Programme Name: B.SC. (H) COMPUTER SCIENCE
SEMESTER-V
Paper Title: INTERNET TECHNOLOGIES PRACTICAL

Submitted by:

Preeti Trikha
Examination Roll No: 18044570031
College RollNo: CSC/18/49

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

B.Sc. (H) Computer Science 5th Semester

Java

1. Implement a Bank Account having Instance variables: Account Number, Balance and having methods:

        float Deposit (float x)
        float withdraw (float x)
        int get account no ()
        float get balance ()
        tax deduction ()

    Then implement class Bank having an array list of accounts of type BankAccount. Implement following methods:

      AddAccount in Bank
      Get Total balance in Bank
      Get account number with max. and min. balance
      Find an account given a bank account no.
      Count no. of accounts having atleast specific balance

```
package demo1;

import java.util.ArrayList;

import java.util.Scanner;


class BankAccount{

        private

        float balance;

        int accNo;


        public

        BankAccount(){

          balance=0;

        accNo=0;

        }

        BankAccount(float b){

        balance=b;
```

```
accNo=accNo+1;

}

BankAccount(int a){

accNo=a;

}

BankAccount(float b,int acc){

balance=b;

accNo=acc;

}


float deposit(float x){

float newbal=balance+x;

balance=newbal;

return balance;

}

float withdrawl(float x){

float newbal=balance-x;

balance=newbal;

return balance;

}

int getAccNo(){

return accNo;

}

float getBalance(){

return balance;

}

 int taxDeduction(int d){

int x=(int)(d-(.02*d));

return x;
```

```
        }
};

class Bank {

        public
        ArrayList<BankAccount> accounts=new ArrayList<BankAccount> ();

        public void addAccount(BankAccount a) {   //add account in bank
        accounts.add(a);
        System.out.println("YOUR ACCOUNT HAS BEEN CREATED");
        }

        public double getTotalBalance(){    //get total balance in bank
        double total=0;
        for(BankAccount a: accounts){
        total=total+a.getBalance();
        }

        return total;
        }

        public BankAccount getMax(){   //get account number with maximum balance
        if(accounts.size()==0)
        return null;
        BankAccount largestYet=accounts.get(0);
        for(int i=1;i<accounts.size();i++){
        BankAccount a=accounts.get(i);
       if(a.getBalance()>largestYet.getBalance()){
```

```
 largestYet=a;
 }
 }
 return largestYet;
 }

 public BankAccount getMin(){       //get account number with minimum balance
 if(accounts.size()==0)
 return null;
 BankAccount smallestYet=accounts.get(0);
 for(int i=1;i<accounts.size();i++){
 BankAccount a=accounts.get(i);
if(a.getBalance()<smallestYet.getBalance()){
 smallestYet=a;
 }
 }
 return smallestYet;
 }

 public BankAccount find(int accountNumber){ //find an account given a bank account number
 for(BankAccount a: accounts){
if(a.getAccNo()==accountNumber){  //found a match
 return a;
 }
 }
 return null;
 }

 public int count(double atleast){   //count numbr of accounts having atleast specific balance
```

```
int matches=0;

for(BankAccount a:accounts){

if(a.getBalance()>=atleast)//found match

matches++;

}

return matches;

}


public void menu(){

Scanner a = new Scanner(System.in);

int ch,x;

do{

System.out.println("BANK MENU");

System.out.println("1. Create new account.");

System.out.println("2. Get total balance in bank.");

System.out.println("3. Get account number with maximum balance.");

System.out.println("4. Get account number with minimum balance.");

System.out.println("5. Find an account of given account number.");

System.out.println("6. Count number of accounts having atleast specific balance.");

System.out.println("ENTER YOUR CHOICE:");

x=a.nextInt();

switch(x){

case 1:

          BankAccount acc=new BankAccount();

         this.addAccount(acc);

          break;

case 2:

          double bal=getTotalBalance();

   System.out.println("Total balance of bank is "+bal);
```

```
          break;
case 3:
          BankAccount a1=getMax();
          int m=a1.getAccNo();
          float n=a1.getBalance();
    System.out.println("Account number "+m+" has maximum balance of Rs. "+n);
          break;
case 4:
          BankAccount a2=getMin();
          int y=a2.getAccNo();
          float z=a2.getBalance();
    System.out.println("Account number "+y+" has minimum balance of Rs. "+z);
          break;
case 5:
    System.out.println("Enter the account number you want to search:");
          int AccNo=a.nextInt();
          BankAccount a3=find(AccNo);
          if(a3==null)
       System.out.println("This account does not exist.");
          else
          {
          BankAccount a4=getMax();
          int p=a4.getAccNo();
          float q=a4.getBalance();
       System.out.println("Account number "+p+" has balance of Rs. "+q);
          }
          break;
case 6:
    System.out.println("Enter the minimum amount you want to search:");
```

```
                double amount=a.nextInt();
                int c=count(amount);
            System.out.println("Number of accounts with minimum balance "+amount+" are "+c);
                break;
        default:
            System.out.println("INVALID CHOICE");
                break;
        }
        System.out.println("Press 1 to continue and 0 to exit.");
        ch=a.nextInt();
        if(ch==0)
        System.exit(0);
        }while(ch==1);
        }
        };



public class Demo1 {
        public static void main(String[] args) {
        int ch,ch1;
        BankAccount b1=new BankAccount(2500);
        Bank b=new Bank();
        Scanner sc= new Scanner(System.in);
        System.out.println("1.Existing account");
        System.out.println("2.Bank ");
        ch1=sc.nextInt();

        if(ch1==1)
```

```
{
int r;
do{
System.out.println("Enter your account number:");
int m=sc.nextInt();
BankAccount bacc=new BankAccount(m);
         System.out.println("MENU");
 System.out.println("1.Deposit in Account");
 System.out.println("2.Withdraw from bank");
System.out.println("3. Get account number");
System.out.println("4.Get Balance");
System.out.println("5. Deduce tax");
         System.out.println("6.Go to bank menu");
         System.out.println("Enter Your Choice:");
         ch = sc.nextInt();
switch(ch){
         case 1:
      System.out.println("Enter the amount you want to deposit:");
         float x=sc.nextInt();
         float bal=bacc.deposit(x);
         System.out.println(bal);
         break;
         case 2:
      System.out.println("Enter the amount you want to withdraw:");
         float y=sc.nextInt();
         float bal1=bacc.withdrawl(y);
      System.out.println(bal1);
         break;
         case 3:
```

```
		int ano=bacc.getAccNo();
	System.out.println("Your account number is "+ano);
		break;
		case 4:
		float bal2=bacc.getBalance();
	System.out.println("Your balance is "+bal2);
		break;
		case 5:
		int tax=bacc.taxDeduction(5);
	System.out.println("Deduced tax is "+tax);
		break;
		case 6:
		b.menu();
		break;
		default:
	System.out.println("INVALID CHOICE");
		break;
}
System.out.println("Press 1 to continue and 0 to exit:");
r=sc.nextInt();
if(r==0)
		System.exit(0);
}while(r==1);
}
else
{
b.menu();
}
}
```

```
}
```

2. Implement an Abstract Class Stack with methods push, pop, display for two classes: StaticStack and DyanamicStack. StaticStack uses one dimensional integer array to store numbers and DyanamicStack uses an integer ArrayList to store.

```
package demo2;
import java.io.*;
import java.util.*;
import java.io.PrintStream;
import java.util.Scanner;



abstract class stack
{
            public int size;
          public int top;
          public abstract void push(int e);
          public abstract int pop();
          public abstract void display();

          public stack()
          {
          top=-1;
          }
};

 class staticstack extends stack
```

```
{
        public int arr[];
        public staticstack()
        {
        super();
        }
        public staticstack(int size)
        {
        super();
        arr=new int[size];
        }
        public void push(int e)
        {
    if((super.top)==arr.length-1)
        {
      System.out.println("Stack Overflow");
        }
        else
        {
        super.top++;
        arr[super.top]=e;
        }
        }
        public int pop()
        {
        if((super.top)==-1)
        {
```

```
        return 0;
        }
        else
        {
        int x=arr[super.top];
        super.top--;
        return x;
        }
        }
        public void display()
        {
      System.out.println("Elements in stack are:");
        for(int i=0;i<arr.length;i++)
        {
        System.out.print(arr[i]+" ");
        }
        System.out.println();
        }
};

class dynamicstack extends stack
{
        ArrayList<Integer> arr;
        public dynamicstack()
        {
        this.arr = new ArrayList();
        }
```

```
     public void push(int e)

     {

     this.arr.add(++this.top, Integer.valueOf(e));

     }

     public int pop()

     {

     if((this.top)==-1)

     {

     return -1;

     }

     else

     {

     return ((Integer)this.arr.remove(this.top--)).intValue();

     }

     }

     public void display()

     {

    System.out.print("Elements in stack are: ");

            for (int i = 0; i <this.arr.size(); i++)

            {

                         System.out.print(this.arr.get(i) + " ");

            }

            System.out.println();

     }

};
```

```
public class Demo2
{
        public static void main(String[] args)
        {
        Scanner a = new Scanner(System.in);
    System.out.println("Enter");
    System.out.println("1.StaticStack");
        System.out.println("2.DynamicStack");
        System.out.println("3.Exit");
        int m = a.nextInt();
        int p,ch;
        switch(m)
        {
        case 1: System.out.println("Enter array size");
                int s = a.nextInt();
                staticstack ob=new staticstack(s);
                do
                {
             System.out.println("Enter the operation you want to perform");
             System.out.println("1.Push");
             System.out.println("2.Pop");
             System.out.println("3.Display");
             System.out.println("Enter your choice");
             p=a.nextInt();
                switch(p)
                {
                case 1:
```

```
                int j=0;
                      System.out.println("Enter a number to be pushed into the stack");
                j=a.nextInt();
                ob.push(j);
                break;


              case 2:
                int v =ob.pop();
                if(v==-1)
                System.out.println("Stack Underflow");
                else
                System.out.println("Element popped =" + v);
                break;
              case 3:
                ob.display();
                break;
              }
           System.out.println("Press 1 to Continue with Static Stack...");
              ch = a.nextInt();
       }while(ch==1);
case 2:
dynamicstack ob1=new dynamicstack();
do
{
    System.out.println("DYNAMIC STACK");
    System.out.println("Enter the operation you want to perform");
    System.out.println("1.Push");
```

```
     System.out.println("2.Pop");
              System.out.println("3.Display");
     System.out.println("Enter your choice");
             p=a.nextInt();
             switch(p)
             {
             case 1:
             int j=0;
            System.out.println("Enter a number to be pushed into the stack");
             j = a.nextInt();
            ob1.push(j);
             break;
             case 2:
                  int v =ob1.pop();
            if(v==-1)
                System.out.println("Stack Underflow");
             else
                System.out.println("Element popped =" + v);
             break;
             case 3:
            ob1.display();
             break;
             }
     System.out.println("Press 0 to Continue with dynamic Stack..");
              ch = a.nextInt();
}while(ch==0);
}
```

```
	}
};
```

JavaBeans

1. Implement Student JavaBean using Serializability Interface.

First.jsp

```
<%--
	Document   : first
	Created on : Oct 19, 2020, 11:53:21 AM
	Author   : hp
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<jsp:useBean id="stud"  scope="page" class="stu.student"/>
<jsp:setProperty name="stud" property="name" value="riya"/>
	Name of person is:<jsp:getProperty name="stud" property="name"/><br>
<jsp:setProperty name="stud" property="gender" value="f"/>
	Gender of person is:<jsp:getProperty name="stud" property="gender"/><br>
<jsp:setProperty name="stud" property="email" value="riaa.8053@gmail.com"/>
	Email of person is:<jsp:getProperty name="stud" property="email"/>
</body>
```

```
</html>
```

Student.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package stu;
import java.io.Serializable;
/**
 *
 * @author hp
 */
public class student implements java.io.Serializable{

        private static String DATE_FORMAT_PATTERN="yyyy-MM-dd";
        private static String[]GENDER_LIST={"m","f"};

        public student() {
        }




        private String dob;

        /**
```

```
    * Get the value of dob
    *
    * @return the value of dob
    */
    public String getDob() {
    return dob;
    }

    /**
    * Set the value of dob
    *
    * @param dob new value of dob
    */
    public void setDob(String dob) {
this.dob = dob;
    }

    private String email;

    /**
    * Get the value of email
    *
    * @return the value of email
    */
    public String getEmail() {
    return email;
    }

    /**
```

```
        * Set the value of email
        *
        * @param email new value of email
        */
        public void setEmail(String email) {
this.email = email;
        }

        private double[] marks;

        /**
        * Get the value of marks
        *
        * @return the value of marks
        */
        public double[] getMarks() {
        return marks;
        }

        /**
        * Set the value of marks
        *
        * @param marks new value of marks
        */
        public void setMarks(double[] marks) {
this.marks = marks;
        }

        /**
```

```
            * Get the value of marks at specified index

            *

            * @param index the index of marks

            * @return the value of marks at specified index

            */

            public double getMarks(int index) {

            return this.marks[index];

            }


            /**

            * Set the value of marks at specified index.

            *

            * @param index the index of marks

            * @param marks new value of marks at specified index

            */

            public void setMarks(int index, double marks) {

this.marks[index] = marks;

            }


            private String gender;


            /**

            * Get the value of gender

            *

            * @return the value of gender

            */

            public String getGender() {

            return gender;

            }
```

```
        /**
        * Set the value of gender
        *
        * @param gender new value of gender
        */
        public void setGender(String gender) {
this.gender = gender;
        }

        private String address;

        /**
        * Get the value of address
        *
        * @return the value of address
        */
        public String getAddress() {
        return address;
        }

        /**
        * Set the value of address
        *
        * @param address new value of address
        */
        public void setAddress(String address) {
this.address = address;
        }
```

```
    */

    private String course;

    /**
    * Get the value of course
    *
    * @return the value of course
    */
    public String getCourse() {
    return course;
    }

    /**
    * Set the value of course
    *
    * @param course new value of course
    */
    public void setCourse(String course) {
this.course = course;
    }

    private String name;

    /**
    * Get the value of name
    *
    * @return the value of name
    */
    public String getName() {
```

```
return name;
}

/**
* Set the value of name
*
* @param name new value of name
*/
public void setName(String name) {
this.name = name;
}

private int roll_no;

/**
* Get the value of roll_no
*
* @return the value of roll_no
*/
public int getRoll_no() {
return roll_no;
}

/**
* Set the value of roll_no
*
* @param roll_no new value of roll_no
*/
public void setRoll_no(int roll_no) {
```

```
this.roll_no = roll_no;
            }


            public booleanisnameValid(){
booleanisValid=false;
            if(name!=null)
isValid=true;
            return isValid;
            }



            public booleanisgenderValid(){
booleanisValid=false;
if(gender=="m" || gender=="f")
isValid=true;
            return isValid;
            }
            public booleanisemailValid(){
booleanisValid=false;
            if(email!=null)
isValid=true;
            return isValid;
            }
}
```

2. Implement Employee JavaBean using Serializability Interface.

Index.html

```
<!DOCTYPE html>
```

```
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<html>
<head>
<title>employee bean</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<form action="second.jsp" method="post">
        Employee name:<input type="text" name="ename"><br>
        Employee mail:<input type="text" name="mail"><br>
        Employee gender:<input type="text" name="gender"><br>
<input type="submit"/>
</form>
</html>
```

Second.jsp

```
<%--
        Document   : second
        Created on : Oct 20, 2020, 11:45:02 AM
        Author   : hp
--%>

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

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<jsp:useBean id="empinfo" class="employeebean.employeebean"/>
<jsp:setProperty property="*" name="empinfo"/>
        You have entered below details:<br>
        Name of employee is:<jsp:getProperty property="ename" name="empinfo"/><br>
        Email id of employee is:<jsp:getProperty property="mail" name="empinfo"/><br>
        Gender of employee is:<jsp:getProperty property="gender" name="empinfo"/><br>

</html>
```

Employeebean.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package employeebean;

/**
 *
 * @author hp
 */
public class employeebean {

        private static String DATE_FORMAT_PATTERN="yyyy-MM-dd";
```

```
		private static String[]GENDER_LIST={"m","f"};



		public employeebean() {
		}

		private String dob;

		/**
		* Get the value of dob
		*
		* @return the value of dob
		*/
		public String getDob() {
		return dob;
		}

		/**
		* Set the value of dob
		*
		* @param dob new value of dob
		*/
		public void setDob(String dob) {
this.dob = dob;
		}

		private String email;

		/**
```

```
        * Get the value of email
        *
        * @return the value of email
        */
        public String getEmail() {
        return email;
        }

        /**
        * Set the value of email
        *
        * @param email new value of email
        */
        public void setEmail(String email) {
this.email = email;
        }

        private String gender;

        /**
        * Get the value of gender
        *
        * @return the value of gender
        */
        public String getGender() {
        return gender;
        }

        /**
        * Set the value of gender
        *
        * @param gender new value of gender
        */
        public void setGender(String gender) {
this.gender = gender;
        }

        private double salary;

        /**
        * Get the value of salary
        *
        * @return the value of salary
        */
        public double getSalary() {
        return salary;
        }

        /**
        * Set the value of salary
        *
        * @param salary new value of salary
        */
        public void setSalary(double salary) {
this.salary = salary;
        }

        private String address;
```

```
    /**
    * Get the value of address
    *
    * @return the value of address
    */
    public String getAddress() {
    return address;
    }

    /**
    * Set the value of address
    *
    * @param address new value of address
    */
    public void setAddress(String address) {
this.address = address;
    }

    private String dept;

    /**
    * Get the value of dept
    *
    * @return the value of dept
    */
    public String getDept() {
    return dept;
    }
```

```
        /**
        * Set the value of dept
        *
        * @param dept new value of dept
        */
        public void setDept(String dept) {
this.dept = dept;
        }

        private int empid;

        /**
        * Get the value of empid
        *
        * @return the value of empid
        */
        public int getEmpid() {
        return empid;
        }

        /**
        * Set the value of empid
        *
        * @param empid new value of empid
        */
        public void setEmpid(int empid) {
this.empid = empid;
        }
```

```
            */



            private String ename;


            /**
            * Get the value of ename
            *
            * @return the value of ename
            */
            public String getEname() {
            return ename;
            }


            /**
            * Set the value of ename
            *
            * @param ename new value of ename
            */
            public void setEname(String ename) {
this.ename = ename;
            }




            /**
            * Get the value of Salary
            *
            * @return the value of Salary
            */
```

```
        private double sal;

        /**
        * Get the value of sal
        *
        * @return the value of sal
        */
        public double getSal() {
        return sal;
        }

        /**
        * Set the value of sal
        *
        * @param sal new value of sal
        */
        public void setSal(double sal) {
this.sal = sal;
        }

        /**
        * Set the value of Salary
        *
        * @param Salary new value of Salary
        */

        private String mail;
        /**
        * Get the value of mail
        *
        * @return the value of mail
        */
        public String getMail() {
        return mail;
  }

        /**
        * Set the value of mail
        *
        * @param mail new value of mail
        */
        public void setMail(String mail) {
this.mail = mail;
        }

        private String[] property;

        /**
        * Get the value of property
        *
        * @return the value of property
        */
        public String[] getProperty() {
        return property;
        }
```

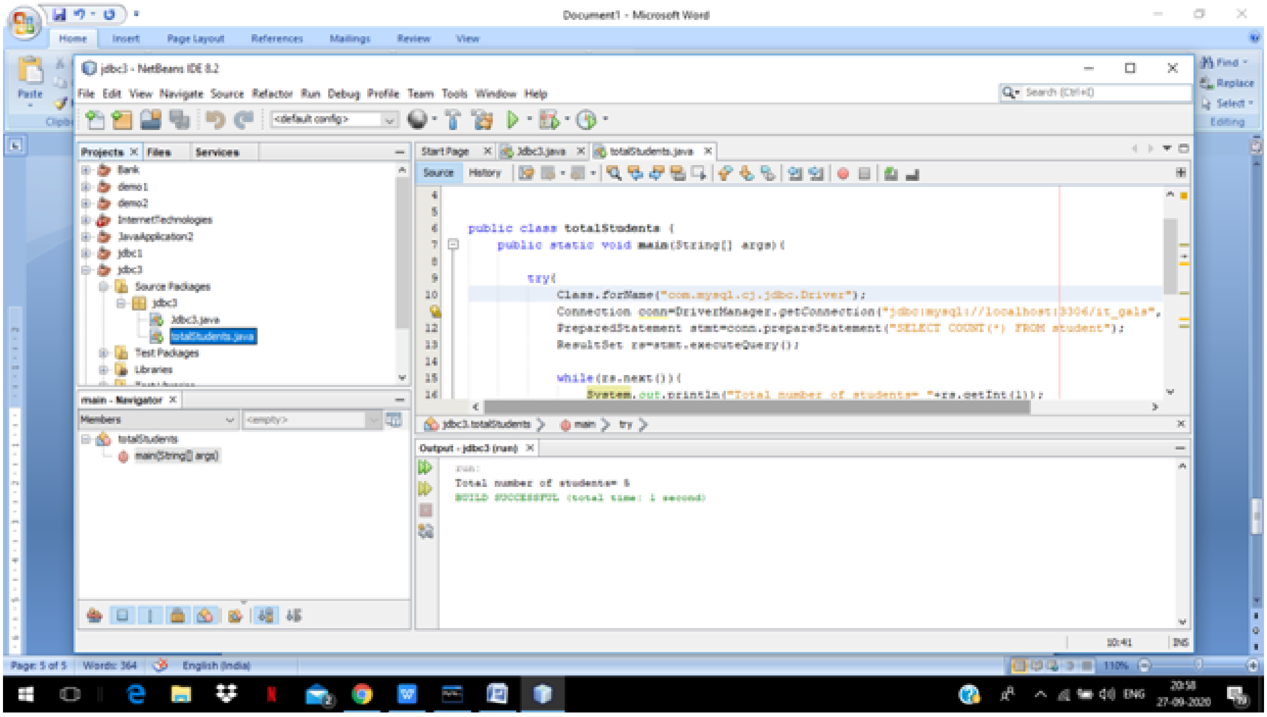
```
    /**
    * Set the value of property
    *
    * @param property new value of property
    */
    public void setProperty(String[] property) {
this.property = property;
    }

    /**
    * Get the value of property at specified index
    *
    * @param index the index of property
    * @return the value of property at specified index
    */
    public String getProperty(int index) {
    return this.property[index];
    }

    /**
    * Set the value of property at specified index.
    *
    * @param index the index of property
    * @param property new value of property at specified index
    */
    public void setProperty(int index, String property) {
this.property[index] = property;
    }
```

```
 public booleanisnameValid(){
booleanisValid=false;
            if(ename!=null)
isValid=true;
            return isValid;
            }



            public booleanisgenderValid(){
booleanisValid=false;
if(gender=="m" || gender=="f")
isValid=true;
            return isValid;
            }
            public booleanisemailValid(){
booleanisValid=false;
            if(mail!=null)
isValid=true;
            return isValid;
            }


}
```

JDBC

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

   a. Find total number of students

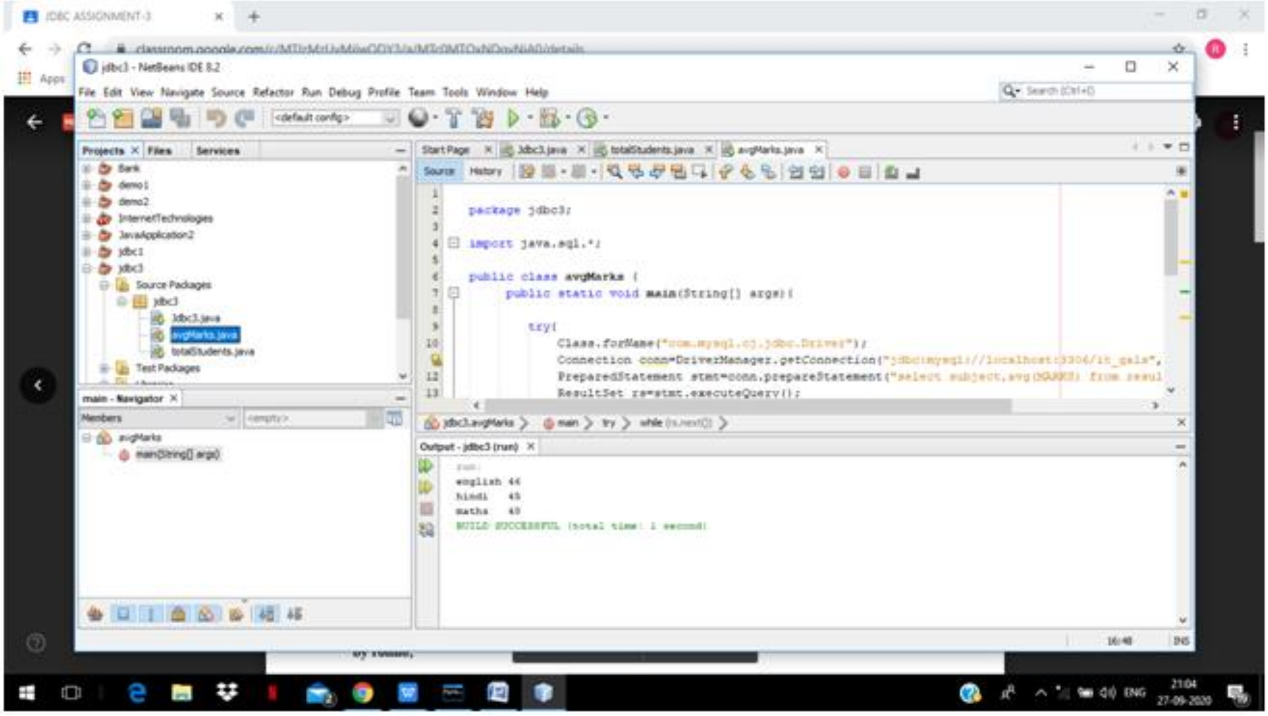


```
public class totalStudents {
    public static void main(String[] args){

        try{
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/it_gals",
            PreparedStatement stmt=conn.prepareStatement("SELECT COUNT(*) FROM student");
            ResultSet rs=stmt.executeQuery();

            while(rs.next()){
                System.out.println("Total number of students= "+rs.getInt(1));
```

```
run:
Total number of students= 5
BUILD SUCCESSFUL (total time: 1 second)
```

```
package jdbc3;

import java.sql.*;




public class totalStudents {

    public static void main(String[] args){



    try{

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

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

            PreparedStatement stmt=conn.prepareStatement("SELECT COUNT(*) FROM student");

            ResultSet rs=stmt.executeQuery();



            while(rs.next()){
            System.out.println("Total number of students= "+rs.getInt(1));

            }

            conn.close();

    }

    catch(Exception e){

     System.out.println(e);

    }

    }

}
```

b. Print average marks for each subject input by user.



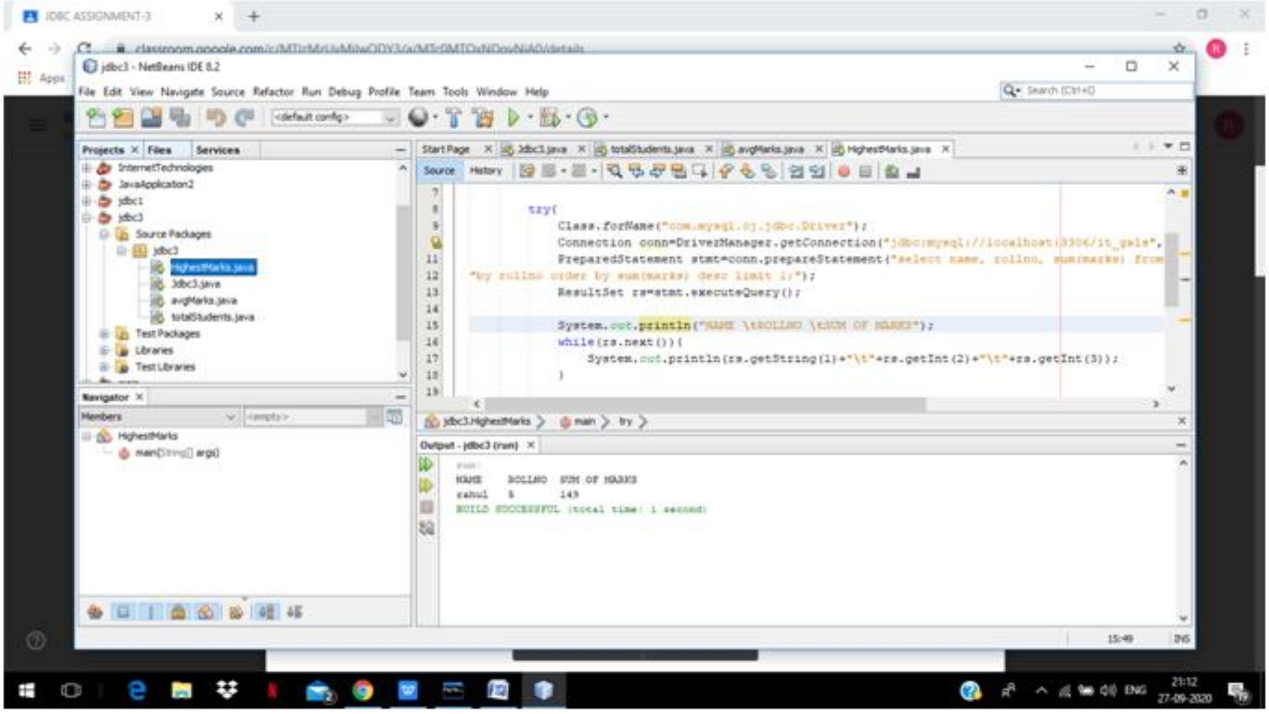
```
package jdbc3;

import java.sql.*;

public class avgMarks {
    public static void main(String[] args){

        try{
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/it_gals",
            PreparedStatement stmt=conn.prepareStatement("select subject,avg(MARKS) from resul
            ResultSet rs=stmt.executeQuery();
```

```
run:
english 44
hindi   45
maths   40
BUILD SUCCESSFUL (total time: 1 second)
```

```
package jdbc3;


import java.sql.*;


public class avgMarks {

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

```
  try{

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

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

PreparedStatement stmt=conn.prepareStatement("select subject,avg(MARKS) from results group by subject;");

ResultSet rs=stmt.executeQuery();



while(rs.next()){

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

  }



conn.close();

  }

  catch(Exception e){

System.out.println(e);

  }

  }

}
```

c. Find the name of student getting highest marks.



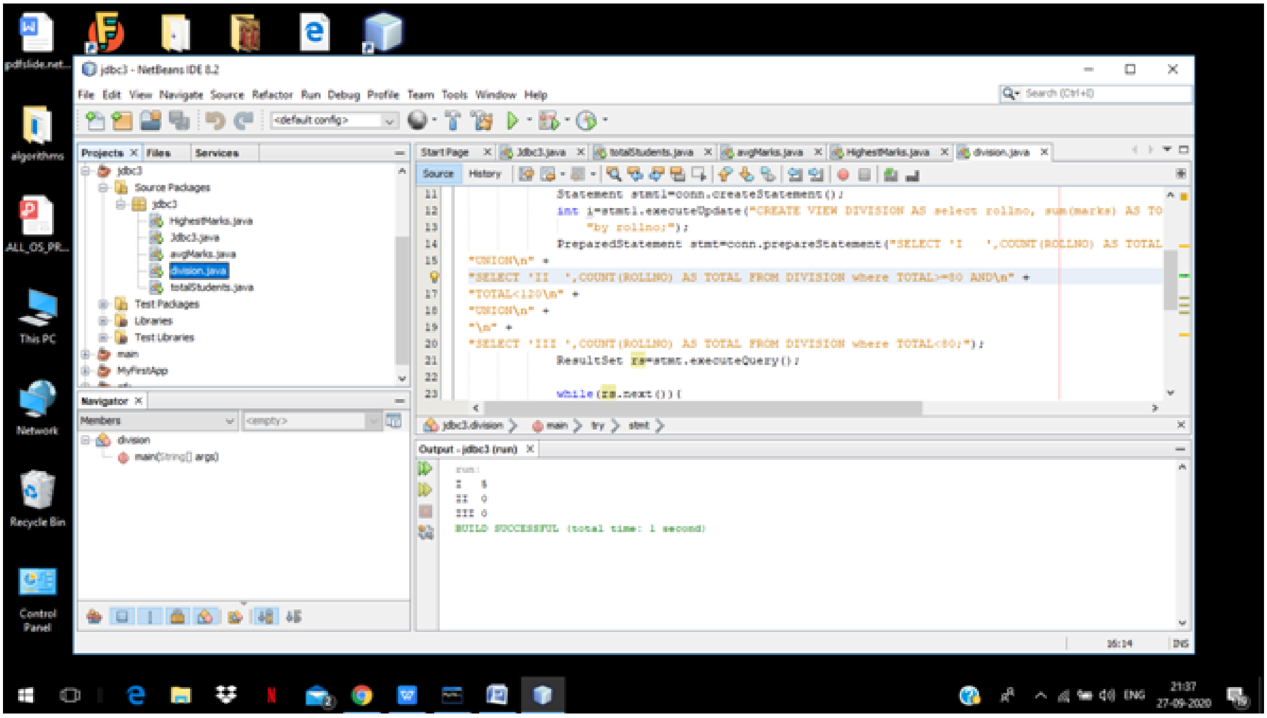
```
            try{
                Class.forName("com.mysql.cj.jdbc.Driver");
                Connection conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/it_gals",
                PreparedStatement stmt=conn.prepareStatement("select name, rollno, sum(marks) from
"by rollno order by sum(marks) desc limit 1;");
                ResultSet rs=stmt.executeQuery();

                System.out.println("NAME \tROLLNO \tSUM OF MARKS");
                while(rs.next()){
                    System.out.println(rs.getString(1)+"\t"+rs.getInt(2)+"\t"+rs.getInt(3));
                }
```

```
run:
NAME    ROLLNO  SUM OF MARKS
rahul   5       149
BUILD SUCCESSFUL (total time: 1 second)
```

```
package jdbc3;

import java.sql.*;



public class HighestMarks {

        public static void main(String[] args){



        try{

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

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

      PreparedStatement stmt=conn.prepareStatement("select name, rollno, sum(marks) from student, results where roll=rollno group\n" +

"by rollno order by sum(marks) desc limit 1;");

      ResultSet rs=stmt.executeQuery();



      System.out.println("NAME \tROLLNO \tSUM OF MARKS");

      while(rs.next()){

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

        }
```

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

d. Find no of students getting first, second and third division.

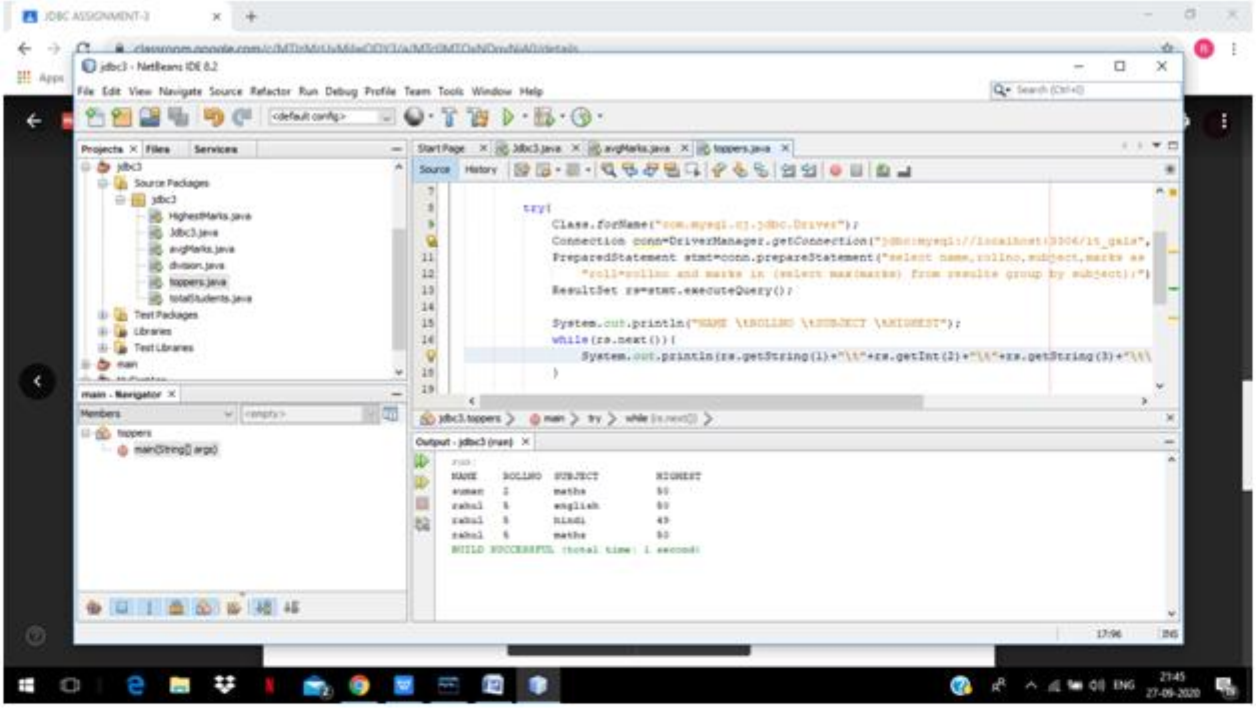


```
                    Statement stmt1=conn.createStatement();
                    int i=stmt1.executeUpdate("CREATE VIEW DIVISION AS select rollno, sum(marks) AS TO
                        "by rollno;");
                    PreparedStatement stmt=conn.prepareStatement("SELECT 'I   ',COUNT(ROLLNO) AS TOTAL
"UNION\n" +
"SELECT 'II  ',COUNT(ROLLNO) AS TOTAL FROM DIVISION where TOTAL>=80 AND\n" +
"TOTAL<120\n" +
"UNION\n" +
"\n" +
"SELECT 'III ',COUNT(ROLLNO) AS TOTAL FROM DIVISION where TOTAL<80;");
                    ResultSet rs=stmt.executeQuery();

                    while(rs.next()){
```

```
run:
I   5
II  0
III 0
BUILD SUCCESSFUL (total time: 1 second)
```

```
package jdbc3;
import java.sql.*;


public class division {
            public static void main(String[] args){


            try{
          Class.forName("com.mysql.cj.jdbc.Driver");
          Connection conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/it_gals","root","root");

          Statement stmt1=conn.createStatement();

            int i=stmt1.executeUpdate("CREATE VIEW DIVISION AS select rollno, sum(marks) AS TOTAL from
results group\n" +

             "by rollno;");

          PreparedStatement stmt=conn.prepareStatement("SELECT 'I   ',COUNT(ROLLNO) AS TOTAL FROM
DIVISION where TOTAL>=120\n" +

"UNION\n" +

"SELECT 'II  ',COUNT(ROLLNO) AS TOTAL FROM DIVISION where TOTAL>=80 AND\n" +

"TOTAL<120\n" +

"UNION\n" +

"\n" +

"SELECT 'III ',COUNT(ROLLNO) AS TOTAL FROM DIVISION where TOTAL<80;");

          ResultSet rs=stmt.executeQuery();


          while(rs.next()){

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

          }


          conn.close();

            }

       catch(Exception e){

          System.out.println(e);

            }

            }

}
```

e. Find subject wise toppers

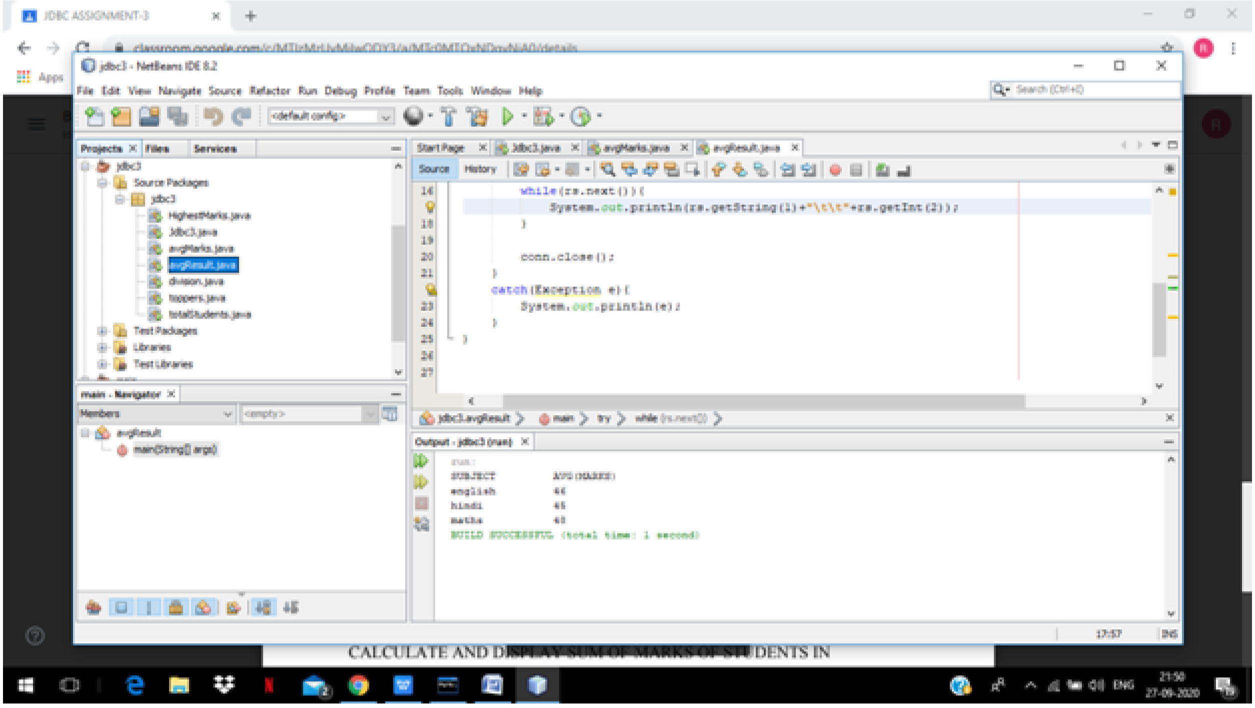


```
try{
    Class.forName("com.mysql.cj.jdbc.Driver");
    Connection conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/it_gals",
    PreparedStatement stmt=conn.prepareStatement("select name,rollno,subject,marks as
        "roll=rollno and marks in (select max(marks) from results group by subject);")
    ResultSet rs=stmt.executeQuery();

    System.out.println("NAME \tROLLNO \tSUBJECT \tHIGHEST");
    while(rs.next()){
        System.out.println(rs.getString(1)+"\t"+rs.getInt(2)+"\t"+rs.getString(3)+"\t\
    }
```

```
run:
NAME    ROLLNO  SUBJECT         HIGHEST
suman   2       maths           50
rahul   5       english         50
rahul   5       hindi           49
rahul   5       maths           50
BUILD SUCCESSFUL (total time: 1 second)
```

```
package jdbc3;

import java.sql.*;



public class toppers {

        public static void main(String[] args){



        try{

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

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

        PreparedStatement stmt=conn.prepareStatement("select name,rollno,subject,marks as \"Highest\" from
student, results where\n" +

          "roll=rollno and marks in (select max(marks) from results group by subject);");

        ResultSet rs=stmt.executeQuery();



        System.out.println("NAME \tROLLNO \tSUBJECT \tHIGHEST");

        while(rs.next()){

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

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

f. Find the average marks



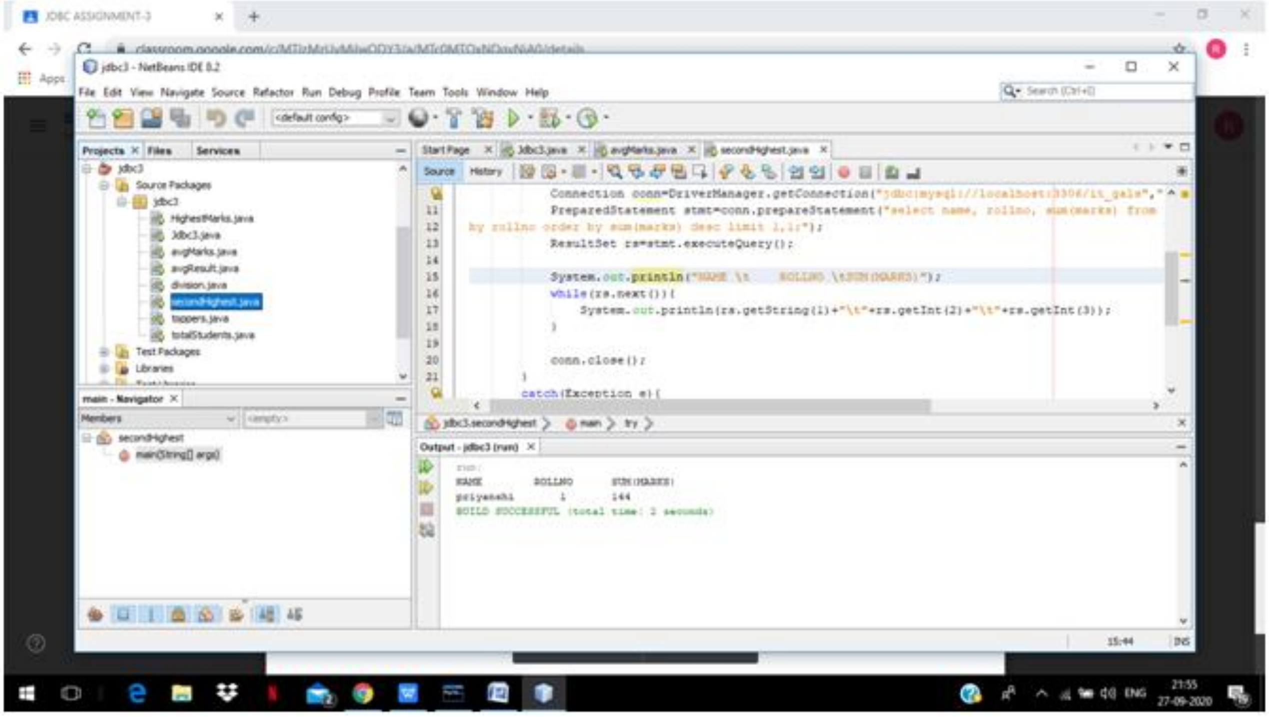
```
            while(rs.next()){
                System.out.println(rs.getString(1)+"\t\t"+rs.getInt(2));
            }

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

```
run:
SUBJECT         AVG(MARKS)
english         44
hindi           45
maths           48
BUILD SUCCESSFUL (total time: 1 second)
```

```
package jdbc3;

import java.sql.*;



public class avgResult {

        public static void main(String[] args){


        try{
```

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

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

        PreparedStatement stmt=conn.prepareStatement("select subject,avg(marks) from results group by subject;");

        ResultSet rs=stmt.executeQuery();


        System.out.println("SUBJECT \tAVG(MARKS)");

        while(rs.next()){

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

          }


        conn.close();

          }

     catch(Exception e){

        System.out.println(e);

          }

          }

}
```

g. Find the student getting second highest marks.



```
        Connection conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/it_gals","
        PreparedStatement stmt=conn.prepareStatement("select name, rollno, sum(marks) from
by rollno order by sum(marks) desc limit 1,1;");
        ResultSet rs=stmt.executeQuery();

        System.out.println("NAME \t    ROLLNO \tSUM(MARKS)");
        while(rs.next()){
            System.out.println(rs.getString(1)+"\t"+rs.getInt(2)+"\t"+rs.getInt(3));
        }

        conn.close();
    }
    catch(Exception e){
```

```
run:
NAME        ROLLNO      SUM(MARKS)
priyanshi       1       144
BUILD SUCCESSFUL (total time: 2 seconds)
```

```
package jdbc3;

import java.sql.*;



public class secondHighest {

      public static void main(String[] args){



      try{

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

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

      PreparedStatement stmt=conn.prepareStatement("select name, rollno, sum(marks) from student, results
where roll=rollno group\n" +

"by rollno order by sum(marks) desc limit 1,1;");

    ResultSet rs=stmt.executeQuery();



    System.out.println("NAME \t    ROLLNO \tSUM(MARKS)");

    while(rs.next()){

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

      }
```

```
          conn.close();

            }

        catch(Exception e){

          System.out.println(e);

            }

            }

}
```

2. Create a procedure in MySQL to count the number of Rows in table 'Student'. Use CallableStatement to call this method from Java code.

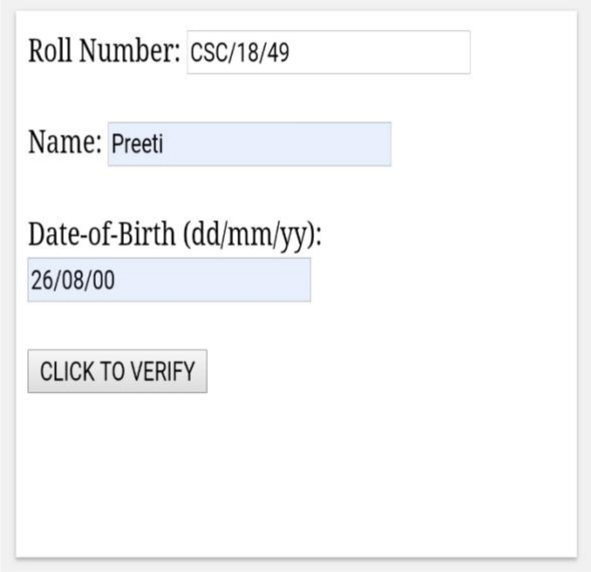
```
package     javaapplication6;     import

java.sql.*;             public             class

JavaApplication6 {



   public static void main(String[] args) {
       try{
Class.forName("com.mysql.cj.jdbc.Driver");
Connection
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/jdbc_ass2","root", "root"
);
CallableStatement stmt=conn.prepareCall("{call getTotal(?)}");

stmt.registerOutParameter(1,Types.INTEGER);

stmt.execute();
System.out.println("No of rows= "+stmt.getInt(1)); conn.close();

}       catch(Exception e)

       { System.out.println(e);}
   }
}
```

JavaScript

1. Create a student registration form. Create functions to perform the following checks:
   a. Roll number is a 7 digit numeric value
   b. Name should be an alphabetical value (String)
   c. DOB entered in dd/mm/yy format and should be display in words (e.g. Saturday, January 01, 2000)
   d. Check on non-empty fields

```
<!DOCTYPE html>
<html>
  <head>

    <title> Student Registration Form </title>

 <script>

      function check(){

      alert(“HELLO, WELCOME”);

      var rollno =document.myform.elements[0].value

      var uname=document.myform.elements[1].value

      var dob = document.myform.elements[2].value

//Roll No validation
        if(rollno =="” || rollno == null)  {
               alert(“Your rollno field is empty”);
            document.myform.elements[0].focus();
            }

        else
        {   if(/^\d+$/.test(rollno))
            {      if(rollno.length==7)
                         alert(“Your rollno field is valid !!!”);
                   else
                   alert(“Warning : Roll No. should contain 7 digits”);
            }

            else
                   alert(“Rollno should be a number”);
         }

//Name validation
         if(uname =="” ||uname== null)

         {  alert(“Your name field is empty”);
        document.myform.elements[1].focus();

        }

 else

   {

        if(/^[a-zA-Z ]+$/.test(uname))

        {

                alert(“Your name is valid !!!”);
        }
        else{
                alert(“Your name is not valid !!!”);

        }

    }

//DoB validation

    if(dob ==”” ||dob== null)

    {   alert(“Your Date-of-Birth field is empty”);

        document.myform.elements[2].focus();

        }

    else

    {//regular expression

        var frmat= /^(0?[1-9]|[12][0-9]|3[01])[\/](0?[1-9]|1[012])[\/]\d{2}$/;


        //Check whether valid dd/MM/yy Date format

        if(frmat.test(dob))

        {

                alert(“Your date of birth is valid “);
```

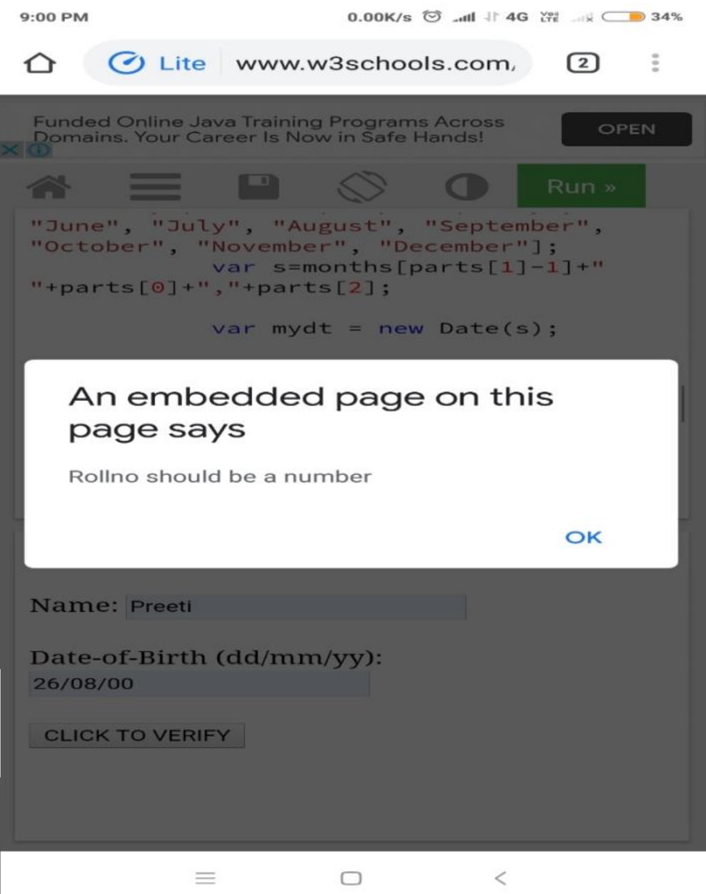
```
                                  var parts=dob.split('/');
                                         var months = ["January", "February", "March", "April", "May", "June",
"July", "August", "September", "October", "November", "December"];
              var s=months[parts[1]-1]+" "+parts[0]+","+parts[2];
                                  var mydt = new Date(s);

                                  var days =
["Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday"];
L

                                  var day = days[mydt.getDay()];
                                  var mon = months[mydt.getMonth()];
                    var dt = mydt.getDate();
                                  var yr = mydt.getFullYear();
                    alert("You were born on "+day+", "+dt+" "+mon+" "+ yr);
                    }
                    else
                                  alert("Your date of birth is not valid ");
            }

                    alert("END");
        }

</script>
   </head>

   <body>
      <div>
         <form name=myform>

         Roll Number: <input type="text" name="rollno" value"" ><br/><br/>
         Name: <input type="text" name="uname" value="" ><br/><br/>
         Date-of-Birth (dd/mm/yy): <input type="text" name="dob" value=""><br/><br/>
         <button type="button" name="Submit" onclick = "return check()"> CLICK TO
VERIFY</button>

         </form>
      </div>

   </body>

</html>
```



Roll Number: CSC/18/49

Name: Preeti

Date-of-Birth (dd/mm/yy):
26/08/00



9:00 PM
0.00K/s 4G 34%
Lite
www.w3schools.com/
2
Funded Online Java Training Programs Across Domains. Your Career Is Now in Safe Hands!
OPEN
Run »

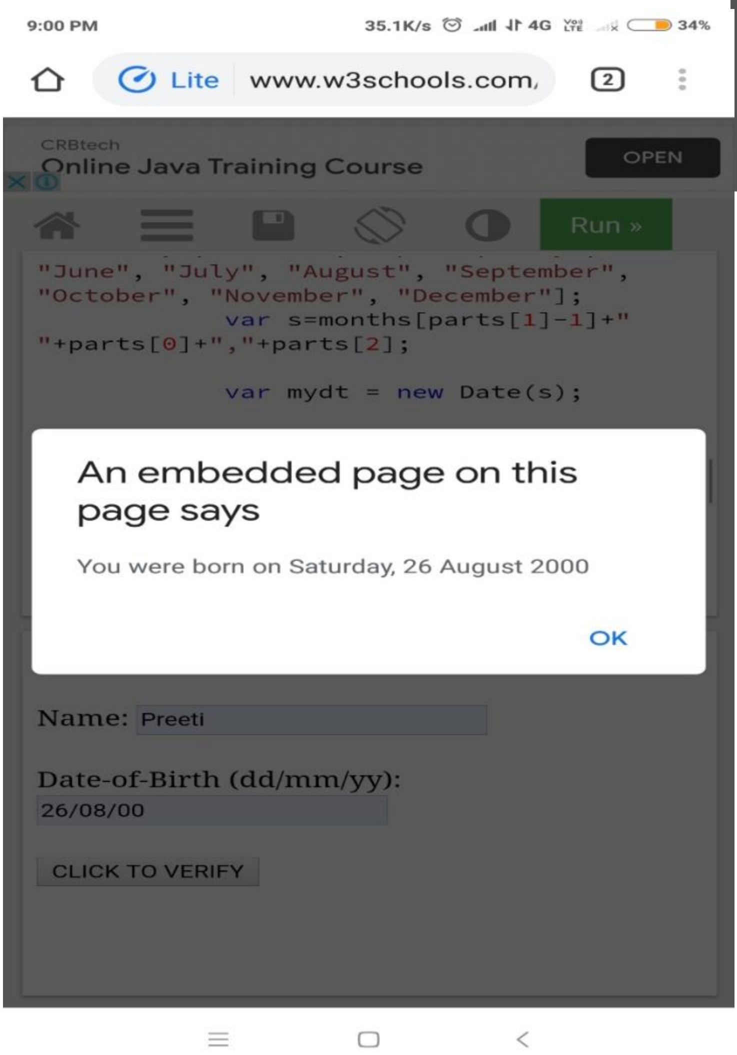
```
"June", "July", "August", "September",
"October", "November", "December"];
            var s=months[parts[1]-1]+"
"+parts[0]+","+parts[2];

            var mydt = new Date(s);
```

An embedded page on this page says

Rollno should be a number

Name: Preeti

Date-of-Birth (dd/mm/yy):
26/08/00



```
"June", "July", "August", "September",
"October", "November", "December"];
            var s=months[parts[1]-1]+"
"+parts[0]+","+parts[2];

            var mydt = new Date(s);
```

An embedded page on this page says

You were born on Saturday, 26 August 2000

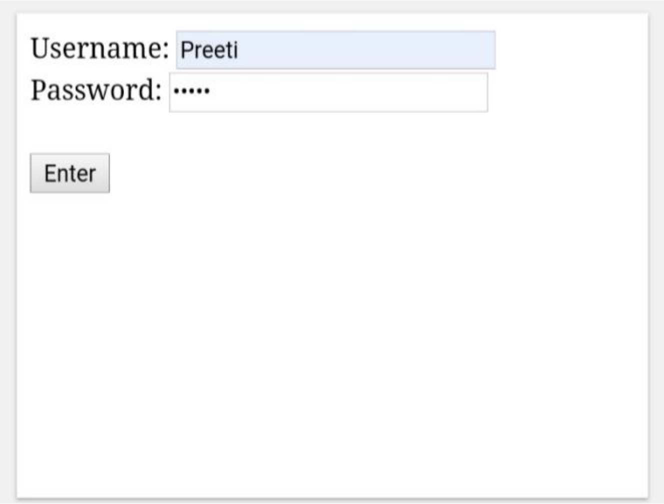
Name: Preeti

Date-of-Birth (dd/mm/yy):
26/08/00

Q2) Implement a Static Password Protection

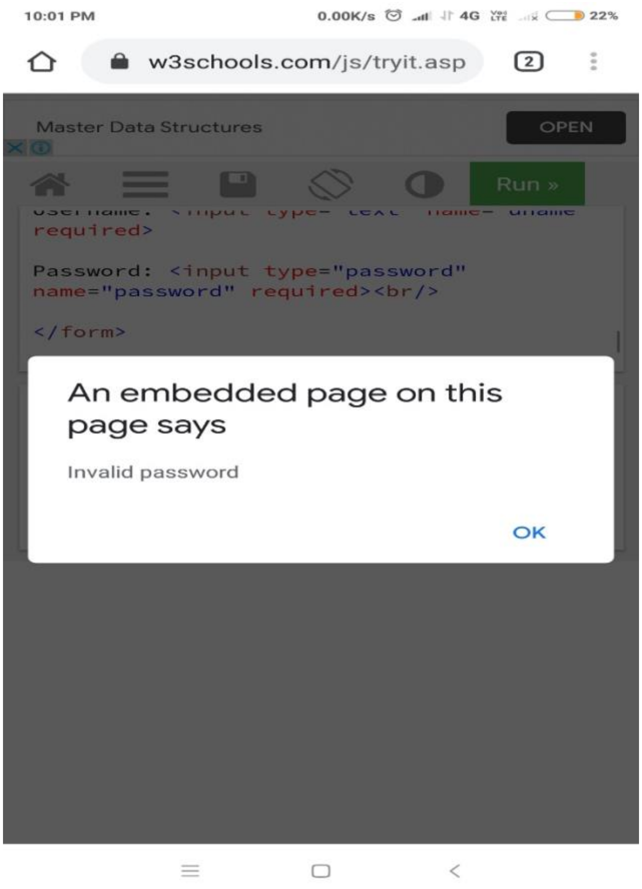
```
<!DOCTYPE html>
<html>
<body>
<script>
function mycheck(){
var name=document.myform.uname.value;
var password=document.myform.password.value;
if (name!=”Preeti”){
  alert(“Incorrect username”);
   }
else if(password!=”12345”){
  alert(“Invalid password”);
}
else{
alert(“You’re Welcome”);
}
}
</script>
<body>
<form name=”myform” >
Username: <input type=”text” name=”uname” required>
Password: <input type=”password” name=”password” required><br/>
</form>
 <button onclick=”mycheck()”> Enter </button>
</body
```

```
</html>
```



Username: Preeti

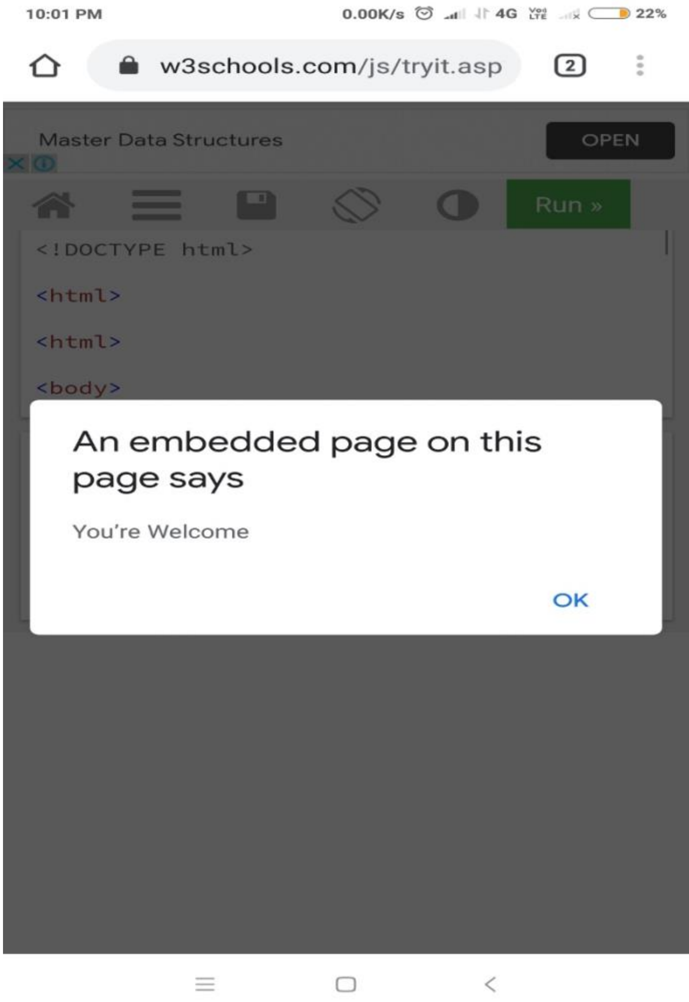
Password: •••••



```
Username: <input type="text" name="uname"
required>

Password: <input type="password"
name="password" required><br/>

</form>
```

An embedded page on this page says

Invalid password



```
<!DOCTYPE html>

<html>

<html>

<body>
```

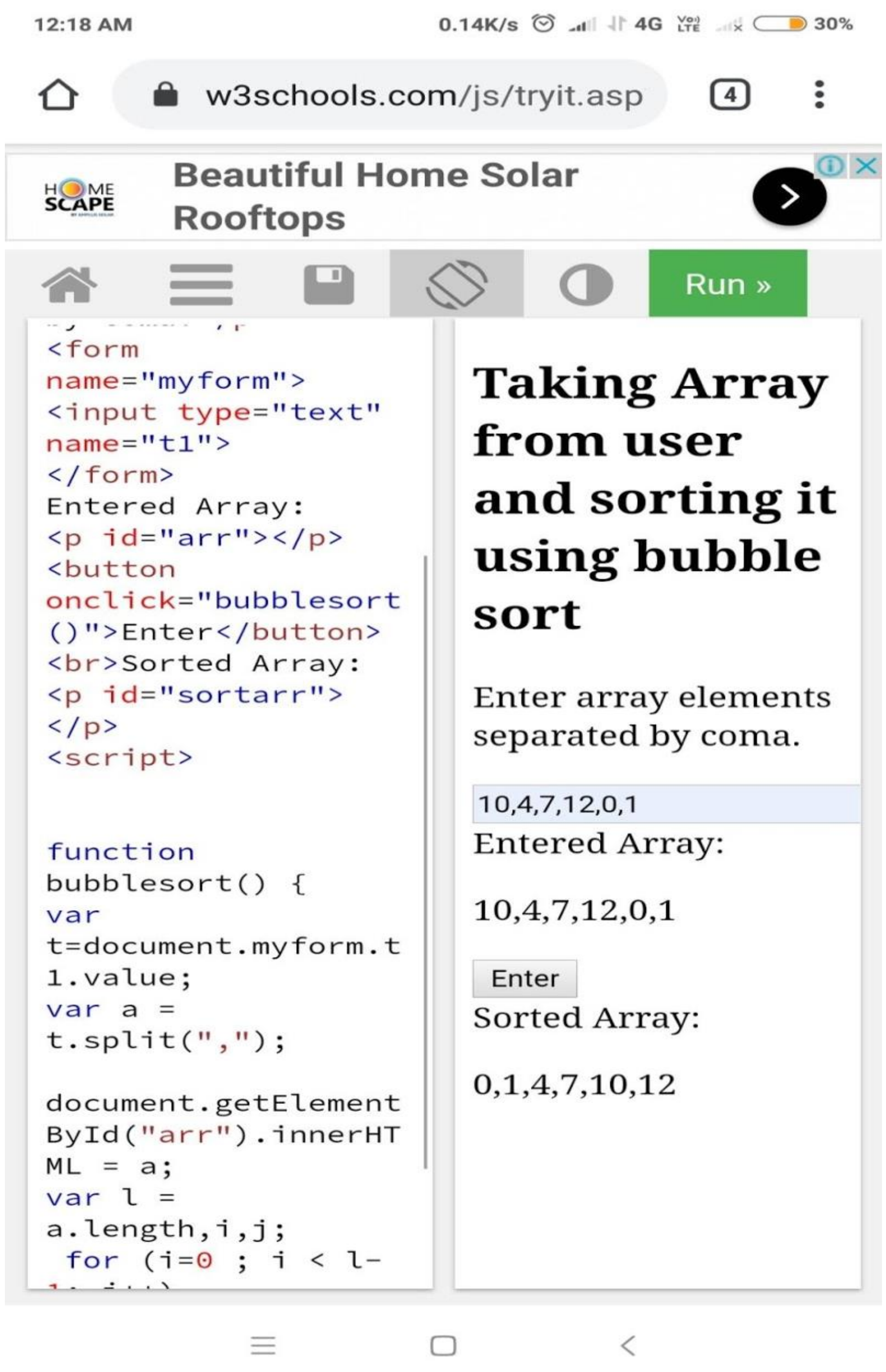
An embedded page on this page says

You're Welcome

3. Write a java script to sort an array using bubble sort. Take the number of elements and array from user.

```
 <!DOCTYPE html>
<html>
<body>

<h2>Taking Array from user and sorting it using bubble sort</h2>

<p>Enter array elements separated by coma.</p>
<form name="myform">
<input type="text" name="t1">
</form>
Entered Array:
<p id="arr"></p>

<button onclick="bubblesort()">Enter</button>
Sorted Array:
<p id="sortarr"></p>

<script>
function bubblesort() {
var t=document.myform.t1.value;
var a = t.split(",");
document.getElementById("arr").innerHTML = a;
var l = a.length,i,j;
 for (i=0 ; i < l-1; i++)
 {
      for (j=0; j <(l-i-1); j++)
           {
           if (parseInt(a[j]) >parseInt( a[j+1]))
                    {
```

```
                                var t = a[j];
                                           a[j] = a[j+1];
                                          a[j+1] = t;
                      }
        }
    }
  document.getElementById("sortarr").innerHTML = a;
}
</script>
</body>
</html>
```



```
<form
name="myform">
<input type="text"
name="t1">
</form>
Entered Array:
<p id="arr"></p>
<button
onclick="bubblesort
()">Enter</button>
<br>Sorted Array:
<p id="sortarr">
</p>
<script>


function
bubblesort() {
var
t=document.myform.t
1.value;
var a =
t.split(",");

document.getElement
ById("arr").innerHT
ML = a;
var l =
a.length,i,j;
 for (i=0 ; i < l-
```

Taking Array from user and sorting it using bubble sort

Enter array elements separated by coma.

10,4,7,12,0,1

Entered Array:

10,4,7,12,0,1

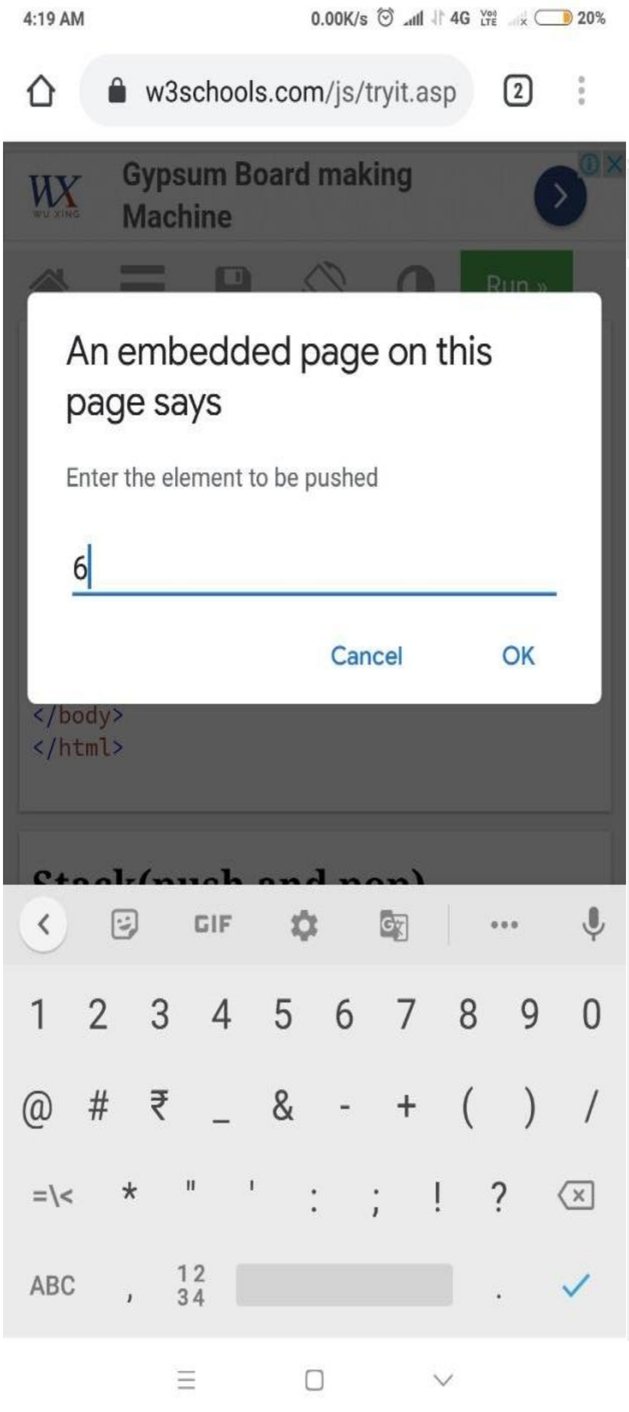
Enter

Sorted Array:

0,1,4,7,10,12

4. Write a JavaScript to implement stack methods (push and pop).

```
<!DOCTYPE html>
<html>
<head>
<script>
Var stack=[];
Var t=-1;var txt=””;
Function push() {
  Var e=prompt(“Enter the element to be pushed”);
if(e!=""){
  t=t+1;
  stack[t]=e;


  }
  else{
  alert("Value not entered");
  }
stack.forEach(display);
document.getElementById("dis").innerHTML = txt;
txt="";
}
function pop() {
    if(t!=-1){
    alert(" Element popped= "+ stack[t]);
t=t-1;
    stack.pop();
    }
    else{
        alert("Stack is empty");
    }
    stack.forEach(display);
document.getElementById("dis").innerHTML = txt;txt="";

}
function display(value) {
  txt = value + "<br>"+txt;


    }
</script>
</head>
<body>
<h2>Stack(push and pop)</h2>

<p id="demo">Current elements of stack-</p>
<p id="dis"> Empty </p>
<button type="button" onclick="push()">PUSH an element in STACK</button><button type="button"
onclick="pop()">POP an element from STACK</button>

</body>
</html>
```

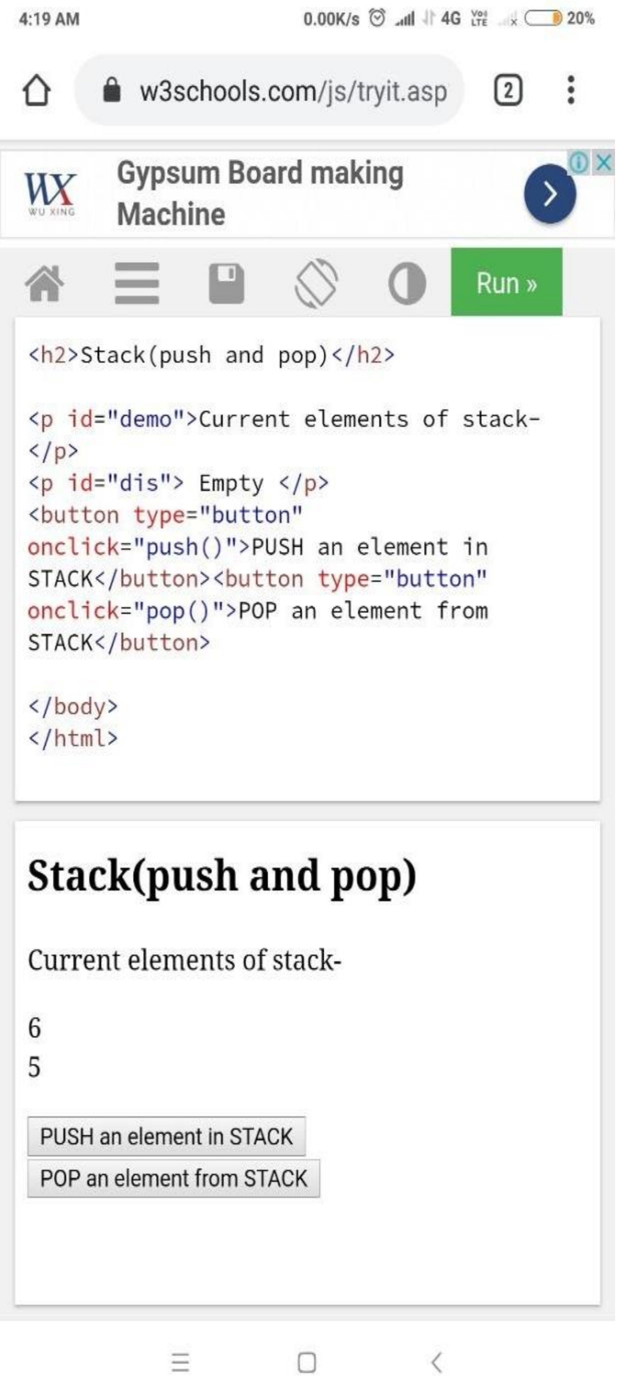


An embedded page on this page says

Enter the element to be pushed

6

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



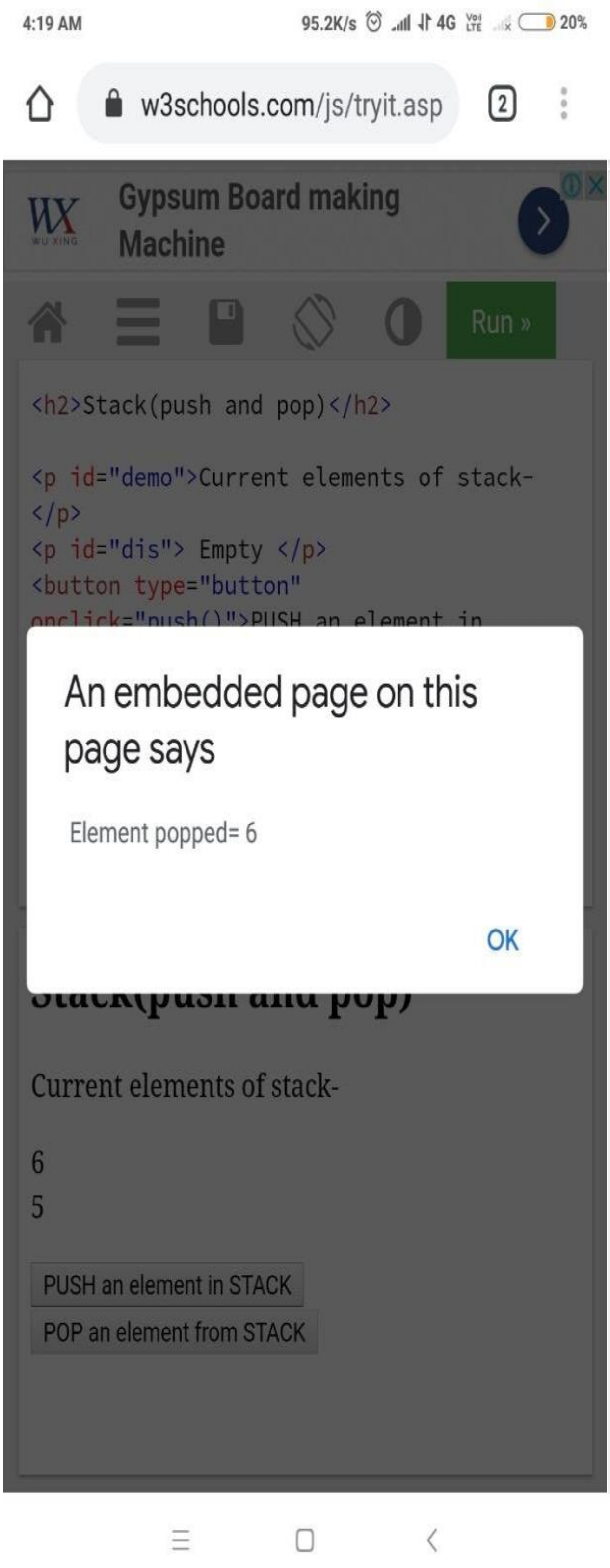
```
<h2>Stack(push and pop)</h2>

<p id="demo">Current elements of stack-
</p>
<p id="dis"> Empty </p>
<button type="button"
onclick="push()">PUSH an element in
STACK</button><button type="button"
onclick="pop()">POP an element from
STACK</button>

</body>
</html>
```

Stack(push and pop)

Current elements of stack-

6
5

4:19 AM
0.00K/s 4G 20%
w3schools.com/js/tryit.asp
Gypsum Board making Machine
Run »
PUSH an element in STACK
POP an element from STACK



```
<h2>Stack(push and pop)</h2>

<p id="demo">Current elements of stack-
</p>
<p id="dis"> Empty </p>
<button type="button"
onclick="push()">PUSH an element in
```

An embedded page on this page says

Element popped= 6

OK

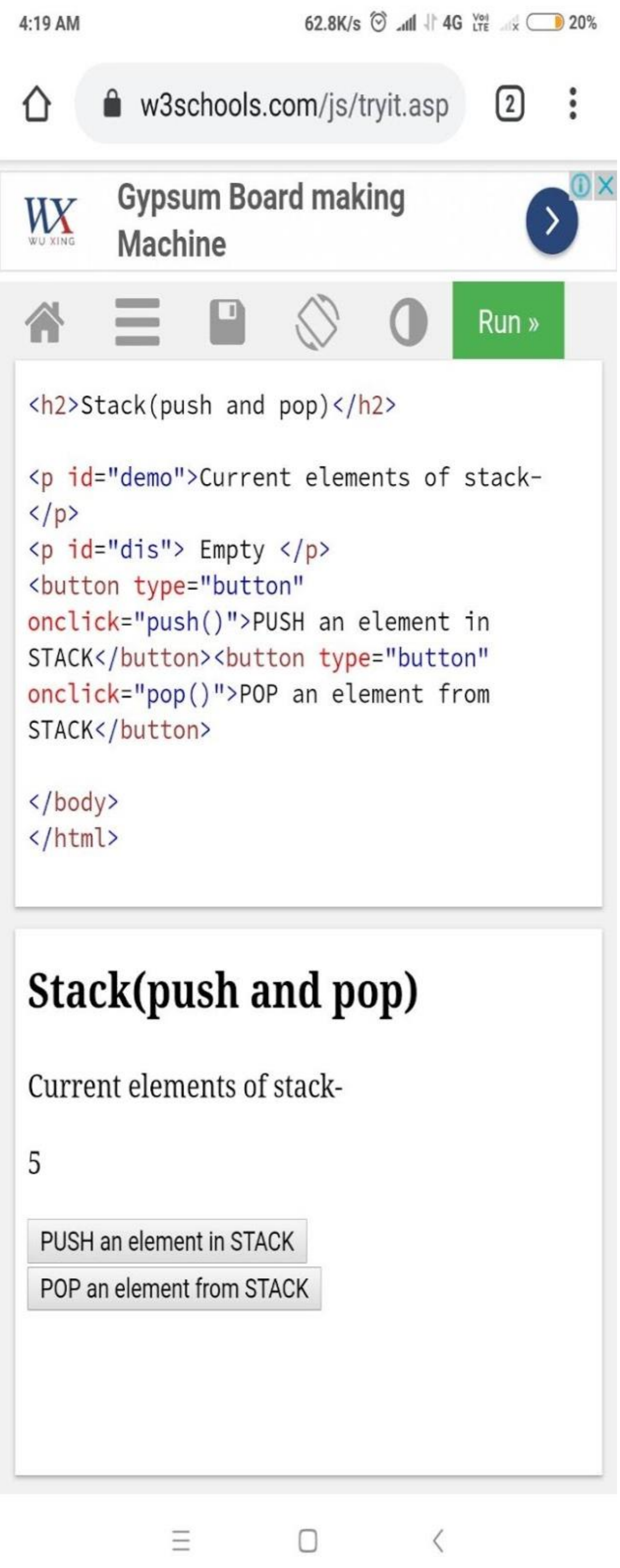
Stack(push and pop)

Current elements of stack-

6
5

PUSH an element in STACK

POP an element from STACK



```
<h2>Stack(push and pop)</h2>

<p id="demo">Current elements of stack-
</p>
<p id="dis"> Empty </p>
<button type="button"
onclick="push()">PUSH an element in
STACK</button><button type="button"
onclick="pop()">POP an element from
STACK</button>

</body>
</html>
```

Stack(push and pop)

Current elements of stack-

5

PUSH an element in STACK

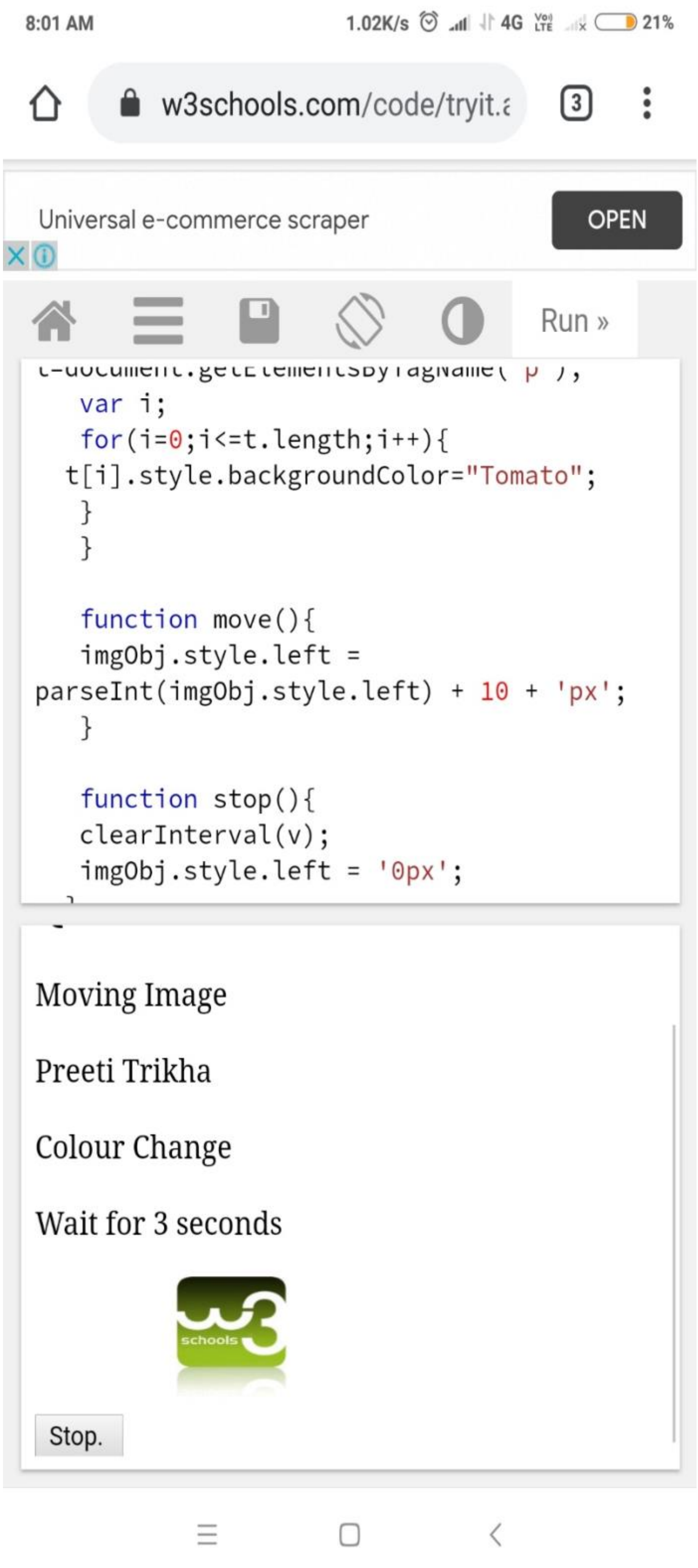
POP an element from STACK

5. Write a JavaScript

a. to change the color of text using setTimeOut()

b. to move an image across screen using setInterval()

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript Animation</title>
</head>

<body>

<h2>Q5</h2>

<p>Moving Image</p>
<p>Preeti Trikha</p>
<p>Colour Change</p>
<p>Wait for 3 seconds</p>

<img id="myImage"
src="https://www.w3schools.com/images/w3schools_green.jpg"  alt="HTML
tutorial" style="width:62px;height:62px;">
<br>
<button onclick="stop()">Stop. </button>

 <script>
   imgObj =  document.getElementById('myImage');
    imgObj.style.position= "relative";
   imgObj.style.left = '0px';

    var v=setInterval(move,200);

  var y=setTimeout(change,3000);

   function change(){
   var t=document.getElementsByTagName("p");
   var i;
   for(i=0;i<=t.length;i++){
  t[i].style.backgroundColor="Tomato";
   }
}

   function move(){
   imgObj.style.left = parseInt(imgObj.style.left) + 10 + 'px';
   }

   function stop(){
   clearInterval(v);
imgObj.style.left = '0px';
  }

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



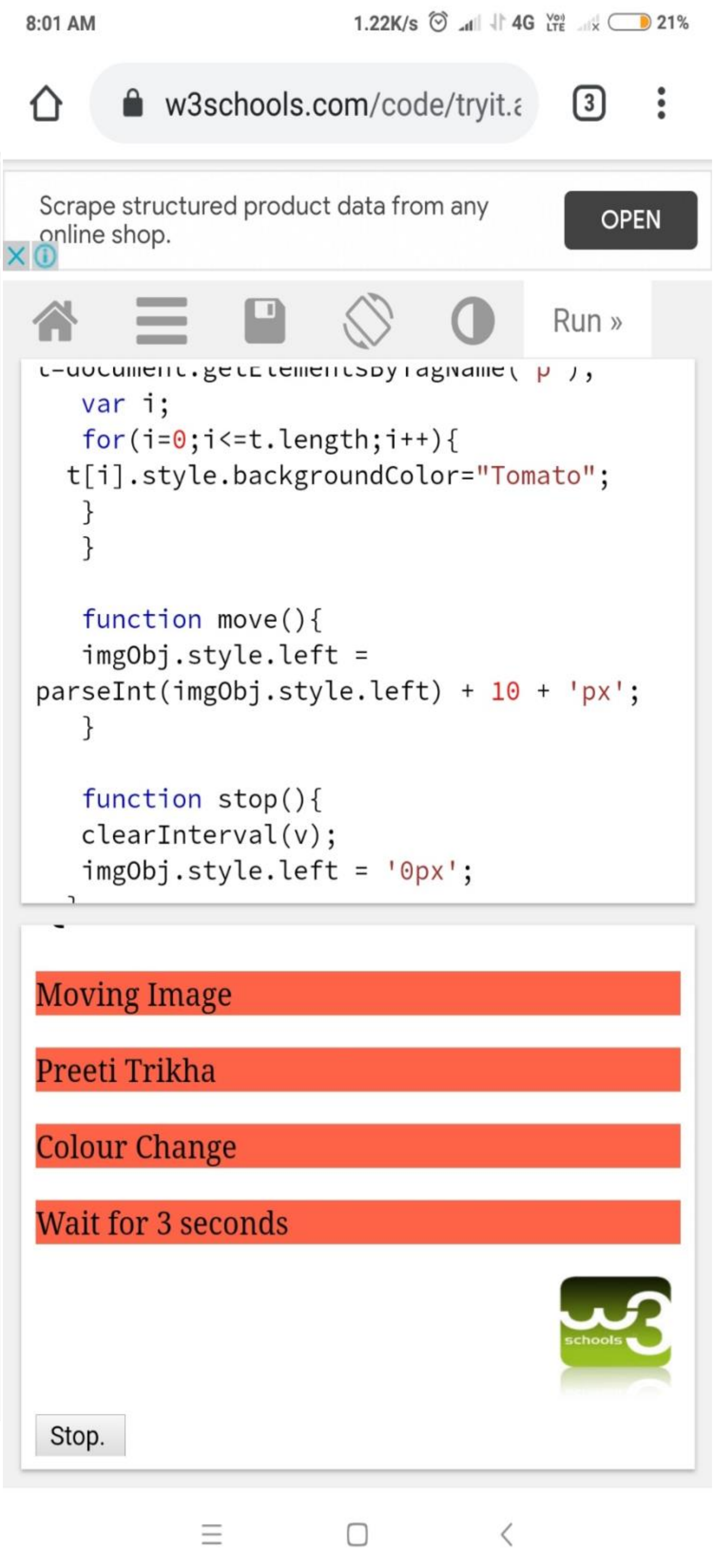
```
t=document.getElementsByTagName('p');
   var i;
   for(i=0;i<=t.length;i++){
  t[i].style.backgroundColor="Tomato";
   }
   }

   function move(){
   imgObj.style.left =
parseInt(imgObj.style.left) + 10 + 'px';
   }

   function stop(){
   clearInterval(v);
   imgObj.style.left = '0px';
  }
```

Moving Image

Preeti Trikha

Colour Change

Wait for 3 seconds



```
t=document.getElementsByTagName('p');
   var i;
   for(i=0;i<=t.length;i++){
  t[i].style.backgroundColor="Tomato";
   }
   }

   function move(){
   imgObj.style.left =
parseInt(imgObj.style.left) + 10 + 'px';
   }

   function stop(){
   clearInterval(v);
   imgObj.style.left = '0px';
  }
```

Moving Image

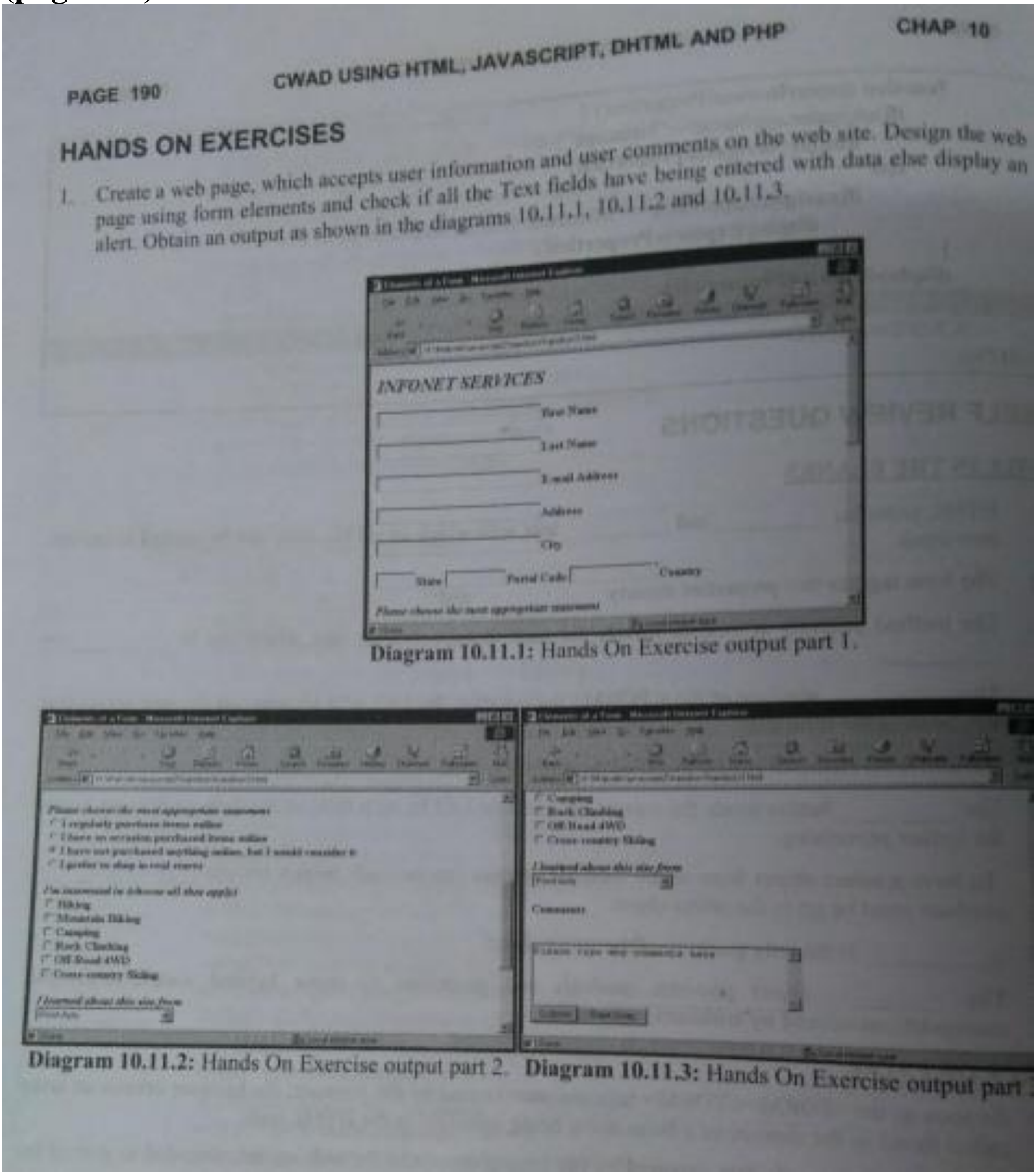
Preeti Trikha

Colour Change

Wait for 3 seconds

L

6. Implement the question no. 1 of hands on exercises of chapter 10 (page 190).



PAGE 190 CWAD USING HTML, JAVASCRIPT, DHTML AND PHP CHAP 10

HANDS ON EXERCISES

1. Create a web page, which accepts user information and user comments on the web site. Design the web page using form elements and check if all the Text fields have being entered with data else display an alert. Obtain an output as shown in the diagrams 10.11.1, 10.11.2 and 10.11.3.

INFONET SERVICES

First Name

Last Name

E-mail Address

Address

City

State Postal Code Country

Please choose the most appropriate statement

Diagram 10.11.1: Hands On Exercise output part 1.

Please choose the most appropriate statement

○ I regularly purchase items online

○ I have on occasion purchased items online

● I have not purchased anything online, but I would consider it

○ I prefer to shop in retail stores

I'm interested in (choose all that apply)

□ Hiking

□ Mountain Biking

□ Camping

□ Rock Climbing

□ Off-Road 4WD

□ Cross-country Skiing

I learned about this site from

Print Ads

Diagram 10.11.2: Hands On Exercise output part 2.

□ Camping

□ Rock Climbing

□ Off-Road 4WD

□ Cross-country Skiing

I learned about this site from

Print Ads

Comments

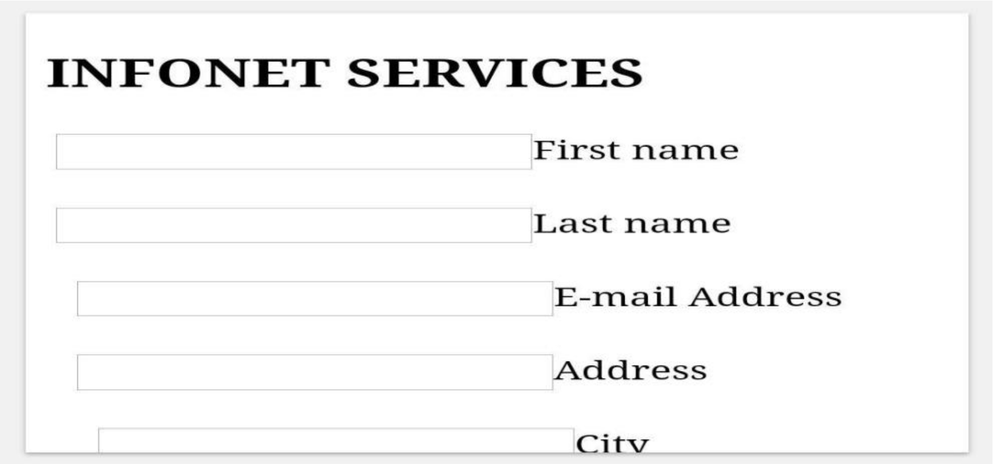
Please type any comments here

Submit Start Over

Diagram 10.11.3: Hands On Exercise output part

```
<!DOCTYPE html>
<html>
<body>

<h2>INFONET SERVICES</h2>

<form action=”/action_page.php”>
  <input type=”text” id=”fname”
name="fname"><label for="fname">First name</label><br><br>
   <input type="text" id="lname" name="lname"><label for="lname">Last name</label><br><br>
 <input type="email" id="ema" name="ema"><label for="ema">E-mail Address</label><br><br>
    <input type="text" id="addr" name="addr"><label for="addr">Address</label><br><br>
     <input type="text" id="city">City</label><br><br>
<input type="text" id="state" name="state"><label for="state">State</label><inputtype="text" id="pc"
name="pc"><label for="pc">Postal Code</label> <input type="text" id="country" name="country"><label
for="country">Country</label><br><br>

Please, choose the most appropriate statement.<br
<input type="radio" id="r1" name="statement" value="I regularly purchase these items online"> <label
for="r1"> "I regularly purchase these items online"</label><br>
 <input type="radio" id="r2" name="statement" value="I have on occasions purchased items online">
<label for="r2">"I have on occasions purchased items online"</label><br>
 <input type="radio" id="r3" name="statement" value="I have not purchased anything online,but I would
consider it"> <label for="r3"> "I have not purchased anything online,but I would consider it"</label><br>
 <input type="radio" id="r4" name="statement" value="I prefer to shop in real stores "> <label for="r4">
"I prefer to shop from real store."</label><br>
 <br>
I am interested in(choose all that apply):<br>


<input type="checkbox" id="c1" name="c1" value="Hiking">
  <label for="c1">Hiking</label><br>
<input type="checkbox" id="c2" name="c2" value="Mountain biking">
 <label for="c2"> Mountain Biking</label><br>
<input type="checkbox" id="c3" name="c3" value="Camping">
  <label for="c3">Camping</label><br><input type="checkbox" id="c4" name="c4" value="Rock
climbing">
```

```
 <label for="c4"> Rock Climbing </label><br>

<input type="checkbox" id="c5" name="c5" value="Off-road 4WD">

  <label for="c5">Off-Road 4WD</label><br><input type="checkbox" id="c6" name="c6" value="Cross Country Skiing">

  <label for="c6"> Cross country skiing  </label><br>

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

<Input type=”Reset” value=”Start Over”>

</form>


</body>

</html>
```



INFONET SERVICES

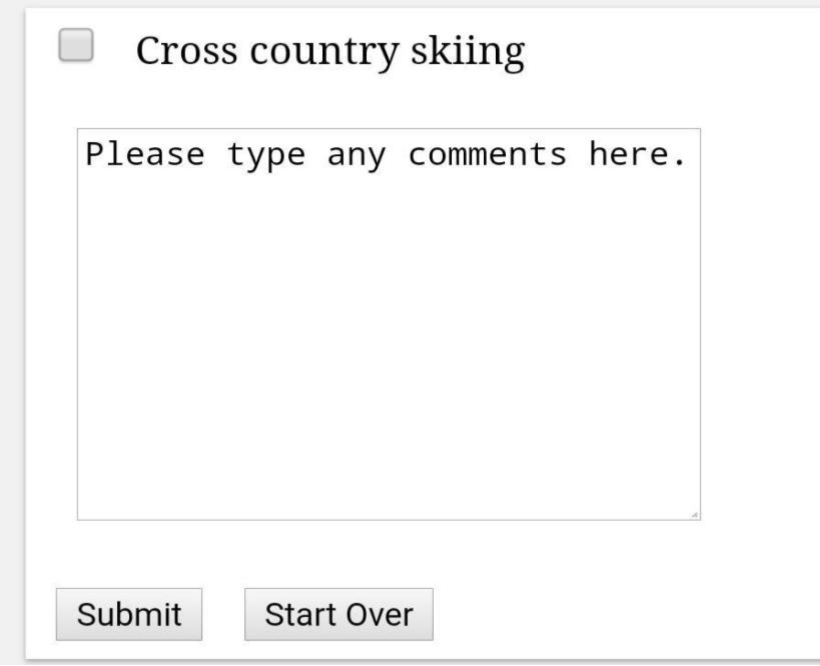
First name

Last name

E-mail Address

Address

City



□ Cross country skiing

Please type any comments here.

JSP

QUESTION-1

Display the pattern:
1
1 2
1 2 3
Take ‘n’ in a textbox from user.

Display this pattern using
--- Scriptlets----

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8" language="java"%>


<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <h1>Hello World!</h1>
    <form action="q1.jsp" method="GET">
      <input type =text" name="num" onclick = "this.select()"/>
      <input type="submit" value="Press to see the pattern"/>
    </form>
    <%
      String r = request.getParameter("num");
      if(r!=null && r!=""){              int n = Integer.parseInt(r);

out.println("<U><B>Printing the pattern using scriptlets.</U></B><BR/><BR/>");

        if(n>=1){                  for(int i=0;i<n;i++){

for(int j=1;j<=i+1;j++)

out.println(j+"&nbsp;&nbsp;&nbsp;");

out.println("<BR/>");

          }
        }
      }
    %>



    <c:if test="${param.num!=null}">
      <BR/> <U><B> Printing pattern using for-Each loop </U></B><BR/>
      <c:forEach begin="1" end="${param.num}" var="i">
        <c:forEach begin="1" end="${i}" var="j">
          <c:out value="${j}"/>&nbsp;&nbsp;&nbsp;
        </c:forEach>
        <BR/>
      </c:forEach>
    </c:if>
  </body>
</html>
```

QUESTION-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

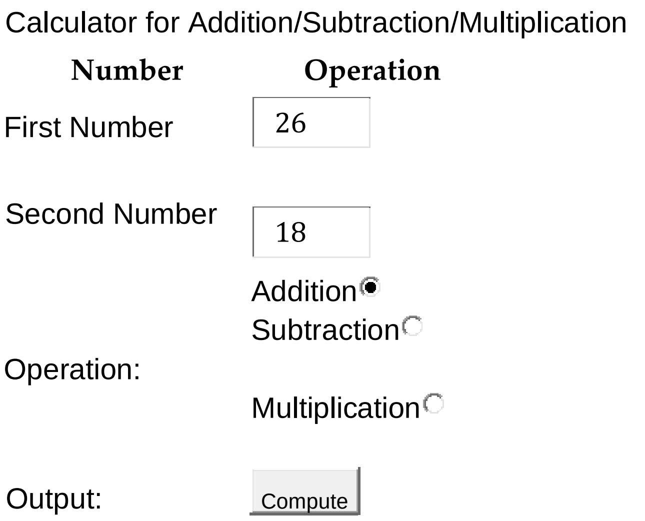
```
<!--
Make two files as follows:

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)

-->
<html>
  <head>
    <title>Operations Form Page</title>
  </head>
  <body>
```

```
    <div>Calculator for Addition/Subtraction/Multiplication</div>
    <form name="myForm" action="operate.jsp">
      <table>
        <thead>
          <tr>
            <th>Number</th>
            <th>Operation</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td>First Number</td>
            <td><input type="text" name="num1" value="0"
onfocus="this.select()"/></td>
          </tr>
          <tr>
            <td>Second Number</td>
            <td><input type="text" name="num2" value="0"
onfocus="this.select()"/></td>
          </tr>
          <tr>
            <td>Operation:</td>
            <td>
              Addition<input type="radio" name="operation" value="addition"
checked />
              <br/>Subtraction<input type="radio" name="operation"
value="subtraction" />
              <br/>Multiplication<input type="radio" name="operation"
value="multiplication" />
            </td>
          </tr>
          <tr>
            <td>Output:</td>
            <td><input type="submit" value="Compute" name="compute" /></td>
          </tr>
        </tbody>
      </table>


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

operate.jsp :

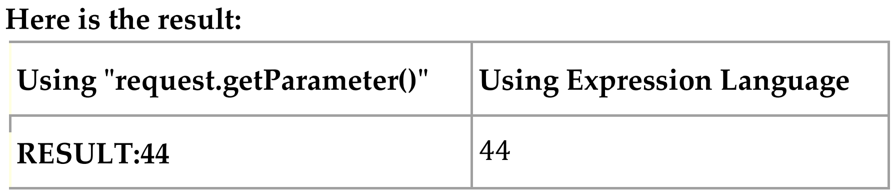
```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Output Page</title>
   </head>
   <body>
      <h1>Here is the result:</h1>
      <table border="1">
         <thead>
            <tr>
               <th>Using "request.getParameter()"</th>
               <th>Using Expression Language</th>
            </tr>
         </thead>
         <tbody>
            <tr>
               <td>
                  <%
                  int r=Integer.parseInt(request.getParameter("num1"));

         int s=Integer.parseInt(request.getParameter("num2"));

                     String op = request.getParameter("operation");
                      switch(op){

                         case "addition":

                         out.println("<B>RESULT:"+(r+s)+"</B>");

                        break;
                        case "subtraction": out.println("<B>RESULT:"+(r-s)+"</B>");
                        break;
                        case "multiplication": out.println("<B>RESULT:"+(r*s)+"</B>");
                        break;
                        default: out.println("<BR/>No operation<BR/>");
                     }
                  %>
                   <BR/>



               </td>
               <td>



                  <c:set var="oper" value="${param.operation}"/>
            <c:set var="num1" value= "${param.num1}"/>
            <c:set var="num2" value= "${param.num2}"/>
            <c:choose>
              <c:when test="${oper == 'addition'}">
                <c:out value="${param.num1 + param.num2}"/>
              </c:when>
              <c:when test= "${oper== 'subtraction'}">
                <c:out value="${param.num1 - param.num2}"/>
              </c:when>
              <c:when test= "${oper=='multiplication'}">
                <c:out value="${param.num1 * param.num2}"/>
              </c:when>
              <c:otherwise>
                <c:out value="No operation"/>
              </c:otherwise>
            </c:choose>
          </td>
        </tr>


      </tbody>
    </table>
  </body>
</html>
```

OUTPUT-



Calculator for Addition/Subtraction/Multiplication

Number	**Operation**
First Number	26
Second Number	18
Operation:	Addition ● Subtraction ○ Multiplication ○
Output:	Compute



Here is the result:

Using "request.getParameter()"	**Using Expression Language**
RESULT:44	44

QUESTION-3

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

---pagevalidate.jsp---

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core"  prefix="c"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <c:set var="uname" value="${param.uname}"/>
    <c:set var="dob" value="${param.dob}"/>
    <c:set var="Id" value="${param.Id}"/>
    <c:set var="lno" value="${param.lno}"/>
    <c:set var="food" value="${param.food}"/>



    <c:if var = "cur" test="${uname == '' || dob == '' || Id == ''  || lno == ' ' || food == ' '}">
        <p style="font: bolder; color: red">NO FIELD CAN BE LEFT UNFILLED!</p>
          <jsp:include page="newhtml.html"/>
    </c:if>


          <c:if test = "${cur==false}">
              <p style="font: bolder; color: darkblue">WELCOME </p>
              <c:out value="${uname}"/><br/><br/>
          </c:if>
  </body>
</html>






<html>
  <head>
```

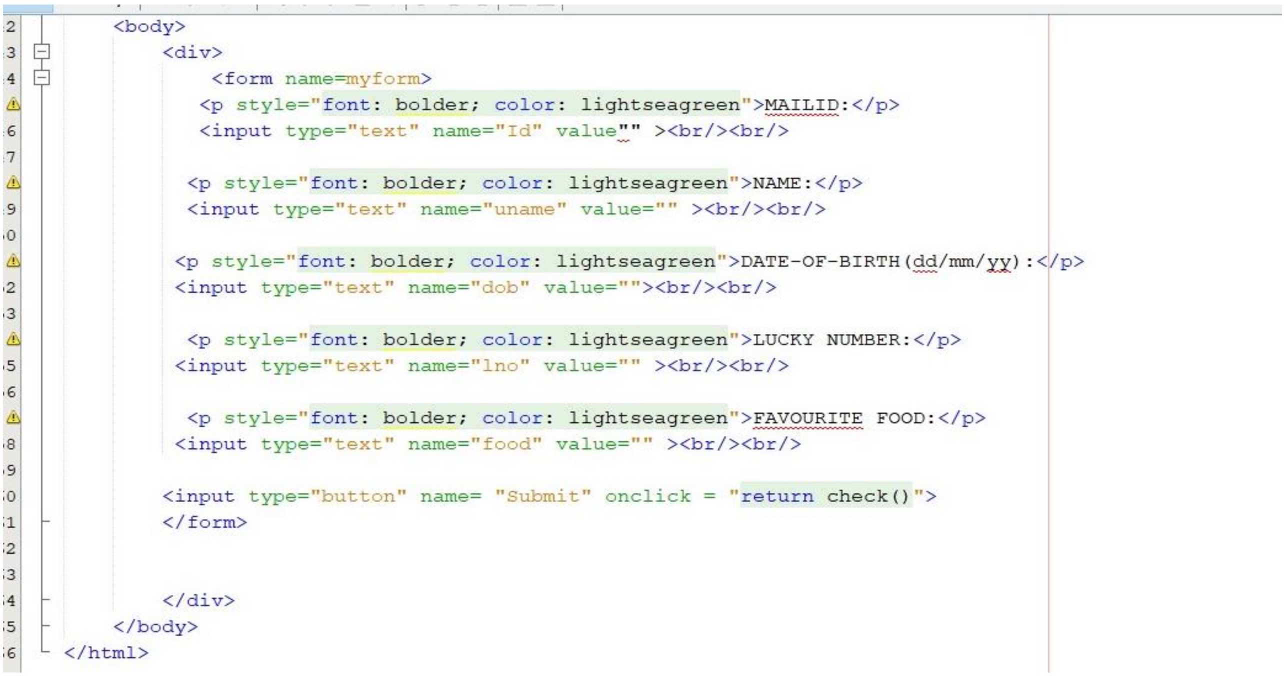
```
     <title> User validation Form </title>
 <script>



        function check()

       {


       alert("cdjkg");
       var Id =document.myform.elements[0].value
var uname=document.myform.elements[1].value
var dob = document.myform.elements[2].value
var lno = document.myform.elements[3].value
var food = document.myform.elements[4].value


           if(Id =="" || Id == null)
         {     alert("Your EmailId field is empty");
document.myform.elements[0].focus();
              }



          if(uname =="" ||uname== null)
   {   alert("Your name field is empty");
                document.myform.elements[1].focus();
              }


 else
  {


               if(/^[a-zA-Z ]+$/.test(uname))
              {
                     alert("Your name is valid !!!");
```

```
            }

            else

            {
                    alert("Your name is not valid !!!");

            }

        }


          if(dob =="" ||dob== null)

           {   alert("Your Date-of-Birth field is empty");

              document.myform.elements[2].focus();

            }
        else

        {//regular expression                  var frmat= /^(0?[1-9]|[12][0-
9]|3[01])[\/](0?[1-9]|1[012])[\/]\d{2}$/;




            //Check whether valid dd/MM/yy Date format
if(frmat.test(dob))

            {
                  alert("Your date of birth is valid ");

var parts=dob.split('/');                          var mydt = new

Date(parts[2],parts[1]-1,parts[0]);

                  var days =
["Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday"];


                    var day = days[mydt.getDay()];

                      var months = ["January", "February", "March", "April", "May",
"June", "July", "August", "September", "October", "November", "December"];

                  var mon =

months[mydt.getMonth()];                           var dt =

mydt.getDate();                            var yr =
mydt.getFullYear();                 alert("You were born on
"+day+", "+dt+" "+mon+" "+ yr);

                      }

                      else

                                     alert("Your date of birth is not valid ");

               }



                      alert("1111111111");

           }

       if(lno =="" ||lno== null)

                   {   alert("Your lucky number field is empty");

                         document.myform.elements[3].focus();

                      }

 else
   {


                       if(/^[0-9 ]+$/.test(lno))

                      {

                                    alert("Your number is valid !!!");


                      }

                      else
                      {

                                     alert("Your number is not valid !!!");

                      }
                }
       if(food =="" ||food== null)

                   {   alert("Your favourite food field is empty");

                         document.myform.elements[4].focus();
```

```
                    }

 else
   {

                      if(/^[a-zA-Z ]+$/.test(food))
                    {
                               alert("It is valid !!!");


                    }
                    else
                    {
                               alert("It is not valid !!!");
                    }
              }
</script>




  </head>
   <body>
      <div>
          <form name=myform>
            <p style="font: bolder; color: lightseagreen">MAILID:</p>
         <input type="text" name="Id" value"" ><br/><br/>


        <p style="font: bolder; color: lightseagreen">NAME:</p>

              <input type="text" name="uname" value="" ><br/><br/>


          <p style="font: bolder; color: lightseagreen">DATE-OF-BIRTH(dd/mm/yy):</p>
 <input type="text" name="dob" value=""><br/><br/>


        <p style="font: bolder; color: lightseagreen">LUCKY NUMBER:</p>
<input type="text" name="lno" value="" ><br/><br/>
```

```
        <p style="font: bolder; color: lightseagreen">FAVOURITE FOOD:</p>
<input type="text" name="food" value="" ><br/><br/>


            <input type="button" name= "Submit" onclick = "return check()">

         </form>




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

---newhtml.html---



```
    <body>
        <div>
            <form name=myform>
           <p style="font: bolder; color: lightseagreen">MAILID:</p>
           <input type="text" name="Id" value"" ><br/><br/>

          <p style="font: bolder; color: lightseagreen">NAME:</p>
          <input type="text" name="uname" value="" ><br/><br/>

         <p style="font: bolder; color: lightseagreen">DATE-OF-BIRTH(dd/mm/yy):</p>
         <input type="text" name="dob" value=""><br/><br/>

          <p style="font: bolder; color: lightseagreen">LUCKY NUMBER:</p>
         <input type="text" name="lno" value="" ><br/><br/>

          <p style="font: bolder; color: lightseagreen">FAVOURITE FOOD:</p>
         <input type="text" name="food" value="" ><br/><br/>

        <input type="button" name= "Submit" onclick = "return check()">
        </form>


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

QUESTION-4

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

```
<%@page import="java.time.LocalDateTime"%>
```

```
<%@page import="java.util.Calendar"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <div>Input your details</div>
    <form action="Q4.jsp" method="get">


    <table>
      <tr>
        <td>Name:</td><td><input type="text" name="uName" value =
"${param.uName!=null ? param.uName: ''}"
                    placeholder="Type your name here and press enter: "></td>
      </tr>
      <tr>


    </table>
    </form>


    <%        int hr =
LocalDateTime.now().getHour();




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


      if(uname!="" && uname!=null ){
if(hr<12 && hr>4)
out.println("<H2>Good Morning "+ uname
+"! </H2>");          else if(hr>12 &&
hr<16)             out.println("<H2>Good
Afternoon "+ uname +"! </H2>");
        else if(hr<20)
          out.println("<H2>Good Evening "+ uname +"! </H2>");
      }
    %>
  </body>
```

```
</html>
```

QUESTION-5

Let the user enter a word a in a textbox and let her/him select one of even or odd radio buttons. If she/he selects odd, check the odd positions in the word entered, if they all contain vowels, then display the message 'You win', else display 'You lose'. Similarly, if the user selects even, check for vowels in all even positions in the word entered. Use jstl's 'fn' library.

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>JSP Page</title>
   </head>
   <body>
      <div>!GAME!</div>
      <form action="Q5.jsp" method="get">


      <table>
         <tr>
            <td>Enter a word:</td><td><input type="text" name="word" value =
"${param.word!=null ? param.word: ''}"
                              placeholder="Type a word here and press enter"></td>
         </tr>
         <tr>
            <td>
               Select a Button:<BR/>
               <input type="radio" name="oddeven" value="odd"/> ODD <BR/>
               <input type="radio" name="oddeven" value="even"/> EVEN <BR/>
            </td>


      </table>
                              <input type="submit"/><br/>
```

```
</form>


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


<c:set var="winflag" value="t"/>
<c:set var="word" value="${param.word}"/>


<c:if test = "${param.oddeven == 'odd'}">
   <c:forEach var="i" begin="0" end="${param.word.length()-2}" step="2">
      <c:choose>
         <c:when test="${fn:substring(word,i,i+1)=='a' ||
fn:substring(word,i,i+1)=='e'|| fn:substring(word,i,i+1)=='i'|| fn:substring(word,i,i+1)=='o'||
fn:substring(word,i,i+1)=='u'}">
            <c:set var = "winflag" value = 't'/>
         </c:when>
         <c:otherwise>
            <c:set var = "winflag" value = 'f'/>
            <c:set var = "i" value = "${param.word.length()}"/>
         </c:otherwise>
      </c:choose>


   </c:forEach>
   <c:if test="${winflag == 't'}">
      <c:out value="You win in odd game! "/>
   </c:if>
   <c:if test="${winflag == 'f'}">
      <c:out value="You lose in odd game! "/>
   </c:if>
</c:if>
<c:if test = "${param.oddeven == 'even'}">
   <c:forEach var="i" begin="1" end="${param.word.length()-1}" step="2">
      <c:choose>
         <c:when test="${fn:substring(word,i,i+1)=='a' ||
fn:substring(word,i,i+1)=='e'|| fn:substring(word,i,i+1)=='i'|| fn:substring(word,i,i+1)=='o'||
fn:substring(word,i,i+1)=='u'}">
            <c:set var = "winflag" value = 't'/>
         </c:when>
         <c:otherwise>
            <c:set var = "winflag" value = 'f'/>
            <c:set var = "i" value = "${param.word.length()}"/>
         </c:otherwise>
      </c:choose>
   </c:forEach>
   <c:if test="${winflag == 't'}">
```

```
            <c:out value="You win in even game! "/>
          </c:if>
          <c:if test="${winflag == 'f'}">
            <c:out value="You lose in even game! "/>
          </c:if>
        </c:if>
    </body>
</html>
```

QUESTION-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.

```
---HelloTag.java---
package mypkg; import java.io.IOException;
import javax.servlet.jsp.tagext.*; public class
HelloTag extends SimpleTagSupport{
private String name = "";     public void
setName(String k){
        name=k;
    }
    public void doTag() throws IOException{
getJspContext().getOut().println("Hello "+name);
    }
}
```

---ChocoBean.java---

```
package mypkg;


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


public class ChocoBean extends SimpleTagSupport{
   private String texture ="";
public void setTexture(String
m){        texture = m;
   }
   public void doTag() throws IOException{


      JspWriter out =
getJspContext().getOut();
if(texture.equalsIgnoreCase("Chewy"))
out.println("FiveStar, BarOne");        else
if(texture.equalsIgnoreCase("Crunchy"))
out.println("Munch, Kitkat");


   }
}
```

---custom.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>
  <short-name>custom</short-name>
  <uri>/WEB-INF/tlds/custom</uri>
  <tag>
     <name>Hello</name>
     <tag-class>mypkg.HelloTag</tag-class>
     <attribute>
```

```
        <name>name</name>
        <required>true</required>
        <type>String</type>
    </attribute>
    <body-content>empty</body-content>
  </tag>
  <tag>
    <name>choco</name>
    <tag-class>mypkg.ChocoBean</tag-class>
    <attribute>
        <name>texture</name>
        <required>true</required>


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

---hc.jsp---

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>Showing Development and use of a tag library and a tag handler</h1>
        <%@taglib uri="/WEB-INF/tlds/custom.tld" prefix="m"%>
        <m:Hello name="Ankita"/>
        <m:choco texture="crunchy"></m:choco>


    </body>
</html>
```

-

QUESTION-7 AND QUESTION-8

Create a custom tag “substring” with 3 mandatory attributes “input”, “start”, “end” which will do substring operation on given input .Create a custom tag “reverse” with a mandatory attribute “input” to reverse a string.

---Substringreverse.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>
  <short-name>substringreverse</short-name>
  <uri>/WEB-INF/tlds/Substring_reverse</uri>
  <jsp-version>2.0</jsp-version>

  <tag>
      <name>Substring</name>
      <tag-class>pkg.CustomTag</tag-class>
      <body-content>scriptless</body-content>
      <attribute>
          <name>input</name>
          <fragment>true</fragment>
          <type>String</type>
          <required>true</required>
      </attribute>
      <attribute>
          <name>start</name>
          <fragment>true</fragment>
          <type>Integer</type>
          <required>true</required>
      </attribute>
      <attribute>
          <name>end</name>
          <fragment>true</fragment>
          <type>Integer</type>
          <required>true</required>
      </attribute>
  </tag>

  <tag>
      <name>Reverse</name>
      <tag-class>pkg.CustomTag</tag-class>
      <body-content>scriptless</body-content>
      <attribute>
          <name>s</name>
      <required>true</required>
      <fragment>true</fragment>
      <type>String</type>
    </attribute>
  </tag>
  </taglib>
```

---CustomTag.java---

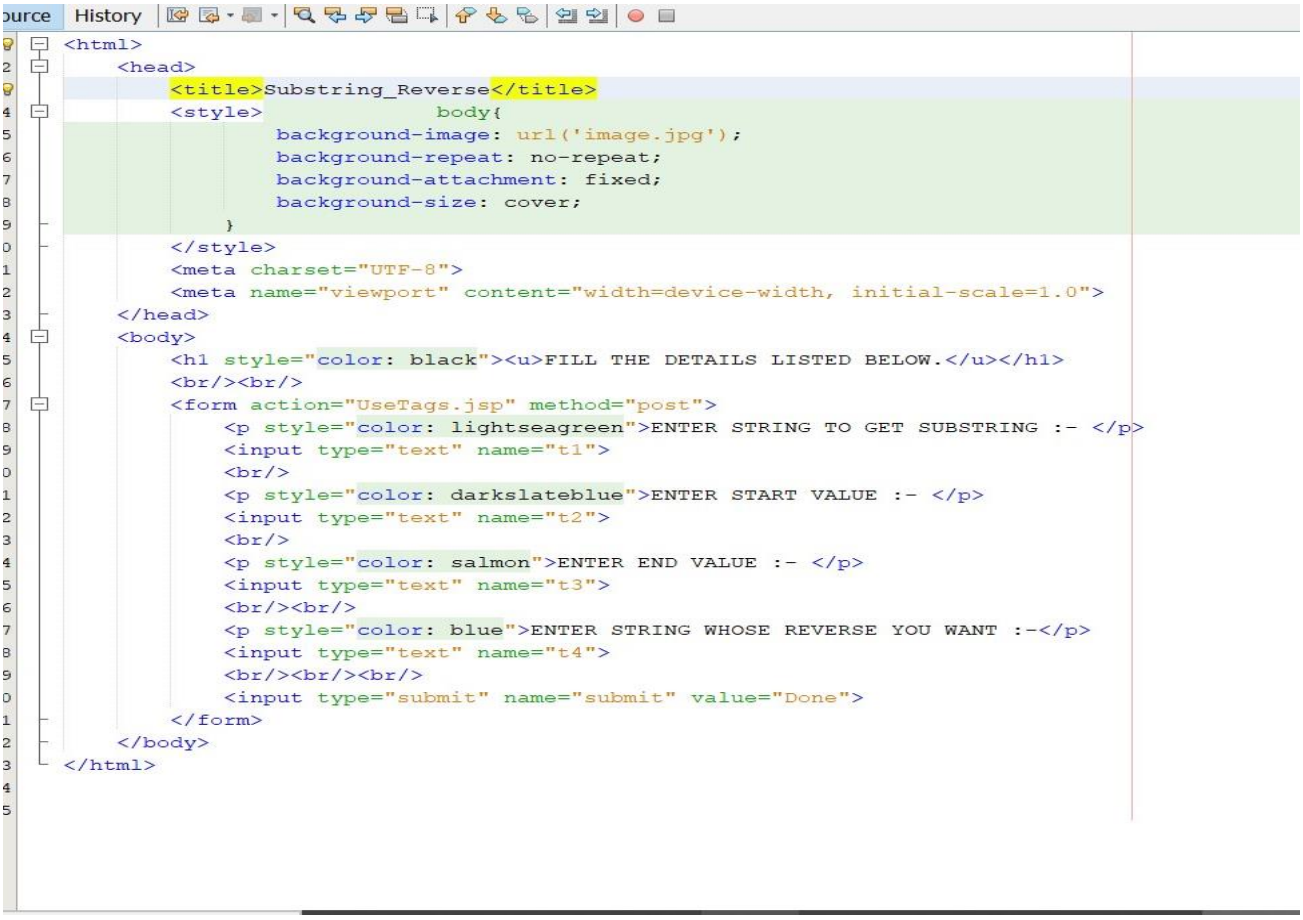
```
package pkg; import
javax.servlet.jsp.*; import
javax.servlet.jsp.tagext.*;
import java.io.*;  import
java.io.Serializable;

public class CustomTag extends SimpleTagSupport implements Serializable {         private String
input, s;      private int start, end;

   public void setS(String s) {
this.s = s;
   }


   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{
JspWriter out = getJspContext().getOut();
if(input!=null){
            String sub = input.substring(start, end);
out.println("Substring of the string is :- "+sub);
        }
        else if(s!=null){
            //String rev = s.substring(-1);
            StringBuilder inp = new
StringBuilder();                inp.append(s);
inp = inp.reverse();
out.println("Reversed String is :- "+ inp);

}
else
{
            getJspBody().invoke(sw);
getJspContext().getOut().println(sw.toString());
        }
    }

}
```

---UseTags.jsp---

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib uri="/WEB-INF/tlds/Substringreverse.tld" prefix="m"%>
<!DOCTYPE html>
<html>
    <head>
        <style>                         body{
background-image: url('image.jpg');
background-repeat: no-repeat;
background-attachment: fixed;

background-size: cover;

            }
        </style>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <%! String i, str, start,

end;          int s, e;%>


        <% i =

request.getParameter("t1");

str = request.getParameter("t4");

start = request.getParameter("t2");

s = Integer.parseInt(start);

end = request.getParameter("t3");

e = Integer.parseInt(end); %>

        <h1 style="color: blueviolet; text-align: center">THIS IS MY TAGS!</h1>
        <br/>
        <h3 style="color: seagreen">THIS IS SUBSTRING TAG.</h3>
        <br/>
        <h3 style="background-color: captiontext; color: steelblue"><m:Substring
input="<%= i%>" start="<%= s%>" end="<%= e%>">
            This is the body of substring tag.
        </m:Substring>
            </h3>
            <br/><br/>
             <h3 style="color: seagreen">THIS IS REVERSE TAG.</h3>
             <br/>
            <h3 style="color: darkcyan; background-color:
lightgoldenrodyellow"><m:Reverse s="<%= str%>">this id</m:Reverse></h3>
</body>
</html>
```

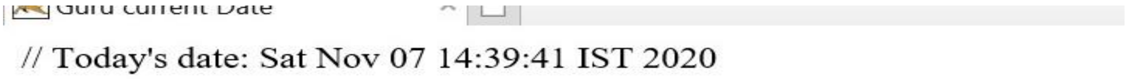
---index.html---



```
<html>
    <head>
        <title>Substring_Reverse</title>
        <style>             body{
                background-image: url('image.jpg');
                background-repeat: no-repeat;
                background-attachment: fixed;
                background-size: cover;
            }
        </style>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <h1 style="color: black"><u>FILL THE DETAILS LISTED BELOW.</u></h1>
        <br/><br/>
        <form action="UseTags.jsp" method="post">
            <p style="color: lightseagreen">ENTER STRING TO GET SUBSTRING :- </p>
            <input type="text" name="t1">
            <br/>
            <p style="color: darkslateblue">ENTER START VALUE :- </p>
            <input type="text" name="t2">
            <br/>
            <p style="color: salmon">ENTER END VALUE :- </p>
            <input type="text" name="t3">
            <br/><br/>
            <p style="color: blue">ENTER STRING WHOSE REVERSE YOU WANT :-</p>
            <input type="text" name="t4">
            <br/><br/><br/>
            <input type="submit" name="submit" value="Done">
        </form>
    </body>
</html>
```

QUESTION-9

Create a custom tag "today" that displays today's date and time.



// Today's date: Sat Nov 07 14:39:41 IST 2020

Question 10-

Ask a user's name and age on a HTML form. Then display Hello <uname> on a JSP. On the same page ask the product the user would like to buy. Then redirect to another JSP which would display: Hello <uname>, You have ordered <product>. (Use Session Scope Variable using setTag)

user.html --❼ page1.jsp -❼ page2.jsp

user.html:

```
<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project
Properties. To change this template file, choose Tools | Templates and
open the template in the editor.
-->
<html>
    <head>
        <title>User entry form</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <div>Enter your details: </div>
        <form name="form1" action="page1.jsp">
            <table>
                <tr><td>Your name:</td>
                    <td>
                        <input type="text" name="uname" value="" /><br/><br/>
                    </td>
                </tr>
                <tr>
                    <td>Your age:</td>
                    <td>
                        <input type="text" name="age" value="" /> <br/><br/>
                    </td>
                </tr>
             </table>


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

page1.jsp

```
<%--
    Document   : page1 --%>


<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
        <h1>Hello ${param.uname}</h1>
        <c:set var="session_uname" value="${param.uname}" scope="session"/>
        <form name="myForm" action="page2.jsp">
            Select a product that you would like to buy: <br/>
            <input type="radio" name="product" value="electronics" checked="checked"
/>Electronics <BR/>
            <input type="radio" name="product" value="cosmetics" />Cosmetics <BR/>
            <input type="submit" value="Submit" name="submit" />
        </form>


    </body>
</html>
```

page2.jsp"

```
<%--
    Document   : page2


--%>


<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>Hello ${sessionScope.session_uname}, You have ordered
${param.product}</h1>
    </body>
</html>
```