ATSS's
Institute of Industrial and Computer Management and Research, Nigdi Pune

MCA Department

Academic Year: 2022-23

Practical Journal

on

IT11L- Java Programming (SEM-I)

Submitted By:

Roll no: 03
Name: Achal anilsingh Pardeshi
Seat no: 20291

Date: 15-03-2023

Course Outcomes:

Student will be able to
CO1: Demonstrate Collection framework (Apply)
CO2: Develop GUI using AWT and swing (Apply)
CO3: Develop Web application using JSP and Servlet, JDBC (Apply)
CO4: Apply Data Structure to solve problems using JavaScript (Apply)
CO5: Demonstrate the concepts of Core Java (Apply)

ATSS's
Institute of Industrial and Computer Management and Research, Nigdi Pune

MCA Department

INDEX

Students Name: Achal anilsingh Pardeshi **Roll No. 03**

Sr. No	Program Title	Course Outcome	Page No.	Teacher's Sign with Date	Remarks
1.	Print the pattern: 5 10 15 20 25 30 35 40 45 50 55 60 65	**CO5**			
2.	Design an interface AdvancedArithmetic which contains a method signature int divisor_sum(int n). You need to write a class called MyCalculator which implements the interface. divisor_sum(int n) function takes an integer as input and return the sum of all its divisors. Divisors of 6 are 1, 2, 3 and 6, so divisor_sum should return 12. (0<n<100)	**CO5**			
3.	Create an Interface 'Animals' with abstract method **'void sound()'** and default method **'void walk()'**. Implement abstract method in class 'Cat' & 'Dog'. Now create an object for each of the subclasses and call their respective methods and default method too.	**CO5**			
4.	Declare the integer array with 10 numbers. Generate 2 new arrays **Prime** and **NonPrime** with prime and non-prime numbers from main array.	**CO5**			
5.	Write an application to identify and move all 0's to the end of an array. Maintain the sequence of the other (non-zero) array elements.	**CO5**			
6.	Write an application which will throw OverwtProductException if Product weight is above 60kg. (Use User defined exception)	**CO5**			
7.	Given two arrays, 1,2,3,4,5 and 2,3,1,1,0,5,0,2,1 find which number is not present in the second array.	**CO5**			
8.	Write code to check whether a no is a power of two or not?	**CO5**			

9.	Write a code to display string in reverse order of words.	**CO5**			
10.	Write a code to accept a string and check if there are two same consecutive letters, delete one of them.	**CO5**			
11.	Write a threaded application to print in one text area 1,2,3,4.... and in other textarea 2,4,9,16	**CO2**			
12.	Write a code to create calculator application using AWT, which will calculate simple Arithmetic operations.	**CO2**			
13.	Write a Menu Driven Program for Blood Donor application for following task a. Insert blood donor details into database. b. Display blood group-wise details of donors c. Update the address of a specific donor. d. Delete the record of donors whose age is below 18.	**CO3**			
14.	Write a servlet to check username & password passed from html page. If it is “Scott” & “tiger”, display welcome message else show the same html page again. [With res.sendRedirect (“http://localhost:8080/login.html”)]	**CO3**			
15.	Write a program to draw a circle on panel and move the circle as mouse is moving.	**CO2**			
16.	Write a servlet to add a Cookie to clients machine that stores username, current date & time. Display the same.	**CO3**			
17.	Write java program to generate 10 terms of Fibonacci series using threads.	**CO2**			
18.	Create a menu driven program for Bank account(acc_no, Name, amt) (Hint: use vector) 1. Add 2. Search 3. Delete 4. Display	**CO1**			
19.	Write a program to store employee in TreeSet and make sure employees are stored in sorted order of their age.	**CO1**			
20.	Create the list of patients and display the names of patients starting with ‘A’	**CO1**			

1) Print the pattern:

```
          5
     10   15   20
25   30   35   40   45
     50   55   60
          65
```

Solution:

```
import java.util.Scanner;
public class Pattern{

   public static void main(String args[]){
      int  i, k,  j, count=1; //n=3,

      for (j = 1; j<= 3; j++)
      {
         for (i = 1; i<=3-j; i++){
            System.out.print("  ") ;
         }
         for (k = 1; k <= 2 * j - 1; k++){
            System.out.print(5*count+" ");
            count++;
         }
         System.out.println(" ");
      }
      for (j = 2; j>=1; j--)
      {
         for (i = 1; i <=3-j; i++){
            System.out.print("  ");
         }
         for (k = 1; k <= 2 * j - 1; k++){
            System.out.print(5*count+" ");
            count++;
         }
         System.out.println(" "); }
   }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
 javac Pattern.java } ; if ($?) { java Pattern }
    5
  10 15 20
25 30 35 40 45
  50 55 60
    65
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

2) . Design an interface AdvancedArithmetic which contains a method signature int divisor_sum(int n). You need to write a class called MyCalculator which implements the interface. divisor_sum(int n) function takes an integer as input and return the sum of all its divisors. Divisors of 6 are 1, 2, 3 and 6, so divisor_sum should return 12. (0<n<100)

Solution:

```
import java.util.*;
interface advancedArithmetic{
        int divisorSum(int n);
}
class MyCalculator implements advancedArithmetic{
        public int divisorSum(int n){
          int sum = 0;
          for(int i = 1; i<=n; i++){
                  if(n%i==0){
                          sum+=i;
                  }
          }
          return sum;
 }
}
public class MyCalsiDemo{
        public static void main(String[] args){
       Scanner sc = new Scanner(System.in);
       int n = sc.nextInt();
    MyCalculator sum = new MyCalculator();
  System.out.println("The sum is: "+sum.divisorSum(n));
 }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Downloads\" ; if ($?) { javac Ex.java } ; if ($?)
 { java Ex }
The sum is: 12
PS C:\Users\Pardeshi\Downloads>
```

3) Create an Interface 'Animals' with abstract method 'void sound()' and default method ‘void walk()’. Implement abstract method in class 'Cat' & 'Dog'. Now create an object for each of the subclasses and call their respective methods and default method too

Solution:

```
interface Animals{
	abstract void sound();
	void walk();
}
class Cat implements Animals{
	 public void sound(){
		System.out.println("Meow");
	}

	public void walk(){
		System.out.println("Walking");
	}
}
class Dog implements Animals{
	public void sound(){
		System.out.println("Bhuu Bhuu");
	}

	public void walk(){
		System.out.println("Running");
	}
}
class AbsMethod{
	public static void main(String args[]){
		Cat c = new Cat();
		Dog d = new Dog();

		c.sound();
		c.walk();
		d.sound();
		d.walk();
	}
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
 javac AbsMethod.java } ; if ($?) { java AbsMethod }
Meow
Walking
Bhuu Bhuu
Running
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

4) Declare the integer array with 10 numbers. Generate 2 new arrays Prime and NonPrime with prime and non-prime numbers from main array.

Solution:

```
import java.util.ArrayList;
public class PrimeNonP{
public static void main(String[] args) {
 int nums[] = {4,50,11,6,7,79,8,4,3,7};
 ArrayList<Integer> prime = new ArrayList<Integer>();
 ArrayList<Integer> notPrime = new ArrayList<Integer>();
 for(int n:nums){
 if(prime(n)){
 prime.add(n);
 }else{
 notPrime.add(n);
 }
 }
 System.out.println("Prime numbers from array are: "+prime);
 System.out.println("Non Prime numbers from array are: "+notPrime);
 }
static boolean prime(int n){
 if(n==1) return false;
 for(int i = 2; i<n; i++){
 if(n%i==0){
 return false;
 }
 }
 return true;
 }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
  javac PrimeNonP.java } ; if ($?) { java PrimeNonP }
Prime numbers from array are: [11, 7, 79, 3, 7]
Non Prime numbers from array are: [4, 50, 6, 8, 4]
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

5). Write an application to identify and move all 0's to the end of an array. Maintain the sequence of the other (non-zero) array elements.

Solution:

```
public class MoveZeros{
        public static void main(String args[]){
    int[] arr = {1,0,4,0,3,0,3,5,0,2,5,3,0};
    int n = arr.length;
    int j=0;
      for (int i = 0; i < n; i++){
        if (arr[i] != 0) {
            //   Swap - A[j] , A[i]
            swap(arr, j, i);
            j++;
        }
    }
    for (int i = 0; i < n; i++) {
        System.out.print(arr[i] + " ");
    }
  }

  public static void swap(int[] arr, int a, int b)
  {
    int temp = arr[a];
    arr[a] = arr[b];
    arr[b] = temp;
  }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
 javac MoveZeros.java } ; if ($?) { java MoveZeros }
1 4 3 3 5 2 5 3 0 0 0 0 0
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

6) Write an application which will throw OverwtProductException if Product weight is above 60kg. (Use User defined exception)

Solution:

```
import java.util.*;

class OverProductWeightException extends Exception{
 OverProductWeightException(String msg){
 super(msg);
 }
}
public class OverWeight {
public static void main(String[] args){
 Scanner sc = new Scanner(System.in);
 int weight = sc.nextInt();
 try {
 checkWeight(weight);
 } catch (OverProductWeightException e) {
 System.out.println("An Exception Occurred: "+e);
 }

 }
static void checkWeight(int weight) throws OverProductWeightException{
 if(weight>60){
 throw new OverProductWeightException("Sorry..! Product weight can not be more than 60kgs.");
 }
 System.out.println("Product weight is okk.");
 }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Downloads\" ; if ($?) { javac weight.java } ; if
($?) { java weight }
An Exception Occurred: OverProductWeightException: Sorry..! Product weight can not be more than 60kgs.
PS C:\Users\Pardeshi\Downloads> cd "c:\Users\Pardeshi\Downloads\" ; if ($?) { javac weight.java } ; if ($?) { j
ava weight }
```

7) Given two arrays, 1,2,3,4,5 and 2,3,1,1,0,5,0,2,1 find which number is not present in the second array.

Solution:

```
import java.util.ArrayList;
public class FindNo {
	public static void main(String[] args) {
		int arr1[] = {1,2,3,4,5};
		int arr2[] = {2,3,1,1,0,5,0,2,1};
		ArrayList<Integer> notInSecondArray = new ArrayList<Integer>();

		int count = 0;
		for(int i:arr1){
		 for(int j = 0; j<arr2.length; j++){
		  if(i==arr2[j]){
		   count++;
		  }
		 }

		 if(count==0){
		  notInSecondArray.add(i);
		 }
		 count = 0;
		}

		System.out.println("number that are not present in second array are: "+notInSecondArray);

	       }
	      }
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
 javac FindNo.java } ; if ($?) { java FindNo }
number that are not present in second array are: [4]
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

8) Write code to check whether a no is a power of two or not?

Solution:

```
public class Exe{
 public static void main(String[] args) {
  int n = 64;
  if(divide(n)){
   System.out.println(n+" is power of 2");
  }else{
   System.out.println(n+" is not power of 2");
  }

 }

 static boolean divide(int n){
  if(n==1) return true;
  if(n%2!=0) return false;
  return divide(n/2);
 }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Downloads\" ; if ($?) { javac Exe.java } ; if ($?
) { java Exe }
64 is power of 2
PS C:\Users\Pardeshi\Downloads>
```

9) Write a code to display string in reverse order of words.

Solution:

```
public class ReverseString {

        public static String reverseWordWise(String input) {
                        int n = input.length();
        String rstr="";
    for(int i=0;i<n;i++){
        rstr = input.charAt(i)+rstr;
    }
    String output=reverseWord(rstr);
    return output;
        }

  public static String reverseWord(String str){
    String ans="";
    int currentStart=0;
    int i=0;
    for(;i<str.length();i++){
        if(str.charAt(i)==' '){
            int currentEnd=i-1;
            String rWord = "";
            for(int j=currentStart;j<=currentEnd;j++){
                rWord = str.charAt(j)+rWord;
            }
            ans+=rWord+" ";
            currentStart=i+1;
        }
    }

     int currentEnd=i-1;
            String rWord = "";
            for(int j=currentStart;j<=currentEnd;j++){
                rWord = str.charAt(j)+rWord;
            }
            ans+=rWord+" ";
                    return ans;
  }
        public static void main(String[] args) {
                    String str = "Java is best language";
                    System.out.println(reverseWordWise(str));
        }

}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
 javac ReverseString.java } ; if ($?) { java ReverseString }
language best is Java
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

10) Write a code to accept a string and check if there are two same consecutive letters, delete one of them.

Solution:

```
public class RemoveConsDuplicates {
        public static String removeDuplicateChar(String str){
                if(str.length()==1) return str;

                if(str.charAt(0)==str.charAt(1)){
                 return removeDuplicateChar(str.substring(1));
                }else{
                 return str.charAt(0) + removeDuplicateChar(str.substring(1));
                }
               }

        public static void main(String[] args) {
                String str = "sooonaa Good Morningg";
                System.out.println(removeDuplicateChar(str));
               }
}
```

Output:

```
PS C:\Users\Pardeshi\Desktop\practical> cd "c:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe\" ; if ($?) {
 javac RemoveConsDuplicates.java } ; if ($?) { java RemoveConsDuplicates }
sona God Morning
PS C:\Users\Pardeshi\Desktop\practical\aachal\JavaLabExe>
```

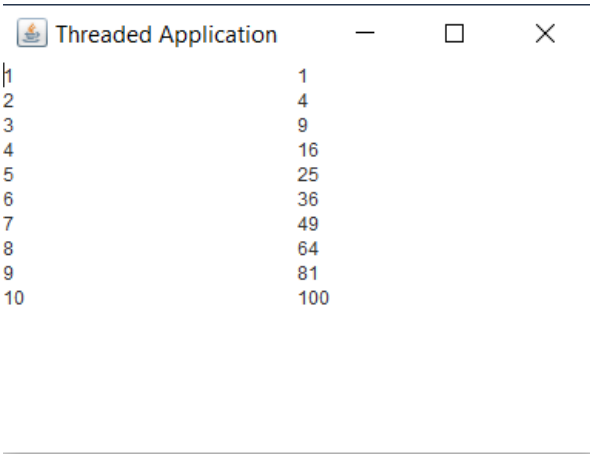
11) Write a threaded application to print in one text area 1,2,3,4.... and in other textarea 2,4,9,16 ....

Solution:

```
import java.awt.*;
import javax.swing.*;

public class ThreadedApp extends JFrame {

  // two text areas to display the sequences of numbers
  JTextArea ta1 = new JTextArea();
  JTextArea ta2 = new JTextArea();

  ThreadedApp() {
    // set the layout of the frame
    setLayout(new GridLayout(1, 2));

    // add text areas to the frame
    add(ta1);
    add(ta2);

    // create and start the first thread
    Thread t1 = new Thread(new Runnable() {
      public void run() {
        for (int i = 1; i <= 10; i++) {
          ta1.append(i + "\n");
          try {
            Thread.sleep(1000);
          } catch (Exception e) {
          }
        }
      }
    });
    t1.start();

    // create and start the second thread
    Thread t2 = new Thread(new Runnable() {
      public void run() {
        for (int i = 1; i <= 10; i++) {
          ta2.append(i * i + "\n");
          try {
            Thread.sleep(1000);
          } catch (Exception e) {
          }
```

```
      }
     }
    });
    t2.start();
  }

  public static void main(String[] args) {
    // create the frame
    ThreadedApp app = new ThreadedApp();

    // set the properties of the frame
    app.setSize(400, 300);
    app.setTitle("Threaded Application");
    app.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    app.setVisible(true);
  }
}
```

Output :



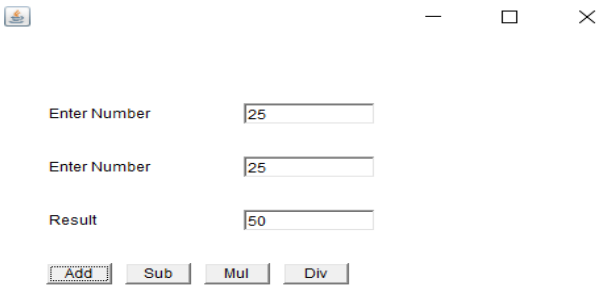
12) Write a code to create calculator application usingAWT, which will calculate simple Arithmetic Operations.

Solution:

```
import java.awt.*;
import java.awt.event.*;

class Calculator implements ActionListener
{

Frame f=new Frame();
Label l1=new Label("Enter Number");
Label l2=new Label("Enter Number");
Label l3=new Label("Result");
TextField t1=new TextField();
TextField t2=new TextField();
TextField t3=new TextField();
Button b1=new Button("Add");
Button b2=new Button("Sub");
Button b3=new Button("Mul");
Button b4=new Button("Div");


Calculator()
{
l1.setBounds(50,100,100,20);
l2.setBounds(50,150,100,20);
l3.setBounds(50,200,100,20);
t1.setBounds(200,100,100,20);
t2.setBounds(200,150,100,20);
t3.setBounds(200,200,100,20);
b1.setBounds(50,250,50,20);
b2.setBounds(110,250,50,20);
b3.setBounds(170,250,50,20);
b4.setBounds(230,250,50,20);
f.add(l1);
f.add(l2);
f.add(l3);
f.add(t1);
f.add(t2);
f.add(t3);
f.add(b1);
f.add(b2);
f.add(b3);
```

```
f.add(b4);
b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
f.setLayout(null);
f.setVisible(true);
f.setSize(500,500);
}


public void actionPerformed(ActionEvent e)
{
int i=Integer.parseInt(t1.getText());
int j=Integer.parseInt(t2.getText());

if(e.getSource()==b1)
{
t3.setText(String.valueOf(i+j));
}

if(e.getSource()==b2)
{
t3.setText(String.valueOf(i-j));
}

if(e.getSource()==b3)
{
t3.setText(String.valueOf(i*j));
}

if(e.getSource()==b4)
{
t3.setText(String.valueOf(i/j));
}
}

public static void main(String args[])
{
new Calculator();
}
}
```

Output:


Enter Number
25
Enter Number
25
Result
50
Add
Sub
Mul
Div

13) Write a Menu Driven Program for Blood Donorapplication for following task
a. Insert blood donor details into database.
b. Display blood group-wise details of donors
c. Update the address of a specific donor.
d. Delete the record of donors whose age is below 18.

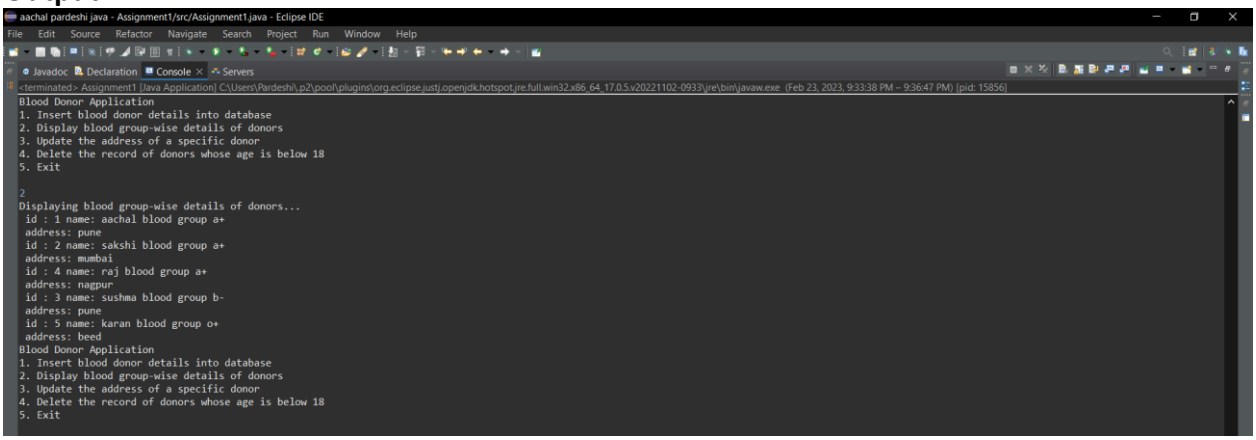
Solution:

```
import java.sql.*;
import java.util.Scanner;
import javafx.scene.effect.ColorInput;
public class Assignment1 {
public static void main(String[] args) {
// create a connection to the MySQL database
String url = "jdbc:mysql://localhost:3306/mydatabase";
String user = "root";
String password = "Project@12";
try {
// Driver
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection(url, user, password);
Scanner scanner = new Scanner(System.in);
int choice;
do{
// display menu and get user input
System.out.println("Blood Donor Application");
System.out.println("1. Insert blood donor details into database");
System.out.println("2. Display blood group-wise details of donors");
System.out.println("3. Update the address of a specific donor");
System.out.println("4. Delete the record of donors whose age is below 18");
System.out.println("5. Exit");
choice = scanner.nextInt();
if(choice<1 || choice>5){
System.out.println("Invalid choice");
}
if(choice==1){
System.out.println("Inserting blood donar details into database...");
// get user input for blood donor details
System.out.print("Enter donor Id: ");
int id = scanner.nextInt();
System.out.print("Enter donor name: ");
String name = scanner.next();
System.out.print("Enter donor blood group: ");
String bloodGroup = scanner.next();
System.out.print("Enter donor address: ");
String address = scanner.next();
```

```
PreparedStatement ps = con.prepareStatement("insert into donar values(?, ?, ?, ?)");
ps.setInt(1, id);
ps.setString(2, name);
ps.setString(3, bloodGroup);
ps.setString(4, address);
ps.executeUpdate();
System.out.println("Donor details have been inserted successfully");
}
if(choice==2){
System.out.println("Displaying blood group-wise details of donors...");
PreparedStatement ps = con.prepareStatement("SELECT * FROM donar ORDER BY bloodgroup");
ResultSet rs = ps.executeQuery();
// display donor details
String prevBloodGroup = "";
while (rs.next()) {
int id = rs.getInt(1);
String name = rs.getString(2);
String bloodGroup = rs.getString(3);
String address = rs.getString(4);
System.out.print(" id : " + id);
System.out.print(" name: " + name);
System.out.println(" blood group "+ bloodGroup);
System.out.print(" address: " + address);
System.out.println();
}
}
if(choice==3){
System.out.println("Updating the address of a specific donor...");
// get user input for donor name and new address
System.out.print("Enter donor id ");
int id = scanner.nextInt();
System.out.print("Enter new address: ");
String address = scanner.next();
PreparedStatement ps = con.prepareStatement("update donar set address=? where id=?");
ps.setString(1, address);
ps.setInt(2, id);
int rowsUpdated = ps.executeUpdate();
if (rowsUpdated > 0) {
System.out.println("Address of donor has been updated successfully ");
} else {
System.out.println("No such donor found in the database");
}
}
if(choice==4){
System.out.println("Deleting the record of donors whose age is below 18...");
PreparedStatement ps = con.prepareStatement("DELETE FROM donar WHERE age < 18");
```

```
int rowsDeleted = ps.executeUpdate();
System.out.println(rowsDeleted + " donor records have been deleted successfully");
}
if(choice==5){
System.out.println("Exiting the Blood Donor Application...");
return;
}
}while(choice!=5);
// close the connection
con.close();
} catch (Exception e) {
System.err.println("Error: " + e.getMessage());
}
}
}
```

Output:



14) Write a servlet to check username & password passedfrom html page. If it is “Scott” & “tiger”, display welcome message else show the same html page again. [With res.sendRedirect (“http://localhost:8080/login.html”)]

Solution:

Index.html

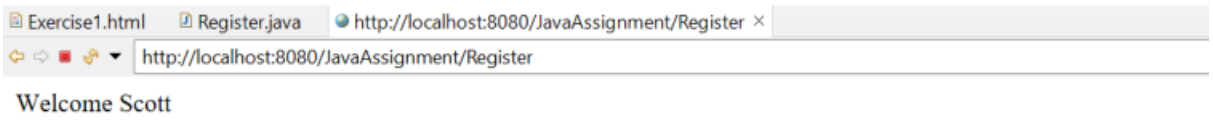
```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1" />
<title>Insert title here</title>
</head>
<body>
<h1>Login Form</h1>
<form method="post" action="Register">
<label id="name">Name</label>
<input type="text" name="name" /> <br />
<br />
<label id="password">Password</label>
<input type="text" name="password" /> <br />
<br />
<button type="submit">Submit</button>
</form>
</body>
</html>
```

Register.java

```
import java.io.*;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
/**
* Servlet implementation class Register
*/
@WebServlet("/Register")
public class Register extends HttpServlet {
private static final long serialVersionUID = 1L;
/**
* @see HttpServlet#HttpServlet()
*/
public Register() {
super();
// TODO Auto-generated constructor stub
```

```
}
/**
* @see HttpServlet#doPost(HttpServletRequest request,
HttpServletResponse response)
*/
protected void doPost(HttpServletRequest req, HttpServletResponse
resp) throws ServletException, IOException {
// TODO Auto-generated method stub
resp.setContentType("text/html");
PrintWriter pw = resp.getWriter();
String username = req.getParameter("name");
String password = req.getParameter("password");
if(username.equals("Scott") && password.equals("tiger")) {
pw.write("Welcome "+username);
return;
}
resp.sendRedirect("http://localhost:8080/JavaAssignment/Exercise1.
html");
}
}
```

Output:



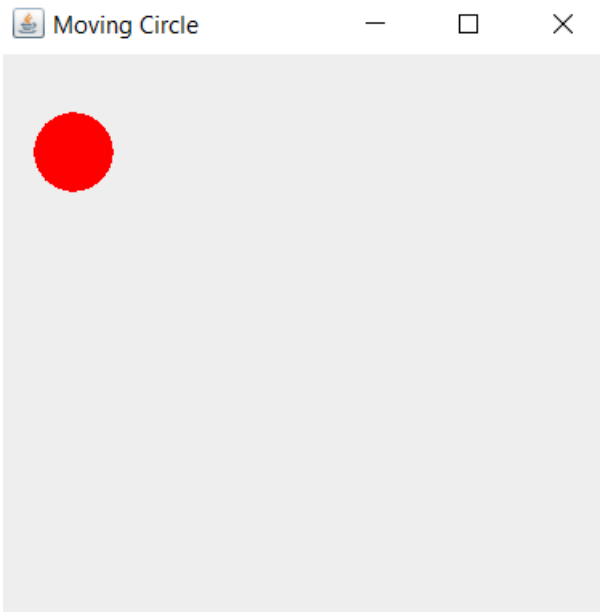
15) Write a program to draw a circle on panel and movethe circle as mouse is moving.

Solution:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class MovingCircle extends JFrame implements MouseMotionListener {

  int x, y;

  MovingCircle() {
    // add mouse motion listener to the frame
    addMouseMotionListener(this);

    // set the size of the frame
    setSize(400, 400);

    // set the title of the frame
    setTitle("Moving Circle");

    // set the default close operation of the frame
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    // set the visibility of the frame
    setVisible(true);
  }

  public void paint(Graphics g) {
    super.paint(g);
    g.setColor(Color.RED);
    g.fillOval(x, y, 50, 50);
  }

  public void mouseDragged(MouseEvent e) {
  }

  public void mouseMoved(MouseEvent e) {
    // get the x and y position of the mouse
    x = e.getX();
    y = e.getY();

    // repaint the frame
    repaint();
  }
```

```
  public static void main(String[] args) {
    // create the frame
    MovingCircle mc = new MovingCircle();
  }
}
```

Output:



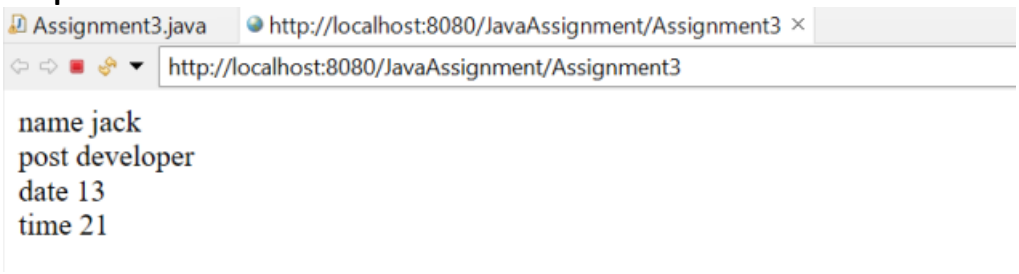
16) Write a servlet to add a Cookie to clients machine thatstores username, current date & time. Display the same.

Solution:

```
import java.io.*;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
/**
* Servlet implementation class Assignment3
*/
@WebServlet("/Assignment3")
public class Assignment3 extends HttpServlet {
private static final long serialVersionUID = 1L;
/**
* @see HttpServlet#HttpServlet()
*/
public Assignment3() {
super();
// TODO Auto-generated constructor stub
}
/**
* @see HttpServlet#doGet(HttpServletRequest request,
HttpServletResponse response)
*/
protected void doGet(HttpServletRequest req, HttpServletResponse
resp) throws ServletException, IOException {
// TODO Auto-generated method stub
resp.setContentType("text/html");
PrintWriter pw = resp.getWriter();
// Date
java.util.Date date = new java.util.Date();
String currentDate = String.valueOf(date.getDate());
String time = String.valueOf(date.getHours());
Cookie c1 = new Cookie("name", "jack");
Cookie c2 = new Cookie("post", "developer");
Cookie c3 = new Cookie("date", currentDate);
Cookie c4 = new Cookie("time", time);
resp.addCookie(c1);
resp.addCookie(c2);
resp.addCookie(c3);
resp.addCookie(c4);
```

```
String name, value;
Cookie c[] = req.getCookies();
for(Cookie i: c) {
name = i.getName();
value = i.getValue();
pw.println(name+" "+value+"<br>");
}
}
}
```

Output:



17) Write java program to generate 10 terms of Fibonacci threads.

Solution:

```
public class Fibonacci implements Runnable {

   private int termCount;
   private int previousTerm;
   private int currentTerm;

   public Fibonacci(int termCount) {
      this.termCount = termCount;
      this.previousTerm = 0;
      this.currentTerm = 1;
   }

   @Override
   synchronized public void run() {
      for (int i = 0; i < termCount; i++) {
         int nextTerm = previousTerm + currentTerm;
         System.out.println(Thread.currentThread().getName() + ": " + nextTerm);
         previousTerm = currentTerm;
         currentTerm = nextTerm;
      }
   }

   public static void main(String[] args) {
      Fibonacci fibonacciSeries = new Fibonacci(10);
      Thread thread1 = new Thread(fibonacciSeries, "Thread 1");

      thread1.start();

   }
}
```

Output:

```
C:\Users\Pardeshi\Desktop\Labex>java Fibonacci
Thread 1: 1
Thread 1: 2
Thread 1: 3
Thread 1: 5
Thread 1: 8
Thread 1: 13
Thread 1: 21
Thread 1: 34
Thread 1: 55
Thread 1: 89
```

18) Create a menu driven program for Bank account(acc_no, Name, amt) (Hint: use vector)

1. Add 2. Search 3. Delete 4. Display

Solution:

```
import java.util.Scanner;
import java.util.Vector;

class BankAccount {
    int acc_no;
    String name;
    double amt;

    public BankAccount(int acc_no, String name, double amt) {
        this.acc_no = acc_no;
        this.name = name;
        this.amt = amt;
    }
}

class Bank {
    Vector<BankAccount> accounts = new Vector<BankAccount>();

    public void addAccount() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter account number: ");
        int acc_no = scanner.nextInt();
        System.out.print("Enter name: ");
        String name = scanner.next();
        System.out.print("Enter amount: ");
        double amt = scanner.nextDouble();
        accounts.add(new BankAccount(acc_no, name, amt));
        System.out.println("Account added successfully!");
    }

    public void searchAccount() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter account number: ");
        int acc_no = scanner.nextInt();
        boolean found = false;
        for (BankAccount account : accounts) {
            if (account.acc_no == acc_no) {
                System.out.println("Account found!");
                System.out.println("Account number: " + account.acc_no);
                System.out.println("Name: " + account.name);
                System.out.println("Amount: " + account.amt);
```

```
            found = true;
            break;
        }
    }
    if (!found) {
        System.out.println("Account not found!");
    }
}

public void deleteAccount() {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter account number: ");
    int acc_no = scanner.nextInt();
    boolean found = false;
    for (BankAccount account : accounts) {
        if (account.acc_no == acc_no) {
            accounts.remove(account);
            System.out.println("Account deleted successfully!");
            found = true;
            break;
        }
    }
    if (!found) {
        System.out.println("Account not found!");
    }
}

public void displayAccounts() {
    if (accounts.size() == 0) {
        System.out.println("No accounts to display!");
    } else {
        for (BankAccount account : accounts) {
            System.out.println("Account number: " + account.acc_no);
            System.out.println("Name: " + account.name);
            System.out.println("Amount: " + account.amt);
            System.out.println();
        }
    }
}
}

public class BankAccMr {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Bank bank = new Bank();
        while (true) {
```

```
        System.out.println("Bank Account Management System");
        System.out.println("1. Add Account");
        System.out.println("2. Search Account");
        System.out.println("3. Delete Account");
        System.out.println("4. Display Accounts");
        System.out.println("5. Exit");
        System.out.print("Enter your choice: ");
        int choice = scanner.nextInt();
        switch (choice) {
            case 1:
                bank.addAccount();
                break;
            case 2:
                bank.searchAccount();
                break;
            case 3:
                bank.deleteAccount();
                break;
            case 4:
                bank.displayAccounts();
                break;
            case 5:
                System.exit(0);
                break;
            default:
                System.out.println("Invalid choice!");
        }
        System.out.println();
    }
  }
}
```

Output:

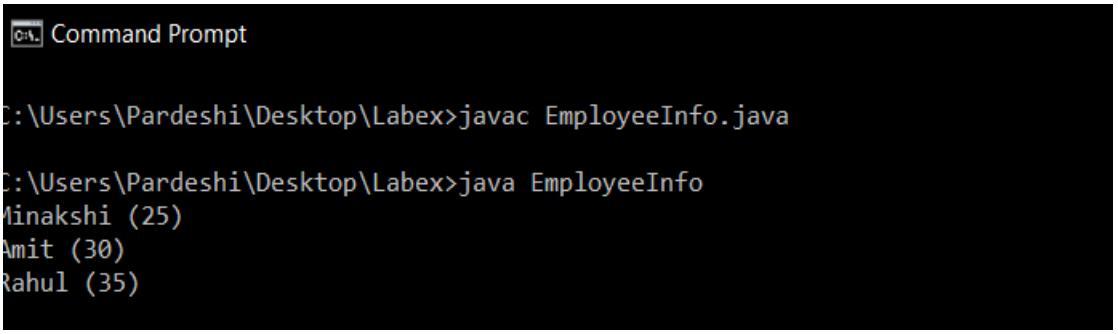
```
PS C:\Users\Pardeshi>  & 'C:\Program Files\Java\jdk1.8.0_351\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:
57695' '-cp' 'C:\Users\Pardeshi\AppData\Local\Temp\vscodesws_36c96\jdt_ws\jdt.ls-java-project\bin' 'BankAccMr'
Bank Account Management System
1. Add Account
2. Search Account
3. Delete Account
4. Display Accounts
5. Exit
Enter your choice: 1
Enter account number: 234
Enter name: aachal
Enter amount: 5000
Account added successfully!

Bank Account Management System
1. Add Account
2. Search Account
3. Delete Account
4. Display Accounts
5. Exit
Enter your choice: 2
Enter account number: 234
Account found!
Account number: 234
Name: aachal
Amount: 5000.0

Bank Account Management System
1. Add Account
2. Search Account
3. Delete Account
4. Display Accounts
5. Exit
Enter your choice: 4
Account number: 234
Name: aachal
Amount: 5000.0


Bank Account Management System
1. Add Account
2. Search Account
3. Delete Account
4. Display Accounts
5. Exit
Enter your choice: 5
PS C:\Users\Pardeshi>
```

19) Write a program to store employee in TreeSet and make sure employees are stored in sorted order oftheir age.

Solution:

```
import java.util.*;

class Employee implements Comparable<Employee> {
   private String name;
   private int age;

   public Employee(String name, int age) {
      this.name = name;
      this.age = age;
   }

   public String getName() {
      return name;
   }

   public int getAge() {
      return age;
   }

   @Override
   public int compareTo(Employee other) {
      return Integer.compare(this.age, other.age);
   }
}

public class EmployeeInfo {
   public static void main(String[] args) {
      TreeSet<Employee> employees = new TreeSet<Employee>();
       employees.add(new Employee("Amit", 30));
      employees.add(new Employee("Minakshi", 25));
      employees.add(new Employee("Rahul", 35));
      for (Employee employee : employees) {
         System.out.println(employee.getName() + " (" + employee.getAge() + ")");
      }
   }
}
```

Output:


Command Prompt
C:\Users\Pardeshi\Desktop\Labex>javac EmployeeInfo.java
C:\Users\Pardeshi\Desktop\Labex>java EmployeeInfo
Minakshi (25)
Amit (30)
Rahul (35)

20) Create the list of patients and display the names ofpatients starting with ‘A’

Solution:

```
import java.util.ArrayList;

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

    ArrayList<String> patients = new ArrayList<String>();
    patients.add("Ram");
    patients.add("Achal");
    patients.add("swaraj");
    patients.add("rish");
    patients.add("Anuj");
    patients.add("priyanka");

    for (String patient : patients) {
      if (patient.startsWith("A")) {
        System.out.println(patient);
      }
    }
  }
}
```

Output: