_1.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
            <style>
input[type= "text"]
{
  width: 50%;
  padding: 10px 15px;
  margin: 13px 0;
  box-sizing: border-box;
  border: 2px solid black;
  border-radius: none;
  font-size: 80%;
}

</style>
        <title>JSP Page</title>
    </head>
    <body>
    <center><b><h2>
        <jsp:useBean id="u" scope="session" class="stud.User"/>
        <jsp:setProperty name="u" property="*" />
        Hello <jsp:getProperty name="u" property= "uname" /><br>
```

```
      You have ordered:
      <jsp:getProperty name="u" property= "prod" /><br>
         </h2></b></center>
   </body>
</html>
```

//OUTPUT