

# INTERNET TECHNOLOGIES PRACTICAL

/\*Kumar Prateek Viraj\*/

/\*Roll 38068\*/

//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.

---

/\*

BankAccount with Persistence

@author: Kumar Prateek Viraj

@StartDate: Oct26/2017

\*/

import java.io.\*;

import java.util.\*;

import javax.swing.\*;

class BankAccount{ //core banking facilities

    PrintStream pw=System.out;

    final double EPSILON=1E-14;

    protected String name;

    protected long accountNumber;

    protected float balance=0;

    public String

choiceList="\n1.AddAccount\n2.Withdraw\n3.Deposit\n4.MyAccount\n5.More

Options\n\n0.Persistence Exit";

    public String moreOptions="\n6.Minimum and Maximum Balance\n7.Accounts with  
specific balance\n8.Total Balance in Bank\n\n0.Persistence Exit";

    ArrayList <BankAccount> customer\_list;

    public BankAccount(String name,long accountNumber,float balance){

        this.accountNumber=accountNumber;

        this.name=name;

        this.balance = balance;

    }

    public String getName(){

        return this.name;

    }

    public long get\_Account\_no(){

        return this.accountNumber;

    }

    public float get\_Balance(){

        return this.balance;

```

    }
    public void setBalance(float newBal){
        this.balance=newBal;
    }

    public BankAccount(){//loads from file to arraylist
        BufferedReader in = null;
        customer_list=new ArrayList<BankAccount>();
        try {
            in = new BufferedReader(new FileReader("bankfile.txt"));
            String str;
            while ((str = in.readLine()) != null) {
                String[] temp_list=str.split(";");
                accountNumber=Long.parseLong(temp_list[1]);
                balance=Float.parseFloat(temp_list[2]);
                BankAccount customer = new
BankAccount(temp_list[0],accountNumber,balance);
                customer_list.add(customer);
            }
        } catch (FileNotFoundException e) {e.printStackTrace();}
        } catch (IOException e) { e.printStackTrace();}
    } finally {
        if (in != null) {
            try{ in.close();}
            } catch(Exception e){e.printStackTrace();}
        }
        for(BankAccount c: customer_list) pw.println(c.getName()+"
"+c.get_Account_no()+"\n");
    }

    void tax_deduction(){/*deducts 15% for 15000/- and above. 7% for 10000/- to
15000/- and 5% for 0- 10000/- */

}

float getTotalbalance(){
    float totalBal=0;
    for(BankAccount cu:customer_list){
        totalBal=totalBal+cu.get_Balance();
    }
    return totalBal;
}

void minMaxAmount(){
    float minBal=customer_list.get(0).get_Balance();
    float maxBal=minBal;
    for(BankAccount cu:customer_list){
        if(cu.get_Balance()<minBal) minBal=cu.get_Balance();
        else if(cu.get_Balance()>maxBal) maxBal=cu.get_Balance();
    }
    pw.printf("Min is %5.2f \nMax is %5.2f",minBal,maxBal);
}

int countAccounts(){/*counts number of accounts for given specific balance*/
    String rb=JOptionPane.showInputDialog("Amount to look for: ");
    float req_bal=Float.parseFloat(rb);
    int numOfAccounts=0;
    for(BankAccount cu:customer_list){
        if((Math.abs(req_bal-cu.get_Balance()))<=EPSILON) numOfAccounts++;
    }
}

```

```

        return numOfAccounts;
    }
}
class additional_functionality extends BankAccount{ // more facilities
    void addAccount(){
        name=JOptionPane.showInputDialog("Your Name");
        String acc_s=JOptionPane.showInputDialog(null,"hey! "+name+ "\nPlease
Enter your Account Number");
        try{
            accountNumber=Long.parseLong(acc_s);
        } catch(NumberFormatException e){ pw.println(e.getMessage());}
        BankAccount customer = new BankAccount(name,accountNumber,0);
        customer_list.add(customer);
        update(name,accountNumber,balance);
        display("Customer Added!");
    }
    float deposit(float x){
        accountNumber=input_account();
        if(findAccount(accountNumber).equals("")) display("");
        else{
            for(int index=0;index<customer_list.size();index++)
                if(customer_list.get(index).get_Account_no()==accountNumber){
                    float curr_bal=customer_list.get(index).get_Balance();
//this thing can be compacted.
                    curr_bal=curr_bal+x;
                    customer_list.get(index).setBalance(curr_bal);
                    pw.printf("Updated balance:
%5.2f%n",customer_list.get(index).get_Balance());
                    display("Deposited!");
                }
        }
        return x;
    }
    float withdraw(float x){
        accountNumber=input_account();
        if(findAccount(accountNumber).equals("")) display("");
        else{
            for(int index=0;index<customer_list.size();index++)
                if(customer_list.get(index).get_Account_no()==accountNumber){
                    float curr_bal=customer_list.get(index).get_Balance();
//this thing can be compacted.
                    if((Math.abs(curr_bal-0)<=EPSILON) || x>curr_bal)
display("Insufficient Funds");
                    else{
                        curr_bal=curr_bal-x;
                        customer_list.get(index).setBalance(curr_bal);
                        pw.printf("Updated balance:
%5.2f%n",customer_list.get(index).get_Balance());
                        display("Withdrawn!");
                    }
                }
        }
        return x;
    }
    String findAccount(long search_accn){
        String search_result="";
        for(BankAccount cust: customer_list){
            if(cust.get_Account_no()==search_accn){

```

```

        search_result="A/C#"+cust.get_Account_no()+"\nName:
"+cust.getName()+"\nBalance: "+cust.get_Balance();
        break;}
    }
    return search_result;
}
void update(String name,long acc, float bal){
    try{
        BufferedWriter writer=new BufferedWriter(new
FileWriter("bankfile.txt",true));//'true' appends data
        writer.write(name+";"+accountNumber+";"+balance+"\r\n");
        writer.close();
    }catch(IOException e){pw.println(e.getMessage());}
}
void persistence(){
    try{
        BufferedWriter writer=new BufferedWriter(new
FileWriter("bankfile.txt"));//'renews file
        for(BankAccount cu:customer_list){

writer.write(cu.getName()+";"+cu.get_Account_no()+";"+cu.get_Balance()+"\r\n");
        }writer.close();
    }catch(IOException e){pw.println(e.getMessage());}
}
void display(String disp_data){
    if(disp_data.equals(""))
        JOptionPane.showMessageDialog(null,"Not
Found","Error",JOptionPane.ERROR_MESSAGE);
    else

        JOptionPane.showMessageDialog(null,disp_data,"Success",JOptionPane.INFORMATION
_MESSAGE);
}
long input_account(){
    String acc_s=JOptionPane.showInputDialog(null,"Input Account Number:");
    try{
        accountNumber=Long.parseLong(acc_s);
    } catch(NumberFormatException e){ pw.println(e.getMessage());}
    return accountNumber;
}
}
class banker{
    public static void main(String args[]){
        additional_functionality af=new additional_functionality();
        BankAccount ba=new BankAccount();
        ba.pw.println("This is.. \n \t\tTHE BANK\nplease wait...");
        Integer choice;
        do{
            String ch=JOptionPane.showInputDialog(ba.choiceList);
            choice=Integer.parseInt(ch);
            switch(choice){
                case 0:af.persistence();
                    System.exit(0);
                    break;
                case 1: af.addAccount();
                    break;
                case 2: String amount=JOptionPane.showInputDialog(null,"Withdrwal Amount:
");

```

```

        af.withdraw(Float.parseFloat(amount));
        break;
    case 3: amount=JOptionPane.showInputDialog(null,"Deposit Amount: ");
        af.deposit(Float.parseFloat(amount));
        break;
    case 4: af.display(af.findAccount(af.input_account()));
        break;
    case 5: ch=JOptionPane.showInputDialog(ba.moreOptions);
        choice=Integer.parseInt(ch);
        switch(choice){
            case 6:ba.minMaxAmount();
            break;
            case 7:af.display(ba.countAccounts()+"account(s) have the
required balance.");
            break;
            case 8: af.display("Total Balance:
Rs."+ba.getTotalbalance());
            break;
        }
        break;
    default: JOptionPane.showMessageDialog(null,"Wrong
Choice!", "ERR",JOptionPane.ERROR_MESSAGE);
}
}while(choice!=0);
System.exit(0);
}
}

```

---

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.

---

#### **Stack.java**

```

public abstract class Stack
{
    public abstract void push(int amount);
    public abstract void pop();
    public abstract void display();
}

```

#### **DynamicStack.java**

```

import java.util.ArrayList;
public class DynamicStack extends Stack
{
    ArrayList<Integer> st=new ArrayList<Integer>();

    public void push(int el)
    {
        st.add(el);
    }

    public void pop()

```

```

    {
        st.remove((st.size()-1);
    }

    public void display()
    {   System.out.println("The elements in the stack are : ");
        for(int i=0;i<st.size();i++)
            { System.out.print(st.get(i) + " ");}
    }
}

```

### **StaticStack.java**

```

public class StaticStack extends Stack
{
    private int a[]=new int[10];

    public void push(int el)
    { for(int i=0;i<10;i++)
        { if(a[i]==0)
            { a[i]=el;
              break;
            }
          else if(a[i]!=0 && i==9)
            { System.out.println("Stack Full!Cannot Push more elements");}
        }
    }

    public void pop()
    { for(int i=0;i<10;i++)
        { if(a[i]==0 && i>0)
            { a[i-1]=0;
              break;
            }
          else if(a[i]==0 && i==0)
            { System.out.println("Cannot Pop.The stack is empty");
              break;
            }
        }
    }

    public void display()
    {   System.out.println("The Stack elements are:");
        for(int i=0;i<10;i++)
            { if(a[i]!=0)
                {System.out.print(a[i] + " ");}
            }
    }
}

```

### **StackStore.java**

```

import java.util.Scanner;
public class StackStore
{
    public static void main(String args[])
    { String ch3;
      int n;
      StaticStack ss=new StaticStack();
      DynamicStack ds=new DynamicStack();
    }
}

```

```

Scanner in=new Scanner(System.in);
System.out.println("*****Main Menu*****");
System.out.println("1.Static Stack");
System.out.println("2.Dynamic Stack");
System.out.println("Enter your choice :");
int ch1=in.nextInt();
do
{
System.out.println("*****Sub Menu*****");
System.out.println("1.Push");
System.out.println("2.Pop");
System.out.println("3.Display");
System.out.println("Enter your choice :");
int ch2=in.nextInt();
switch(ch2)
{
case 1: System.out.print("Enter element to push :");
int ele=in.nextInt();
if(ch1==1)
{ ss.push(ele);}
else
{ //System.out.print("How many elements do you want to p: ");
ds.push(ele);
}
break;
case 2: if(ch1==1)
{ ss.pop();}
else
{ //System.out.print("How many elements do you want to pop:
");
ds.pop();}
break;
case 3:if(ch1==1)
{ ss.display();}
else
{ ds.display();}
break;
}
System.out.print("\nContinue(y/n)? ");
ch3=in.next();
}
while(!ch3.equals("n"));

}
}

```

## //JDBC

- 
1. Create Student and Results Database and perform the following using JDBC programs
    - a. Find total number of students
    - b. Print average marks for each subject input by user.
    - c. Find the name of student getting 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.
-

```

/*
MySQL in WAMP
uses mysql-connector-5.1.44
@Written in:: NetBeans8.2
@First Complete:: 09/11/17
@author :: Kumar Prateek Viraj
*/
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.Scanner;
class student{
    Connection conn = null;
    Statement stmt = null;
    student(){
        System.out.println("\n\n***** MySQL JDBC Connection Testing *****");
        try
        {
            Class.forName ("com.mysql.jdbc.Driver").newInstance ();
            System.out.println("ok till line 13");
            String userName = "root";
            String password = "toor";
            String url =
"jdbc:mysql://localhost:3306/students?autoReconnect=true&useSSL=false";
            conn = DriverManager.getConnection (url, userName, password);
            System.out.println ("\nDatabase Connection Established...");
        }
        catch (Exception ex)
        {
            System.err.println ("Cannot connect to database server");
        }
    }
    void setupDatabase(){
        try{
            stmt = conn.createStatement();
            String createDb="CREATE DATABASE IF NOT EXISTS students";
            stmt.executeUpdate(createDb);
            System.out.println("\nDatabase setup done.");
            String createTable="CREATE TABLE IF NOT EXISTS students.result ("
                + "roll INT(3) NOT NULL,"
                + "name VARCHAR(255),"
                + "sub1 INT(3),"
                + "sub2 INT(3),"
                + "sub3 INT (3),"
                + "total INT(3),"
                + "PRIMARY KEY(roll)"
                + ")";
            stmt.executeUpdate(createTable);
            System.out.println("\nTable setup done.");
        }
        catch(SQLException s){
            System.out.println("\nProblem in database setup."+s);
        }
    }
    void inputValues(){

```



```

        int total;
// System.out.println("\nEnter Roll: ");
Scanner reader=new Scanner(System.in);
//int roll=reader.nextInt();
System.out.println("\nEnter Name: ");
String name=reader.next();
System.out.println("Marks in Subject 1:");
int sub1=reader.nextInt();
System.out.println("Marks in Subject 2: ");
int sub2=reader.nextInt();
System.out.println("Marks in Subject 3: ");
int sub3=reader.nextInt();
total=sub1+sub2+sub3;
insertValues(name,sub1,sub2,sub3,total);
}
void insertValues(String name,int sub1,int sub2,int sub3,int total){
    int rowCount=0;
    try {
        String insertQuery= "INSERT INTO `result` "
            + "(`name`, `sub1`, `sub2`, `sub3`, `total`)"
            + " VALUES(?,?,?, ?,?)";
        PreparedStatement pst=conn.prepareStatement(insertQuery);
        pst.setString(1,name);
        pst.setInt(2, sub1);
        pst.setInt(3, sub2);
        pst.setInt(4, sub3);
        pst.setInt(5,total);
        pst.executeUpdate();
        pst.close();
        System.out.println("Data Inserted!");
    } catch (SQLException ex) {
        System.out.println("Some error"+ex);
    }
}
int totalStudents(){
    int count=0;
    try{
        stmt=conn.createStatement();
        String query="SELECT COUNT(*) FROM students.result";
        ResultSet res =stmt.executeQuery(query);
        while (res.next()){
            count = res.getInt(1);
        }
    }
    catch(Exception e){
        System.out.println("Error!");
    }
    return count;
}
void AverageInSubject(){
    try{
        for(int i=1;i<4;i++){
            String subAverage="SELECT AVG(sub"+i+") FROM result";
            stmt=conn.createStatement();
            ResultSet subav=stmt.executeQuery(subAverage);
            subav.next();
            System.out.println("Average marks in subj"+i+":: "+subav.getInt(1));
        }
    }
}

```

```

        }catch(SQLException e){
        e.printStackTrace();
        }
    }
    String findTopper(){
        String topper=null;
        int roll=0;
        String query="SELECT roll,name FROM result WHERE total=("
            + "SELECT MAX(total) FROM result)";

        try{
            stmt=conn.createStatement();
            ResultSet rs=stmt.executeQuery(query);
            while(rs.next()){
                roll=rs.getInt(1);
                topper=rs.getString(2);
            } System.out.println("Roll is: "+roll);
        }catch(SQLException e){
            System.out.println("Error!");
        }
        return topper;
    }
    void secondThird(){
        int topperRoll=0;
        String name=null;
        String queryForSecond="SELECT roll,name "
            + "FROM result ORDER BY total DESC LIMIT 1,1";
        String queryForThird="SELECT roll,name "
            + "FROM result ORDER BY total DESC LIMIT 2,1";

        try{
            stmt=conn.createStatement();
            ResultSet rs1 = stmt.executeQuery(queryForSecond);
            while(rs1.next()){
                topperRoll=rs1.getInt(1);
                name=rs1.getString(2);
                System.out.println("Second Topper:: Name : "+name+" Roll:
"+topperRoll);
            }
            ResultSet rs2 = stmt.executeQuery(queryForThird);
            while(rs2.next()){
                topperRoll=rs2.getInt(1);
                name=rs2.getString(2);
                System.out.println("\nThird Topper:: Name : "+name+" Roll:
"+topperRoll);
            }
        }catch(SQLException ex){
            System.out.println("Some error "+ex);
        }
    }

    void subjectToppers(int subjNum){
        String topper=null;
        String query="SELECT roll,name FROM result WHERE sub"+subjNum+"=("
            + "SELECT MAX(sub"+subjNum+") FROM result)";

        try{
            stmt=conn.createStatement();
            ResultSet rs=stmt.executeQuery(query);
            while(rs.next()){
                System.out.println("Subj"+subjNum+" Topper::"

```

```

        + " Name"+rs.getString(2)+" Roll no: "+rs.getInt(1));
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}
void averageMarks() {
    try {
        String subAverage = "SELECT AVG(total) FROM result";
        stmt = conn.createStatement();
        ResultSet subav = stmt.executeQuery(subAverage);
        subav.next();
        System.out.println("Average Total marks:: "+subav.getInt(1));
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

void closeConnection() {
    if (conn != null)
    {
        try
        {
            System.out.println("\n***** Let terminate the Connection *****");
            conn.close();
            System.out.println("\nDatabase connection terminated...");
        }
        catch (Exception ex)
        {
            System.out.println("Error in connection termination!");
        }
    }
}
}

public class testjdbc
{
    public static void main (String[] args)
    {
        int choice;
        student s = new student();
        s.setupDatabase();
        do {
            System.out.println("Hey user! Whatd'y wanna do?");
            System.out.println("\n1.Insert Values \n2.Total Number of Students"
                + "\n3.Average marks of Subject \n4.Who's the topper?"
                + "\n5.Who are First, Second & Third\n6.Subjectwise toppers"
                + "\n7.Average Marks\n8.Who is Second Topper?\n\n0.Close&Exit");
            Scanner reader = new Scanner(System.in);
            choice = reader.nextInt();
            switch (choice) {
                case 1: s.inputValues();
                    break;
                case 2: System.out.println("\nTotal Students: "+s.totalStudents());
                    break;
                case 3: s.AverageInSubject();
                    break;
                case 4: System.out.println("Topper is: "+s.findTopper());
                    break;
            }
        } while (choice != 0);
    }
}

```

```

        case 5: System.out.println("First Topper:: "+s.findTopper());
                s.secondThird();
        break;
        case 6: s.subjectToppers(1);
                s.subjectToppers(2);
                s.subjectToppers(3);
        break;
        case 7:s.averageMarks();
        break;
        case 8: s.secondThird();
        break;
        case 0: s.closeConnection();
                System.exit(0);
    }
    }while(choice!=0);
    s.closeConnection();
}
}

```

## //JAVASCRIPT

### Index.html for java script

---

```

<!DOCTYPE html>
<html>
<head>
    <title>Int-Tech</title>
    <script type="text/javascript" src="answer_script.js"></script>
    <script type="text/javascript">
        function displayList(){
            for(i=6;i>0;i--){
                var table = document.getElementById("quesTable");
                var row = table.insertRow(0);
                row.innerHTML ="<button id='"+i+"'
onclick='viewform(this.id) ' >Question"+i+"</button>";
            }
        }
        function viewform(toShow){
            for(i=6;i>0;i--){
                toHide="f"+i;
                document.getElementById(toHide).style.display='none';
            }
            toShow="f"+toShow;
            document.getElementById(toShow).style.display='block';
        }
    </script>
</head>
<body onload="displayList()">
    <div class="container">
        <header>
            <h1>Internet Technologies</h1>
            <h2>--Javascript--</h2>
        </header>
        <nav>
            <table id="quesTable">

```

```

        </table>
    </nav>
    <article>
        <form id="f1">
            <input type="number" name="roll" onblur="check_roll(this.value);"
placeholder="Roll Number"><br />
            <input type="text" name="name" onblur="check_name(this.value);"
placeholder="Your Name"><br />
            <input type="date" id="dob" placeholder="Date Of Birth"><br />
            <input type="password" name="pass" placeholder="Password"
onblur="staticPasswordProtection(this.value)"><br />
            <input type="password" name="cpass" placeholder="Confirm
Password" onblur="confirmPassword(this.value)"><br />
            <input type="button" name="submit" value="Submit"
onclick="validate();"
            printDOB(document.getElementById('dob').value)">
            <p/> <span id="birthdayBash"></span>
        </form>
        <form id="f2" style="display: none;">
            <h2>This has been embedded in Question 1</h2>
        </form>
        <form id="f3" style="display: none;">
            <h2>Bubble Sort</h2>
            <input type="number" id="bubbleInput" placeholder="Input Unsorted
data">
            <input type="button" value='Insert'
onclick="storeIt(document.getElementById('bubbleInput').value)">
            <p /><span id="unsorted">Unsorted List: </span><p />
            <input type="button" value="Sort Now"
onclick="bubbleSort()"><br/>
            <span id="sorted"> Sorted List: </span>
        </form>
        <form id="f4" style="display: none;">
            <h2>Implementing Stack</h2>
            <input type="number" id="stack" placeholder="Your Input">
            <input type="button" name="push" value="Push"
onclick="pushIt(document.getElementById('stack').value);"><br />
            <input type="button" name="pop" value="Pop" onclick="popIt()">
            <table id="stackTable"></table>
        </form>
        <form id="f5" style="display: none;">
            <input type="button" name="timer" value="Start Timer"
            onclick="setTimeout(changeColor,3000)">
            <div id="sample">
                <h2>Color of this text will be changed.<br/> Wait 3
seconds..</h2>
            </div><br/>
            <input type="button" name="move" value="Start Moving"
            onclick="startMoving();"
            document.getElementById('info').style.display='none'">
            <div id="mydiv">
                
            </div><p>
            <pre id="info">
                The image used has been recreated by the author himself.
<Copyleft>
                *CSS has been used to animate elements.</pre>
        </form>

```

```

<form id="f6" style=" display: none;">
  <fieldset>
    <h2>The user registration is same as in Question 1.</h2><br/>
    <h3>Here, only the feedback form will be validated.</h3>
    <table align="center">
      <tr>
        <td>I am a </td>
        <td><input type="radio" name="person"
value="t">Teacher</td>
        <td><input type="radio" name="person"
value="s">Student</td>
      </tr>
      <tr>
        <td>I can code in:</td>
        <td><input type="checkbox" name="lang">
Assembly<br/></td>
        <td><input type="checkbox" name="lang">
Scala<br/></td>
        <td><input type="checkbox" name="lang">
Brainfuck<br/></td>
        <td><input type="checkbox" name="lang">
Kotlin<br/></td>
      </tr>
      <tr>
        <td>I am coding for: </td>
        <td><select>
          <option value="min">0-2 years</option>
          <option value="min">2-5 years</option>
          <option value="min">>5 years</option>
        </select></td>
      </tr>
      <tr>
        <td>My feedback for the author of this code</td>
        <td>
          <textarea name="feedback" rows="4"
cols="40"></textarea>
        </td>
      </tr>
      <tr>
        <td>
          <input type="button" value="submit"
onclick="validation()">
        </td>
      </tr>
    </table>
  </fieldset>
</form>
</article>
<section id="code-section">
</section>
<footer>
  Code by: Kumar Prateek Viraj
</footer>

</div>
</body>
<style type="text/css">
  #myimage {

```

```

        position: absolute;
    }
    #mydiv {
width: 400px;
position: relative;
}
nav{
    float: left;
    max-width: 160px;
    margin-top: 50px;
    padding: 1em;
    font-size: 25px;
}
nav button{
    background-color: grey;
    width: 140px;
    padding-top: 18px;
}
nav button:hover{
    background-color: green;
}
article {
    margin-left: 170px;
    border-left: 1px solid gray;
    padding: 0.2em;
    overflow: hidden;
    min-height: 420px;
    text-align: center;
}
article #f1 input{
    width: 180px;
    padding: 8px;
    margin-bottom: 12px;
    border-radius: 8px;
    border-style: groove;
}
article #f1 input:hover{
    width: 250px;
    animation: ease-in-out;
    transition-duration: 0.3s;
    border-color: #667766;
}
#f4 table{
    border-style: double;
    border-top: none;
}

#f6 table td{
    border-bottom: 1px solid #ddd;
    padding: 20px;
}
header, footer{
    text-align: center;
    background-color: black;
    color: white;
}
}
</style>
</html>

```

## QUESTIONS:

- 
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
  2. Implement a Static Password Protection.
  3. Write a java script to sort an array using bubble sort. Take the number of elements and array from user.
  4. Write a JavaScript to implement stack methods (push and pop).
  5. Write a JavaScript
    - a. to change the color of text using setTimeout()
    - b. to move an image across screen using setInterval()
  6. Implement the question no. 1 of hands on exercises of chapter 10 (page 190).
- 

## Answer\_script.js

```

var stackArray=[]; //global array for stack wala question
var i=0; //global counter also the stack pointer
var bubbleArray=[]; //array for bubble Sort

function validate(){
    var form1=document.getElementById('f1');
    for(i=0;i<5;i++){ //iterate through all form1 elements
        if(form1.elements[i].value==""){
            alert("One or more fields are empty");
            break;
        }
    }
}

function check_roll(roll){
    var len=roll.toString().length //convert to string and find length
    var pat=/\./;
    if(len!=7){
        alert("Roll number must be 7 digits.")
    }
    else if(pat.test(roll)) //must not contain decimal point
        alert("Cannot Conatin dot ");
}

function check_name(name){
    if(name.replace(/\s+/g, '').length<4) //remove white spaces and check length
        alert("Minimum 4 characters");
    pat=/[^a-zA-Z]/; //find anything except a-zA-Z
    if (result=name.match(pat))
        alert("Found unacceptable character: "+result);
}

function staticPasswordProtection(p){
    window.pass=p;//made it global to access in next function
    if(pass.length<6)
        alert("Atleast 6 characters required");
    var pat=/((^[0-9]+[a-z]+)|(^[a-z]+[0-9]+))+[0-9a-z]+$/i; //allow only
    alphanumeric
    if(!pass.match(pat))

```



```

        alert("Must be Alphanumeric")
    }
    function confirmPassword(cpass){
        if(cpass!=pass)
            alert("Passwords do not match");
    }
    function printDOB(mydob){
        var days= new Array(7);
        days[0] = "Sunday";days[1] = "Monday";days[2] = "Tuesday";days[3] =
"Wednesday";
        days[4] = "Thursday";days[5] = "Friday";days[6] = "Saturday";
        var months=new Array(12);
        months[0]="January";months[1]="February";months[2]="March";months[3]="April";m
onths[4]="May";months[5]="June";
        months[6]="July";months[7]="August";months[8]="September";months[9]="October";
months[10]="November";months[11]="December";
        var parsedDate = Date.parse(mydob);
        var d = new Date(parsedDate);
        var dayInWords=days[d.getDay()];
        var monthInWords=months[d.getMonth()];
        var birthday=dayInWords+", "+monthInWords+" "+d.getDate()+"",
"+d.getFullYear();
        if(birthday=="Thursday, July 3, 1997") alert("Amazing! It's Kumar Prateek's
Birthday"); //easter-egg
        document.getElementById('birthdayBash').innerHTML="So, Your Birthday is on
"+birthday;
    }
    function storeIt(element){
        bubbleArray[i]=element;
        var unsortedList=document.getElementById('unsorted');
        unsortedList.innerHTML+=bubbleArray[i++]+" "; //append data to unsorted span
    }
    function bubbleSort(){
        for(i=0;i<bubbleArray.length;i++){
            for(j=0; j<bubbleArray.length;j++){
                if(parseInt(bubbleArray[i])<parseInt(bubbleArray[j])){//is parseInt()
really necessary? -IDTS.
                    var temp=bubbleArray[i];
                    bubbleArray[i]=bubbleArray[j];
                    bubbleArray[j]=temp;
                }
            }
        }
        for(i=0;i<bubbleArray.length;i++)
            document.getElementById('sorted').innerHTML+=bubbleArray[i]+" ";
    }
    function pushIt(toPush){
        if (toPush=="")
            alert("Nothing to push");
        else{
            stackArray[i]=toPush;
            window.stack=document.getElementById('stackTable');
            var row=stack.insertRow(0);
            row.innerHTML=stackArray[i++]; //append to stackTable
        }
    }
    function popIt(){
        if(i<1)

```

```

        alert("Stack is already empty");
    else{
        alert("Element popped: "+stackArray[--i]);
        stack.deleteRow(0);
    }
}

function changeColor(){
    var sample=document.getElementById('sample');
    sample.style.color="red";
    sample.innerHTML+="Yay!! Code-Red";
}

function startMoving(){
    var elem=document.getElementById('myimage');
    var pos = 40;
    var id = setInterval(moveRight, 10);
    function moveRight() {
        if (pos ==450 ) {
            pos=0;
            moveRight();
        }
        else{
            pos++;
            elem.style.left = pos + 'px';
        }
    }
}

function validation(){
    var form=document.getElementById('f6');
    var radio=document.getElementsByName('person');
    var person;
    var flag=0;
    if(radio[0].checked == true)
        person="Teacher";
    else if(radio[1].checked == true)
        person="Student";
    else alert("Teacher/Student? What are you?");
    var lang=document.getElementsByName('lang');
    for(i=0;i<4;i++){
        if(lang[i].checked==true){
            flag=1;
            break;
        }
    }
    if(flag==0) alert("You must select one language!");
    var feedback = document.getElementsByName('feedback');
    var values = feedback.value;
    if(values == null)
        alert("Must give feedback!");
    else alert("Submitted!")
}

```

---

## JSP

1. Display the pattern:

```

1
1 2

```

1 2 3

Take `¥n` in a textbox from user. Display this pattern using

„h Scriptlets

„h `<c:forEach>` loop

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)

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

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

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.

6. Create your custom library which contains two tags: `<hello>`, `<choco>`.

Usage of the tags:

„h `<hello name=¥Ajay>`: Output should be Hello Ajay. It contains a mandatory attribute `¥name` which can accept Dynamic value.

„h `<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.

7. Create a custom tag `¥substring` with 3 mandatory attributes `¥input`, `¥start`, `¥end` which will do substring operation on given input

8. Create a custom tag `¥reverse` with a mandatory attribute `¥input` to reverse a string.

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

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`)

## Index.jsp

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>JSP</title>
```

```
    <font size="5" color="red">
```

```
        <marquee width='850'> The classpaths must be set for executing custom tags!
```

```
        The folder directory must be intact and java files must be compiled duly.
```

```

        </marquee></font>
</head>
<body>
<div id="container">
    <header>
        <h1>Internet Technologies</h1>
        <h3>Javaserver Pages</h3>
    </header>
    <nav>
        <a href="q1.jsp" target="output"><pre>1. Display the pattern</pre></a>
        <a href="main.html" target="output">2. Caluculator</a> <br/>
        <a href="q3.jsp" target="output">3. Validation</a> <br/>
        <a href="q4.jsp" target="output">4. Greeting</a> <br/>
        <a href="q5.jsp" target="output">5. Even - Odd</a> <br/>
        <a href="q61.jsp" target="output">6. Custom Tags</a> <br/>
        <a href="q7.jsp" target="output">7. Custom Substring Tag</a> <br/>
        <a href="q9.jsp" target="output">7. Custom Date Tag</a> <br/>
        <a href="q10.jsp" target="output">7. Shopping</a> <br/>
    </nav>
    <article>
        <iframe name="output" src="myimage.png"></iframe>
    </article>
    <footer>
        Code by : Kumar Prateek Viraj
    </footer>
</div>
</body>
<style type="text/css">
    body{
        text-align: center;
    }
    nav{
        float: left;
        max-width: 170px;
        margin-top: 50px;
        padding: 1em;
        font-size: 25px;
    }
    a{
        font-size: 20px;
        text-decoration: none;
    }
    article{
        min-height: 490px;
        margin-left: 380px;
        border-left: 1px solid gray;
        padding: 0.2em;
        overflow: hidden;
    }
    article iframe{
        float: right;
        min-width: 100%;
        min-height: 450px;
        border: none;
    }
    header, footer{
        color: #fff;

```

```

        background-color: black;
    }
</style>
</html>

```

---

Q1:

```

<%@ page language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
    <title>Number Printer</title>
</head>
<body>
    <form method="post" action="q1.jsp">
        Enter the number of levels <input type="text" name="no" min="1">
        <input type="submit" value="Submit">
        <input type="hidden" name="submitted" value="true">
        <c:if test="${param.submitted}">
            <br><br><br>
            Using scriptlets :
                <% int num=Integer.parseInt(request.getParameter("no"));
                    for(int i=1;i<=num;i++)
                    {
                        %>
                        <br>
                        <%
                            for(int j=1;j<=i;j++)
                            { out.print(j + " "); }
                        %>
                    }
                %>
            <br><br><br><br>
            Using <tc:forEach> loop :
            <c:forEach begin="1" end="${param.no}" step="1" var="current">
                <br>
                <c:forEach begin="1" end="${current}" step="1" var="print">
                    <c:out value="${print}"/>
                </c:forEach>
            </c:forEach>
        </c:if>
    </form>
</body>
</html>

```

---

Main.html

```

<html>
<head>
    <title>Calculator</title>
</head>
<body>
    <form action="operate.jsp" method="post">
    <p>First Number <input type="text" name="first"></p>
    <p>Second Number <input type="text" name="second"></p>
    <input type="radio" name="operation" value="a">Addition
    <input type="radio" name="operation" value="s">Subtraction
    <input type="radio" name="operation" value="m">Multiplication
    <br>

```

```

    <input type="submit" value="Calculate">
</form>
</body>
</html>

```

---

### operate.jsp

```

<%@ page language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
    <title>Your Answer</title>
</head>
<body>
    Using Expression language :
    <br><br>
    Your Answer is
    <c:if test="${param.operation=='a'}">
        ${param.first + param.second}
    </c:if>
    <c:if test="${param.operation=='s'}">
        ${param.first - param.second}
    </c:if>
    <c:if test="${param.operation=='m'}">
        ${param.first * param.second}
    </c:if>
    <br><br><br><br><br>
    Using scriptlets :
    <br>
    Your Answer is
    <% String op=request.getParameter("operation");
        int num1=Integer.parseInt(request.getParameter("first"));
        int num2=Integer.parseInt(request.getParameter("second"));
        if(op.equals("a"))
        { out.print(num1+num2); }
        else if(op.equals("s"))
        { out.print(num1-num2); }
        else if(op.equals("m"))
        { out.print(num1*num2); }
    %>
</body></html>

```

### Q3.jsp

```

<%@ 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">

        <form action="q3.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:otherwise>
                    <input type="radio" name="gender" value="m" checked>Male<br>
                    <input type="radio" name="gender" value="f">Female
                </c:otherwise>
            </c:choose>
        </td>
    </tr>

```

```

        </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:when>
        <c:when test="\${current == 'p'}">
            <c:set var="pastaSelected" value="true" />
        </c:when>
        <c:when test="\${current == 'c'}">
            <c:set var="chineseSelected" value="true" />
        </c:when>
        <c:otherwise>
            <c:set var="invalidSelection" value="true" />
        </c:otherwise>
    </c:choose>
</c:forEach>
<c:if test="\${invalidSelection}">
    <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>

```

---

#### Q4.jsp

```

<%@ page import="java.util.*" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>
<head>
    <title>Greetings</title>
</head>
<body>
<form action="q4.jsp" method="post">
Enter your name <input type="text" name="uname" >
<input type="hidden" name="submitted" value="true">
    <input type="submit" value="Submit">

<c:if test="${param.submitted}">
    <% java.util.Date clock = new java.util.Date( ); %>
    <% if (clock.getHours( ) < 12) { %>
        <h2>Good morning!</h2>
    <% } else if (clock.getHours( ) < 17) { %>
        <h2>Good day!</h2>
    <% } else { %>
        <h2>Good evening!</h2>
    <% } %>
    <c:out value="${param.uname}"/>
</c:if>
</form></html>

```

#### Q5.jsp

```

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

<!DOCTYPE html>
<html>
<head>
    <title></title>
</head>
<body>
<form action="q5.jsp">
    <input type="text" name="word" placeholder="Enter word"></br>
    <input type="radio" name="place" value="even">Even
    <br><input type="radio" name="place" value="odd"> Odd
    <br><input type="Submit" name="submit" value="Play!">

    <c:if test="${param.word!=''}">
    <c:set var="str" value ="${param.word}"/>
    <c:if test="${param.place==even}">
        <c:forEach var="i" begin="0" end="${fn:length(str)}" step="2">
            <c:set var="newstr" value="${fn:substring(str, i, i + 1)}" />
        </c:forEach>
    </c:if>
    <c:if test="${param.place==odd}">

```

```

        <c:forEach var="i" begin="1" end="${fn:length(str)}" step="2">
        <c:set var="newstr" value="${fn:substring(str, i, i + 1)}" />
        </c:forEach>
    </c:if>
</c:if>
</form>
</body>
</html>

```

---

### Q6.jsp

#### Customtag.tld

```

<taglib>
    <tlib-version>1.0</tlib-version>
    <jsp-version>2.0</jsp-version>
    <short-name>Example TLD with Body</short-name>

    <tag>
        <name>Hello</name>
        <tag-class>viraj.tags.HelloTag</tag-class>
        <body-content>scriptless</body-content>
    </tag>
</taglib>

```

#### Hellotag.java

```

package viraj.tags;

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

public class HelloTag extends SimpleTagSupport {
    StringWriter sw = new StringWriter();
    public void doTag()

    throws JspException, IOException {
        getJspBody().invoke(sw);
        getJspContext().getOut().println(sw.toString());
    }
}

```

### Q6.jsp

```

<%@ taglib prefix = "ex" uri = "WEB-INF/custom.tld"%>

<html>
    <head>
        <title>A sample custom tag</title>
    </head>

    <body>
        <ex:Hello/>
    </body>
</html>

```

---

#### .TLD FILE

```

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

```

```

<tlibversion>1.0</tlibversion>
<jspversion>1.1</jspversion>
<shortname>substr</shortname>
<info>Sample taglib for Substr operation</info>
<uri>https://viraj/jsp/taglib/substr</uri>
<tag>
    <name>substring</name>
    <tagclass>virajl.jsp.custom.taglib.substring</tagclass>
    <info>Substring function.</info>
    <attribute>
        <name>input</name>
        <required>true</required>
    </attribute>
    <attribute>
        <name>start</name>
        <required>true</required>
    </attribute>
    <attribute>
        <name>end</name>
        <required>true</required>
    </attribute>
</tag>
</taglib>

```

#### **.JAVA FILE**

```

package viraj.jsp.custom.taglib;

import java.io.IOException;

import javax.servlet.jsp.JspException;
import javax.servlet.jsp.JspWriter;
import javax.servlet.jsp.tagext.TagSupport;

public class substring extends TagSupport {
    private String input;
    private int start;
    private int end;

    @Override
    public int doStartTag() throws JspException {

        try {
            //Get the writer object for output.
            JspWriter out = pageContext.getOut();

            //Perform substr operation on string.
            out.println(input.substring(start, end));

        } catch (IOException e) {
            e.printStackTrace();
        }
        return SKIP_BODY;
    }
    public String getInput() {
        return input;
    }
    public void setInput(String input) {
        this.input = input;
    }

```

```

    }
    public int getStart() {
        return start;
    }
    public void setStart(int start) {
        this.start = start;
    }
    public int getEnd() {
        return end;
    }
    public void setEnd(int end) {
        this.end = end;
    }
}

```

### Q7.jsp

```

<%@taglib prefix="test" uri="/WEB-INF/substring.tld"%>
<html>
<head>
    <title>JSP Custom Taglib example: Substr function</title>
</head>
<body>
    SUBSTR(GOODMORNING, 1, 6) is
    <font color="blue">
        <test:substring input="GOODMORNING" start="1" end="6"/>
    </font>
</body>
</html>

```

---

### Q9.jsp

#### .TLD FILE

```

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

<taglib>
    <tlib-version>1.0</tlib-version>
    <jsp-version>2.0</jsp-version>
    <short-name>d</short-name>
    <uri>http://viraj/tlds/date</uri>

    <tag>
        <name>DateTag</name>
        <tag-class>viraj.DateTagHandler</tag-class>
        <body-content>empty</body-content>
    </tag>

</taglib>

```

#### .JAVA FILE

```

package com.viraj;

```

```

import java.util.Date;
import javax.servlet.jsp.tagext.*;
import javax.servlet.jsp.JspWriter;
import javax.servlet.jsp.JspException;

public class DateTagHandler extends SimpleTagSupport {

    public void doTag() throws JspException {

        JspWriter out=getJspContext().getOut();

        try {

            out.println(new Date());

        } catch (java.io.IOException ex) {
            throw new JspException(ex.getMessage());
        }
    }
}

```

### .Q9.jsp

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="d" uri="http://viraj/tlds/date" %>

<html>
    <head>
        <title>Custom tags</title>
    </head>
    <body>
        <h2>Date</h2>
        <b>Today's Date: </b> <d:DateTag />
    </body>
</html>

```

---

### Q10.jsp

```

<%@ page language="java" contentType="text/html" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions" %>

<html>
    <head>
        <title>Product Catalog</title>
    </head>
    <body bgcolor="white">
        <h1>Product Catalog</h1>
        <form action="q10.jsp">
            <input type="text" name="uname" placeholder="Your name here"><br/>
            <input type="number" name="age" placeholder="Age">
            <input type="hidden" name="submitted" value="true">
            <input type="submit" name="submit">
        </form>
        <c:if test="${param.submitted}">
            <h3>Hey! <c:out value="${param.uname}"/></h3>
        </c:if>
    </body>
</html>

```

```
</c:if>
```

Please select a book from our catalog to read more about it and decide if you like to purchase a copy:

```
<jsp:useBean id="catalog" scope="application"
  class="com.ora.jsp.beans.shopping.CatalogBean"
/>
```

```
<%--
```

Generate a list of all products with links to the product page.

```
--%>
```

```
<ul>
```

```
<c:forEach items="${catalog.productList}" var="product">
```

```
<c:url var="productURL" value="product.jsp">
```

```
<c:param name="id" value="${product.id}" />
```

```
</c:url>
```

```
<li>
```

```
<a href="${productURL}">${fn:escapeXml(product.name)}</a>
```

```
</c:forEach>
```

```
</ul>
```

```
<jsp:useBean
```

```
id="cart" scope="session"
```

```
class="com.ora.jsp.beans.shopping.CartBean"
```

```
>
```

```
<%-- Show the contents of the shopping cart, if any --%>
```

```
<c:if test="${!empty cart.productList}">
```

Your shopping cart contains the following items:

```
<p>
```

```
<table border=0>
```

```
<c:forEach items="${cart.productList}" var="product">
```

```
<tr>
```

```
<td>${fn:escapeXml(product.name)}</td>
```

```
<td>
```

```
<fmt:formatNumber value="${product.price}"
```

```
type="currency" />
```

```
</td>
```

```
</tr>
```

```
</c:forEach>
```

```
<tr><td colspan=2><hr></td></tr>
```

```
<tr>
```

```
<td><b>Total:</b></td>
```

```
<td>
```

```
<fmt:formatNumber value="${cart.total}"
```

```
type="currency" />
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</c:if>
```

```
</body>
```

```
</html>
```

### Addtocart.jsp

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

```

<jsp:useBean id="catalog" scope="application"
  class="com.ora.jsp.beans.shopping.CatalogBean"
/>

<!-- Get the specified ProductBean from the catalog -->
<c:set var="product" value="\${catalog.productsById[param.id]}" />

<jsp:useBean
  id="cart"
  scope="session"
  class="com.ora.jsp.beans.shopping.CartBean"
/>

<!-- Add the product to the cart -->
<c:set target="\${cart}" property="product" value="\${product}" />

<c:redirect url="q10.jsp" />

```

### Product.jsp

```

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

<html>
  <head>
    <title>Product Description</title>
  </head>
  <body bgcolor="white">

    <jsp:useBean id="catalog" scope="application"
      class="com.ora.jsp.beans.shopping.CatalogBean"
    />

    <!-- Get the specified ProductBean from the catalog -->
    <c:set var="product" value="\${catalog.productsById[param.id]}" />

    <h1>\${fn:escapeXml(product.name)}</h1>

    \${fn:escapeXml(product.descr)}

    <p>
    <c:url var="addtocartURL" value="addtocart.jsp">
      <c:param name="id" value="\${product.id}" />
    </c:url>

    <a href="\${addtocartURL}">
      Add this book to the shopping cart</a>

    </body>
</html>

```

---

The file is also available at my github account -  
[https://github.com/VirajPrateek/Int\\_Tech](https://github.com/VirajPrateek/Int_Tech)

EOF