

**Q.1)Write a Java program to display all the alphabets between ‘A’ to ‘Z’ after every 2 seconds.**

```
public class Slip26_1 extends Thread
{
char c;
public void run()
{
for(c = 'A'; c<='Z';c++)
{
System.out.println(""+c);
try

{
Thread.sleep(2000);
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
public static void main(String args[])
{
Slip26_1 t = new Slip26_1();
t.start();
}
}
```

**Q.2)Write a Java program to accept the details of Employee (Eno, EName, Designation, Salary) from a user and store it into the database. (Use Swing)**

Database Assignment 3-Same as SET A Q.a

**Q.3)Write a java program to read ‘N’ names of your friends, store it into HashSet and display them in ascending order.**

```
// Java program to sort a HashSet

import java.util.*;

public class GFG {

    public static void main(String args[])

    {

        // Creating a HashSet

        HashSet<String> set = new HashSet<String>();

        // Adding elements into HashSet using add()

        set.add("geeks");

        set.add("practice");

        set.add("contribute");

        set.add("ide");

        System.out.println("Original HashSet: " + set);

        // Sorting HashSet using List

        List<String> list = new ArrayList<String>(set);

        Collections.sort(list);

        // Print the sorted elements of the HashSet

        System.out.println("HashSet elements "

            + "in sorted order "

            + "using List: "

            + list);

    }

}
```

**Q.4)Design a servlet that provides information about a HTTP request from a client, such as IP-Address and browser type. The servlet also provides information about the server on which the servlet is running, such as the operating system type, and the names of currently loaded servlets.**

Ans) Servlet & JSP Assignment 4 SET A Q.a

**Q.5)Write a JSP program to display the details of Patient (PNo, PName, Address, age, disease) in tabular form on browser.**

Ans))Servlet & JSP Assignment 4 SET B Q.c

**Q.6)Write a Java program to create LinkedList of String objects and perform the following: i. Add element at the end of the list ii. Delete first element of the list iii. Display the contents of list in reverse order**

Ans)Collection Assignment 1 refer

**Q.7)Write a Java program using Runnable interface to blink Text on the frame**

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

class Slip8_1 extends Frame implements Runnable
{
    Thread t;
    Label l1;
    int f;
    Slip8_1()
    {
        t=new Thread(this);
        t.start();
        setLayout(null);
        l1=new Label("Hello JAVA");
        l1.setBounds(100,100,100,40);
        add(l1);
        setSize(300,300);
        setVisible(true);
        f=0;
    }
    public void run()
```

```

    {
        try
        {
            if(f==0)
            {
                t.sleep(200);
                l1.setText("");
                f=1;
            }
            if(f==1)
            {
                t.sleep(200);
                l1.setText("Hello Java");
                f=0;
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
        run();
    }
    public static void main(String a[])
    {
        new Slip8_1();
    }
}

```

**Q.8) Write a Java program to store city names and their STD codes using an appropriate collection and perform following operations: i. Add a new city and its code (No duplicates) ii. Remove a city from the collection iii. Search for a city name and display the code**

```

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

```

```

class Slip16_2 extends JFrame implements ActionListener
{
    JTextField t1,t2,t3;
    JButton b1,b2,b3;

```

```

JTextArea t;
JPanel p1,p2;

Hashtable ts;
Slip16_2()
{
    ts=new Hashtable();
    t1=new JTextField(10);
    t2=new JTextField(10);
    t3=new JTextField(10);

    b1=new JButton("Add");
    b2=new JButton("Search");
    b3=new JButton("Remove");

    t=new JTextArea(20,20);
    p1=new JPanel();
    p1.add(t);

    p2= new JPanel();
    p2.setLayout(new GridLayout(2,3));
    p2.add(t1);
    p2.add(t2);
    p2.add(b1);
    p2.add(t3);
    p2.add(b2);
    p2.add(b3);

    add(p1);
    add(p2);

    b1.addActionListener(this);
    b2.addActionListener(this);
    b3.addActionListener(this);

    setLayout(new FlowLayout());
    setSize(500,500);
    setVisible(true);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

}
public void actionPerformed(ActionEvent e)
{
    if(b1==e.getSource())
    {

```

```

        String name = t1.getText();
        int code = Integer.parseInt(t2.getText());
        ts.put(name,code);
        Enumeration k=ts.keys();
        Enumeration v=ts.elements();
        String msg="";
        while(k.hasMoreElements())
        {
            msg=msg+k.nextElement()+" = "+v.nextElement()+"\n";
        }
        t.setText(msg);
        t1.setText("");
        t2.setText("");
    }
    else if(b2==e.getSource())
    {
        String name = t3.getText();

        if(ts.containsKey(name))
        {
            t.setText(ts.get(name).toString());
        }

        else
            JOptionPane.showMessageDialog(null,"City not found ...");
    }
    else if(b3==e.getSource())
    {
        String name = t3.getText();

        if(ts.containsKey(name))
        {
            ts.remove(name);
            JOptionPane.showMessageDialog(null,"City Deleted ...");
        }

        else
            JOptionPane.showMessageDialog(null,"City not found ...");
    }
}
public static void main(String a[])
{
    new Slip16_2();
}
}

```

**Q.9)Write a Java Program to create the hash table that will maintain the mobile number and student name. Display the details of student using Enumeration interface**

```
import java.io.*;
import java.util.Enumeration;
import java.util.Hashtable;

// Main class
// EnumerationOnKeys
public class GFG {

    // Main driver method
    public static void main(String[] args)
    {
        // Creating an empty hashtable
        Hashtable<Integer, String> ht
            = new Hashtable<Integer, String>();

        // Inserting key-value pairs into hash table
        // using put() method
        ht.put(1, "Geeks");
        ht.put(2, "for");
        ht.put(3, "Geeks");

        // Now creating an Enumeration object
        // to read elements
        Enumeration e = ht.elements();

        // Condition holds true till there is
        // single key remaining

        // Printing elements of hashtable
        // using enumeration
        while (e.hasMoreElements()) {

            // Printing the current element
            System.out.println(e.nextElement());
        }
    }
}
```

**Q.10) Create a JSP page for an online multiple choice test. The questions are randomly selected from a database and displayed on the screen. The choices are displayed using radio buttons. When the user clicks on next, the next question is displayed. When the user clicks on submit, display the total score on the screen**

### **Exam.jsp**

```
<% @page import="java.sql.*,java.util.*"%>
<%
    // load a driver
    Class.forName("org.postgresql.Driver");
    // Establish Connection
    conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90", "");
    Set s = new TreeSet();

    while(true){
        int n = (int)(Math.random()*11+1);

        s.add(n);

        if(s.size()==5) break;
    }

    PreparedStatement ps = con.prepareStatement("select * from questions where qid=?");
%>
<form method='post' action='accept_ans.jsp'>
<table width='70%' align='center'>
<%
    int i=0;
    Vector v = new Vector(s);
    session.setAttribute("qids",v);

    int qid = Integer.parseInt(v.get(i).toString());
    ps.setInt(1,qid);
    ResultSet rs = ps.executeQuery();
    rs.next();
%>
<tr>
    <td><b>Question:<%=i+1%></b></td>
</tr>
<tr>
```



```

        <td><pre><b><%=rs.getString(2)%></pre></b></td>
</tr>
<tr>
    <td>
        <b>
            <input type='radio' name='op' value=1><%=rs.getString(3)%><br>
            <input type='radio' name='op' value=2><%=rs.getString(4)%><br>
            <input type='radio' name='op' value=3><%=rs.getString(5)%><br>
            <input type='radio' name='op' value=4><%=rs.getString(6)%><br><br>
        </b>
    </td>
</tr>
<tr>
    <td align='center'>
        <input type='submit' value='Next' name='ok'>
        <input type='submit' value='Submit' name='ok'>
    </td>
</tr>
</table>
<input type='hidden' name='qno' value=<%=qid%>>
<input type='hidden' name='qid' value=<%=i+1%>>
</form>
</body>

```

### acceptans.jsp:

```

<% @page import="java.sql.*,java.util.*"%>
<%
    Class.forName("org.postgresql.Driver");

    Connection con = DriverManager.getConnection(
        "jdbc:postgresql:ty1","postgres","");

    Vector answers = (Vector)session.getAttribute("answers");

    if(answers==null)
        answers = new Vector();

    int qno = Integer.parseInt(request.getParameter("qno"));
    int ans = Integer.parseInt(request.getParameter("op"));
    int i = Integer.parseInt(request.getParameter("qid"));

    answers.add(qno+" "+ans);

    session.setAttribute("answers",answers);

```

```

String ok = request.getParameter("ok");

if(ok.equals("Submit") || i==5){
    response.sendRedirect("result.jsp");
    return;
}

PreparedStatement ps = con.prepareStatement("select * from questions where qid=?");
%>
<form method='post' action='accept_ans.jsp'>
<table width='70%' align='center'>
<%
    Vector v = (Vector)session.getAttribute("qids");

    int qid = Integer.parseInt(v.get(i).toString());
    ps.setInt(1,qid);
    ResultSet rs = ps.executeQuery();
    rs.next();
%>
<tr>
<td><b>Question:<%=i+1%></b></td>
</tr>
<tr>
<td><pre><b><%=rs.getString(2)%></pre></b></td>
</tr>
<tr>
<td>
<b>
<input type='radio' name='op' value=1><%=rs.getString(3)%><br>
<input type='radio' name='op' value=2><%=rs.getString(4)%><br>
<input type='radio' name='op' value=3><%=rs.getString(5)%><br>
<input type='radio' name='op' value=4><%=rs.getString(6)%><br><br>
</b>
</td>
</tr>
<tr>
<td align='center'>
<input type='submit' value='Next' name='ok'>
<input type='submit' value='Submit' name='ok'>
</td>
</tr>
</table>
<input type='hidden' name='qno' value=<%=qid%>>
<input type='hidden' name='qid' value=<%=i+1%>>
</form>

```

</body>

### result.jsp:

```
<% @page import="java.sql.*,java.util.*,java.text.*"%>
<%
    // load a driver
    Class.forName("org.postgresql.Driver");
    // Establish Connection
    conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90",
    "");
    Vector v = (Vector)session.getAttribute("answers");
    if(v==null){
%>
<h1>No questions answered</h1>
<%
    return;
    }

    PreparedStatement ps = con.prepareStatement("select ans from questions where qid=?");

    int tot=0;

    for(int i=0;i<v.size();i++){
        String str = v.get(i).toString();
        int j = str.indexOf(' ');
        int qno = Integer.parseInt(str.substring(0,j));
        int gans = Integer.parseInt(str.substring(j+1));

        ps.setInt(1,qno);

        ResultSet rs = ps.executeQuery();
        rs.next();

        int cans = rs.getInt(1);

        if(gans==cans) tot++;
    }

    session.removeAttribute("qids");
    session.removeAttribute("answers");
    session.removeAttribute("qid");
%>
<h3>Score:<%=tot%></h1>
<center><a href='exam.jsp'>Restart</a></center>
</body>
```

Postgresql

create table questions(qid serial primary key, question text, option1 text, option2 text, option3 text, option4 text, ans int);

insert into questions(question,option1,option2,option3,option4,ans) values('Who is prime minister of India?','Rahul Gandhi','Narendra Modi','Sonia Gandhi','Manmohan Singh',2),('Who is finance minister of India','Rahul Gandhi','P Chidambaram','Manmohan Singh','Arun Jately',4),('What is square root of 16?','2','4','1','256',4),('Who is chief minister of Maharashtra','Uddhav Tharakey','Devendra Fadanavis','Raj Thakarey','Sharad Pawar',2),('What is full for of LIFO?','Last In First Out','Late In First Out','Long In First Out','Large In First Out',1),('Which is capital of India','Delhi','Maharashtra','Kolkata','Goa',1), ('What is currency of India','Dollar','Rupee','Pound','Yen',2),('Who Invented C?','Kim Thompson','Bill Joy','Dennis Ritche','Balaguru Swamy',3),('Where was Java invented?','Microsoft','Oracle','Sun Microsystem','Intel',3),('What is cube root of 8?','2','3','4','5',1),('What is full form of FIFO','Fast In Fast Out','First in First Out','Fast In First Out','First In Fast Out',2);

**Q.10)Write a Java program to accept ‘n’ integers from the user and store them in a collection. Display them in the sorted order. The collection should not accept duplicate elements. (Use a suitable collection). Search for a particular element using predefined search method in the Collection framework.**

```
import java.util.*;
import java.io.*;

class Slip19_2
{
    public static void main(String[] args) throws Exception
    {
        int no,element,i;
        BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
        TreeSet ts=new TreeSet();
        System.out.println("Enter the of elements :");
        no=Integer.parseInt(br.readLine());
        for(i=0;i<no;i++)
        {
            System.out.println("Enter the element : ");
```

```

        element=Integer.parseInt(br.readLine());
        ts.add(element);
    }

    System.out.println("The elements in sorted order :"+ts);
    System.out.println("Enter element to be serach : ");
    element = Integer.parseInt(br.readLine());
    if(ts.contains(element))
        System.out.println("Element is found");
    else
        System.out.println("Element is NOT found");
}
}

```

**Q.11)Write a java program to simulate traffic signal using threads.**

```

import java.applet.*;
public class signal extends Applet implements Runnable
{
    int r,g1,y,i;
    Thread t;
    public void init()
    {
        r=0;g1=0;y=0;
        t=new Thread(this);
        t.start();
    }
    public void run()
    {
        try    { for (i=24;i>=1;i--)
        {
            t.sleep(100);
            if (i>=16 && i<24)
            {
                g1=1;
                repaint();
            }
            if (i>=8 && i<16)
            {
                y=1;
                repaint();
            }
            if (i>=1 && i<8)
            {

```

```

        r=1;
        repaint();
    }
}
23     if (i==0)
    {
        run();
    }
}
catch(Exception e)
{
}
}

public void paint(Graphics g) {
    g.drawOval(100,100,100,100);
    g.drawOval(100,225,100,100);
    g.drawOval(100,350,100,100);
    g.drawString("start",200,200);
    if (r==1)
    { g.setColor(Color.red);
      g.fillOval(100,100,100,100);
      g.drawOval(100,100,100,100);
      g.drawString("stop",200,200);
      r=0;
    }
    if (g1==1)
    { g.setColor(Color.green);
      g.fillOval(100,225,100,100);
      g1=0;
      g.drawOval(100,225,100,100);
      g.drawString("go",200,200);
    }
    if (y==1)
    { g.setColor(Color.yellow);
      g.fillOval(100,350,100,100);
      y=0;
      g.drawOval(100,350,100,100);
      g.drawString("slow",200,200);
    }
}
}
}

```

**Q.12)Write a java program that implements a multi-thread application that has three threads. First thread generates random integer number after every one second, if the number is even; second thread computes the square of that number and print it. If the number is odd, the third thread computes the of cube of that number and print it**

Ans)Multithreading Assignment 2 SET B Q.c

**Q.13)Write a java program for the following: i. To create a Product(Pid, Pname, Price) table. ii. Insert at least five records into the table. iii. Display all the records from a table.**

Ans) Database Assignment 3 Same SET A Q.a

**Q.14)Write a java program to define a thread for printing text on output screen for ‘n’ number of times. Create 3 threads and run them. Pass the text ‘n’ parameters to the thread constructor. Example: i. First thread prints “COVID19” 10 times. ii. Second thread prints “LOCKDOWN2020” 20 times iii. Third thread prints “VACCINATED2021” 30 times**

Ans) Multithreading Assignment 2 SET A Q.a

**Q.15)Write a JSP program to check whether a given number is prime or not. Display the result in red color.**

Html file

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
</head>
<body>
<form action="Slip30.jsp" method="post">
Enter Number :
<input type="text" name="num">
<input type="submit" value="Submit">
</form>
</body>
</html>
```

JSP FILE :Slip30.jsp

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

```
<%  
int n = Integer.parseInt(request.getParameter("num"));  
int num,i,count;  
for(num=1;num<=n;num++)  
{  
count=0;  
for(i=2;i<=num/2;i++)  
{  
if(num%i==0)  
{  
count++;  
break;  
}  
}  
if(count==0 &&num!=1)  
{  
%>  
<html>  
<body>  
<font size="14" color="blue"><% out.println("\t"+num);%>  
</body>  
</html>  
<%  
}  
}  
%>
```

**Q.16)Write a Java program to create a thread for moving a ball inside a panel vertically. The ball should be created when the user clicks on the start button.**

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



```

import javax.swing.*;

class bouncingthread extends JFrame implements Runnable
{
    Thread t;
    int x,y;

    bouncingthread()
    {
        super();
        t= new Thread(this);
        x=10;
        y=10;
        t.start();
        setSize(1000,200);
        setVisible(true);
        setTitle("BOUNCEING BOLL WINDOW");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
    public void run()
    {
        try
        {
            while(true)
            {
                x+=10;
                y+=10;
                repaint();
                Thread.sleep(1000);
            }
        }catch(Exception e)
        {

        }
    }

    public void paint(Graphics g)
    {

        g.drawOval(x,y,7,7);
    }
}

```

```

}

public static void main(String a[])throws Exception
{
    bouncingthread t=new bouncingthread();
    Thread.sleep(1000);
}
}

```

**Q.17)Write a Java program using Spring to display the message “If you can't explain it simply, you don't understand it well enough”.**

Ans) Spring Assignment 5 SET A Q.a

**Q.18)Write a Java program to display the Current Date using spring**

Ans) Spring Assignmentn 5 SET A Q.b

**Q.19)Write a Java program to display first record from student table (RNo, SName, Per) onto the TextFields by clicking on button. (Assume Student table is already created).**

Ans) Servlet & JSP Assignment 4 same SET B Q. a

**Q.20)Design an HTML page which passes customer number to a search servlet. The servlet searches for the customer number in a database (customer table) and returns customer details if found the number otherwise display error message.**

Ans) Servlet & JSP Assignment 4 SET B Q. a

**Q.21)Write a Java program to display information about all columns in the DONAR table using ResultSetMetaData.**

Ans) Database Assignment 3 SET A Q.c

**Q.22)Write a JSP program to check whether given number is Perfect or not. (Use Include directive).**

**Index.html file:**

```
<!DOCTYPE html>
<html>
<head>
<title>PERFECT NUMBER</title>
</head>
<body>
<form action="perfect.jsp" method="post">
Enter Number :<input type="text" name="num">
<input type="submit" value="Submit" name="s1">
</form>
</body>
</html>
```

**Perfect.jsp file:**

```
<% @ page import="java.util.*" %>

<%
if(request.getParameter("s1")!=null)
{
    Integer num,a,i,sum = 0;

    num = Integer.parseInt(request.getParameter("num"));
    a = num;

    for(i=1;i<a;i++)
    {
        if(a%i==0)
        {
            sum=sum + i;
        }
    }
    if(sum==a)
    {
        out.println(+num+ "is a perfect number");
    }
    else
```

```

        {
            out.println(+num+ "is not a perfect number");
        }
    }
%>

```

**Q.23)Write a Java Program to create a PROJECT table with field's project\_id, Project\_name, Project\_description, Project\_Status. Insert values in the table. Display all the details of the PROJECT table in a tabular format on the screen.(using swing).**

Ans) Database Assignment 3 SET A Q.a

**Q.24)Write a Java program to display information about the database and list all the tables in the database. (Use DatabaseMetaData).**

Ans) Database Assignment 3 SET A Q.b

**Q.25)Write a Java program to show lifecycle (creation, sleep, and dead) of a thread. Program should print randomly the name of thread and value of sleep time. The name of the thread should be hard coded through constructor. The sleep time of a thread will be a random integer in the range 0 to 4999**

```

Class MyThread extends Thread
{ public MyThread(String s)
{
super(s);
}
public void run()
{
System.out.println(getName()+"thread created.");
while(true)
{
System.out.println(this);
int s=(int)(math.random()*5000);
System.out.println(getName()+"is sleeping for :+s+"msec");
try{
Thread.sleep(s);

```

```

    }
    catch(Exception e)
    {
    }
    }
    }
    Class ThreadLifeCycle
    {
    public static void main(String args[])
    {
    MyThread t1=new MyThread("shradha"),t2=new MyThread("pooja");
    t1.start();
    t2.start();
    try
    {
    t1.join();
    t2.join();
    }
    catch(Exception e)
    {
    }
    System.out.println(t1.getName()+"thread dead.");
    System.out.println(t2.getName()+"thread dead.");
    }
    }

```

**Q.26) Write a Java program for a simple search engine. Accept a string to be searched. Search the string in all text files in the current folder. Use a separate thread for each file. The result should display the filename and line number where the string is found**

Ans) Multithreading Assignment 2 SET B Q.b

**Q.27) Write a JSP program to calculate sum of first and last digit of a given number. Display sum in Red Color with font size 18.**

HTML FILE

```

<html>
<body>

```

```
<form method=post action="Slip7.jsp">
Enter Any Number : <Input type=text name=num>
<input type=submit value=Display>
</form>
</body>
</html>
```

JSP FILE:

```
<% @page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<body>
<% ! intn,rem,r; %>
<% n=Integer.parseInt(request.getParameter("num"));
if(n<10)
{
out.println("Sum of first and last digit is ");
%><font size=18 color=red><%= n %>
<%
}
else
{
rem=n% 10;
do
{
r=n% 10;
n=n/10;
}while(n>0);
n=rem+r;
out.println("Sum of first and last digit is ");
%><font size=18 color=red><%= n %>
<%
}
%>
</body>
</html>
```

**Q.28)Write a java program to display name and priority of a Thread**

**public class MainThread**

```

{
    public static void main(String arg[])
    {
        Thread t=Thread.currentThread();
        System.out.println("Current Thread:"+t);
        //Change Name      t.setName("My Thread ");
        System.out.println ("After the name is Changed:"+t);
        try    {
            for(int i=2;i>0;i--)
            {
                System.out.println(i);
                Thread.sleep(1000);
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

```

**Q.29)Write a SERVLET program which counts how many times a user has visited a web page. If user is visiting the page for the first time, display a welcome message. If the user is revisiting the page, display the number of times visited. (Use Cookie)**

JAVA Code

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class count extends HttpServlet
{
    static int count=0,c2=0;
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws
        ServletException, IOException
    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        String name=request.getParameter("t1");

        Cookie c1=new Cookie("count",String.valueOf(count));
        c2=Integer.parseInt(c1.getValue());

        if(c2==0)
        {

```

```

        out.println("Welcome="+name);
        count++;
    }
    else
    {

        c1=new Cookie("count",String.valueOf(count));
        count++;
        out.println("Welcome="+name+"\t"+count);
    }
}
}
}

```

### HTML Code

```

<html>
  <body>
    <form action="http://localhost:8080/serv/count" method="get">
      Username:<input type="text" name="t1">
        <input type="submit" >
    </form>
  </body>
</html>

```

**Q.30)Write a java program to create a TreeSet, add some colors (String) and print out the content of TreeSet in ascending order.**

```

import java.util.TreeSet;
public class Exercise1 {
    public static void main(String[] args) {
        TreeSet<String> tree_set = new TreeSet<String>();
        tree_set.add("Red");
        tree_set.add("Green");
        tree_set.add("Orange");
        tree_set.add("White");
        tree_set.add("Black");
        System.out.println("Tree set: ");
        System.out.println(tree_set);
    }
}

```

**Q.31)Write a Java program to accept the details of Teacher (TNo, TName, Subject). Insert at least 5 Records into Teacher Table and display the details of Teacher who is teaching “JAVA” Subject. (Use PreparedStatement Interface)**



```

import java.sql.*;
import java.io.*;
class Slip6_1
{
    public static void main(String a[])
    {
        PreparedStatement ps;
        Connection con;
        try{
            Class.forName("");
            con=DriverManager.getConnection("");
            if(con==null)
            {
                System.out.println("Connection Failed.....");
                System.exit(1);
            }
            System.out.println("Connection Esablished.....");
            Statement stmt=con.createStatement();

            BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
            String query="insert into Customer values(?,?,?,?)";
            ps=con.prepareStatement(query);

            System.out.println("Customer Details....");
            System.out.println("Enter CID");
            int cid=Integer.parseInt(br.readLine());
            ps.setInt(1,cid);
            System.out.println("Enter name");
            String name=br.readLine();
            ps.setString(2,name);
            System.out.println("Enter Address");
            String add=br.readLine();
            ps.setString(3,add);
            System.out.println("Enter Ph_No");
            String phno=br.readLine();
            ps.setString(4,phno);

            int no=ps.executeUpdate();
            if(no!=0)
                System.out.println("Data inserted succesfully.....");
            else
                System.out.println("Data NOT inserted");
            con.close();

```

```

    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}

```

**Q.32)Write a java program to accept ‘N’ integers from a user. Store and display integers in sorted order having proper collection class. The collection should not accept duplicate elements**

```

import java.util.*;
import java.io.*;

class SortedNumbers{
    public static void main(String[] args) throws Exception {
        BufferedReader br = new BufferedReader(
            new InputStreamReader(System.in));

        Set s = new TreeSet();

        System.out.print("Enter no.of integers:");
        int n = Integer.parseInt(br.readLine());

        for(int i = 0; i < n; i++) {
            System.out.print("Enter number:");
            int x = Integer.parseInt(br.readLine());
            s.add(x);
        }

        Iterator itr = s.iterator();
        while (itr.hasNext()) {
            System.out.println(itr.next());
        }

        System.out.print("Enter element to be searched:");
        int no = Integer.parseInt(br.readLine());

        if(s.contains(no))
            System.out.println("Number "+no+" found.");
        else
            System.out.println("Number "+no+" not found.");
    }
}

```

```
}  
}
```

**Q.33)Write a Multithreading program in java to display the number's between 1 to 100 continuously in a TextField by clicking on button. (Use Runnable Interface).**

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.ActionListener;  
import java.awt.event.ActionEvent;  
public class MultiThread extends JFrame implements ActionListener  
{  
    Container cc;  
    JButton b1,b2;  
    JTextField t1;  
    MultiThread()  
    {  
        setVisible(true);  
        setSize(1024,768);  
        cc=getContentPane();  
        setLayout(null);  
        t1=new JTextField(500);  
        cc.add(t1);  
        t1.setBounds(10,10,1000,30);  
        b1=new JButton("start");  
        cc.add(b1);  
        b1.setBounds(20,50,100,40);  
        b1.addActionListener(this);  
        setDefaultCloseOperation(EXIT_ON_CLOSE);  
    }  
    public void actionPerformed(ActionEvent e)  
    {  
        if(e.getSource()==b1)  
        {  
            new Mythread();  
        }  
    }  
    class Mythread extends Thread  
    {  
        Mythread()  
        {  
            start();  
        }  
        public void run()  
        {  
            for(int i=1;i<=100;i++)
```

```

        {
            try
            {
                Thread.sleep(1000);
            }
            catch (InterruptedException e) {
            }
            t1.setText(t1.getText()+""+i+"\n");
//System.out.println()
        }
    }
    public static void main(String arg[])
    {
        new MultiThread().show();
    }
}

```

**Q.34) Write a SERVLET program in java to accept details of student (SeatNo, Stud\_Name, Class, Total\_Marks). Calculate percentage and grade obtained and display details on page.**

#### Student.html

```

<html>
<body>
<form name="f1" method="Post" action="http://localhost:8080/Servlet/Student">
<fieldset>
<legend><b><i>Enter Student Details :</i><b></legend>
    Enter Roll No :&nbsp;&nbsp;&nbsp;<input type="text" name="txtsno"><br><br>
    Enter Name :&nbsp;&nbsp;&nbsp;<input type="text" name="txtnm"><br><br>
    Enter class :&nbsp;&nbsp;&nbsp;<input type="text" name="txtclass"><br><br>
</fieldset>
<legend><b><i>Enter Student Marks Details :</i><b></legend>
    Subject 1 :&nbsp;&nbsp;&nbsp;<input type="text" name="txtsub1"><br><br>
    Subject 2 :&nbsp;&nbsp;&nbsp;<input type="text" name="txtsub2"><br><br>
    Subject 3 :&nbsp;&nbsp;&nbsp;<input type="text" name="txtsub3"><br><br>
</fieldset>
</fieldset>
<div align=center>
<input type="submit" value="Result">
</div>
</form>
</body>
</html>

```

#### Student.java

```

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

```

```

import javax.servlet.http.*;
public class Student extends HttpServlet
{
    public void doPost(HttpServletRequest req,HttpServletResponse res)throws
IOException,ServletException
    {
        int sno,s1,s2,s3,total;
        String snm,sclass;
        float per;
        40    res.setContentType("text/html");
        PrintWriter out=res.getWriter();
        sno=Integer.parseInt(req.getParameter("txtsno"));
        snm=req.getParameter("txtnm");
        sclass=req.getParameter("txtclass");
        s1=Integer.parseInt(req.getParameter("txtsub1"));
        s2=Integer.parseInt(req.getParameter("txtsub2"));
        s3=Integer.parseInt(req.getParameter("txtsub3"));
        total=s1+s2+s3;
        per=(total/3);
        out.println("<html><body>");
        out.print("<h2>Result of student</h2><br>");
        out.println("<b><i>Roll No :</b></i>" +sno+"<br>");
        out.println("<b><i>Name :</b></i>" +snm+"<br>");
        out.println("<b><i>Class :</b></i>" +sclass+"<br>");
        out.println("<b><i>Subject1:</b></i>" +s1+"<br>");
        out.println("<b><i>Subject2:</b></i>" +s2+"<br>");
        out.println("<b><i>Subject3:</b></i>" +s3+"<br>");
        out.println("<b><i>Total :</b></i>" +total+"<br>");
        out.println("<b><i>Percentage :</b></i>" +per+"<br>");
        if(per<50)
            out.println("<h1><i>Pass Class</i></h1>");
        else if(per<55 && per>50)
            out.println("<h1><i>Second class</i></h1>");
        else if(per<60 && per>=55)
            out.println("<h1><i>Higher class</i></h1>");
        out.close();
    }
}

```

### Web.xml

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app>
<servlet>
<servlet-name>Student</servlet-name>
<servlet-class>Student</servlet-class>
</servlet>

```

```
41<servlet-mapping>
<servlet-name>Student</servlet-name>
<url-pattern>/Student</url-pattern>
</servlet-mapping>
</web-app>
```

**Q.35)Write a java program to accept ‘N’ Integers from a user store them into LinkedList Collection and display only negative integers**

Ans) Collection Assignment 1 refer

**Q.36)Write a SERVLET application to accept username and password, search them into database, if found then display appropriate message on the browser otherwise display error message**

UserPass.html

```
<html>
<body>
<form method=post action="http://localhost:4141/Program/servlet/UserPass">
User Name :<input type=text name=user><br><br>
Password :<input type=text name=pass><br><br>
<input type=submit value="Login">
</form>
</body>
</html>
```

UserPass.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class UserPass extends HttpServlet
{
```

```

public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws IOException, ServletException
{
    PrintWriter out = response.getWriter();
    try{
        String us=request.getParameter("user");
        String pa=request.getParameter("pass");
        // load a driver

Class.forName("org.postgresql.Driver");

// Establish Connection

conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90", "");
        Statement st=cn.createStatement();
        ResultSet rs=st.executeQuery("select * from UserPass");
        while(rs.next())
        {
            if(us.equals(rs.getString("user"))&&pa.equals(rs.getString("pass")))
                out.println("Valid user");
            else
                out.println("Invalid user");
        }
    }catch(Exception e)
    {
        out.println(e);
    }
}

public void doPost(HttpServletRequest request, HttpServletResponse response)
    throws IOException, ServletException
{
    doGet(request, response);
}
}

```

Web.xml file(servlet entry)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app>
<servlet>
    <servlet-name>UserPass</servlet-name>
    <servlet-class>UserPass</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>UserPass</servlet-name>
    <url-pattern>/servlet/UserPass</url-pattern>
</servlet-mapping>
</web-app>
```

**Q.37) Create a JSP page to accept a number from a user and display it in words: Example: 123 – One Two Three. The output should be in red color.**

NumberWord.html

```
<html>
<body>
<form method=get action="NumberWord.jsp">
Enter Any Number : <input type=text name=num><br><br>
<input type=submit value="Display">
</form>
</body>
</html>
```

NumberWord.jsp

```
<html>
<body>
<font color=red>
<%! int i,n;
    String s1;
```



```

%>
<%  s1=request.getParameter("num");
    n=s1.length();
    i=0;
    do
    {
        char ch=s1.charAt(i);
        switch(ch)
        {
            case '0': out.println("Zero ");break;
            case '1': out.println("One ");break;
            case '2': out.println("Two ");break;
            case '3': out.println("Three ");break;
            case '4': out.println("Four ");break;
            case '5': out.println("Five ");break;
            case '6': out.println("Six ");break;
            case '7': out.println("Seven ");break;
            case '8': out.println("Eight ");break;
            case '9': out.println("Nine ");break;
        }
        i++;
    }while(i<n);
%>
</font>
</body>
</html>

```

**Q.38)Write a java program to blink image on the JFrame continuously.**

```

import java.awt.Color;
import java.awt.Graphics;
import java.awt.Image;
import java.io.File;

import javax.imageio.ImageIO;
import javax.swing.ImageIcon;
import javax.swing.JFrame;

public class Game extends JFrame {
    private static final long serialVersionUID = 1L;

```

```

Graphics dbg;
Image dbImage;
static Image block;
static Block block1 = new Block();
static Image player1;
static Player player = new Player(193, 143);

public Game() {
    Image playerIcon = new ImageIcon("res/play.png").getImage();
    setSize(500, 400);
    setTitle("Game");
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    setIconImage(playerIcon);
    setLocationRelativeTo(null);
    setVisible(true);
    addKeyListener(new InputHandler());
    setBackground(Color.BLACK);
    setResizable(false);
}

public static void main(String[] args) {
    new Game();
    Thread p = new Thread(player);
    p.start();
}

@SuppressWarnings("static-access")
public void paint(Graphics g) {
    try {
        dbImage = ImageIO.read(new File("res/background.png"));
    } catch (Exception e) {
        e.printStackTrace();
    }
    try {
        player1 = ImageIO.read(new File("res/play.png"));
    } catch (Exception e) {
        e.printStackTrace();
    }
    try {
        block = ImageIO.read(new File("res/grass.png"));
    } catch (Exception e) {
        e.printStackTrace();
    }
    dbg = dbImage.getGraphics();
    draw(dbg);
    g.drawImage(dbImage, 0, 0, this);
}

```

```

        g.drawImage(player1, player.x, player.y, this);
        g.drawImage(block, block1.x, block1.y, this);
    }

    public void draw(Graphics g) {
        repaint();
    }
}

```

**Q.39)Write a java program to accept ‘N’ Subject Names from a user store them into LinkedList Collection and Display them by using Iterator interface.**

```

import java.util.*;

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

        LinkedList<String>myList = new LinkedList<String>();

        myList.add("Red");
        myList.add("Green");
        myList.add("Blue");
        myList.add("Brown");
        myList.add("Pink");
        myList.add("Purple");

        ListIterator<String>list_it = myList.listIterator(0);
        System.out.println("Elements in the LinkedList:");
        while(list_it.hasNext())
            System.out.print(list_it.next() + " ");
    }
}

```

**Q.40)Write a java program to solve producer consumer problem in which a producer produces a value and consumer consume the value before producer generate the next value. (Hint: use thread synchronization)**

Ans)Multithreading Assignment 2 SET A Q. c

**Q.41)Write a Menu Driven program in Java for the following: Assume Employee table with attributes (ENo, EName, Salary) is already created. 1. Insert 2. Update 3. Display 4. Exit.**

```

import java.io.*;
import java.sql.*;
class Menu
{
    public static void main(String args[])
    {
        DataInputStream din=new DataInputStream(System.in);
        int rno,k,ch,per;
        String nm;
        try {
            // load a driver
            Class.forName("org.postgresql.Driver");
            // Establish Connection
            conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90", "");
            Statement st=cn.createStatement();
            do {
                System.out.println(" 1. Insert \n 2. Update \n 3. Delete \n 4. Search \n 5. Display \n
6. Exit");
                System.out.print("Enter your choice: ");
                ch=Integer.parseInt(din.readLine());
                System.out.println(".....");
                switch(ch)
                {
                    case 1:
                        System.out.print("How many records you want to inserted ? ");
                        Int n=Integer.parseInt(din.readLine());
                        for(int i=0;i<n;i++)
                        {
                            System.out.println("Enter Roll No : ");
                            rno=Integer.parseInt(din.readLine());
                            System.out.println("Enter Name : ");
                            nm=din.readLine();
                            System.out.println("Enter Percentage: ");
                            per=Integer.parseInt(din.readLine());
                            18 k=st.executeUpdate("insert into Stud values(" + rno +
",'" + nm + "','" + per +")");
                            if(k>0)
                            {
                                System.out.println("Record Inserted Successfully..!!");
                                System.out.println(".....");
                            }
                        }
                        break;
                    case 2:
                        System.out.print("Enter the Roll no: ");
                        rno=Integer.parseInt(din.readLine());

```

```

        System.out.print("Enter the Sname: ");
        nm=din.readLine();
        k=st.executeUpdate("update Stud set sname='"+ nm + "' where rno="+rno);
        if(k>0)
        {
            System.out.println("Record Is Updated..!!!");
        }
        System.out.println(".....");
        break;
    case 3:
        System.out.print("Enter the Roll no: ");
        rno=Integer.parseInt(din.readLine());
        k=st.executeUpdate("delete from Stud where rno=" +rno);
        if(k>0)
        {
            System.out.println("Record Is Deleted..!!!");
        }
        System.out.println(".....");
        break;
    case 4:
        System.out.print("Enter the Roll no Whoes search record: ");
        rno=Integer.parseInt(din.readLine());
        System.out.println(".....");
        ResultSet rs1=st.executeQuery("select * from Stud where rno=" +rno);
        while(rs1.next())
        {
            System.out.println(rs1.getInt(1) +"\t" +rs1.getString(2)+"\t"+rs1.getInt(3));
        }
        System.out.println(".....");
        break;
    case 5:
        ResultSet rs=st.executeQuery("select * from Stud");
        while(rs.next())
        {
            System.out.println(rs.getInt(1) +"\t" +rs.getString(2)+"\t"+rs.getInt(3));
        }
        System.out.println(".....");
        break;
    case 6:
        System.exit(0);
    }
}
while(ch!=6);
}
catch(Exception e)
{

```

```

        System.out.println("Error");
    }
}
}

```

**Q.42) Write a JSP program which accepts UserName in a TextBox and greets the user according to the time on server machine.**

JSP File:

```

<html>
<body>
<%
String name=request.getParameter("username");
java.util.Date d=new java.util.Date();
int hr=d.getHours();
if(hr<12)
{
    out.println("Good Morning:"+name);
}
if(hr>12 && hr<16)
{
    out.println("Good Afternoon:"+name);
}
if(hr>16)
{
    out.println("Good Evening:"+name);
}
%>
</body>
</html>

```

HTML file

```

<html>
<body>
<form action="wishuser.jsp" method="post">
<input type="text" name="username">
<input type="submit">
</form>
</body>
</html>

```

**Q.43) Write a java program to accept a String from a user and display each vowel from a String after every 3 seconds.**

Ans) Multithreading Assignment 2 Refer

**Q.44)Write a java program to accept ‘N’ student names through command line, store them into the appropriate Collection and display them by using Iterator and ListIterator interface.**

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

        LinkedList<String>myList = new LinkedList<String>();

        myList.add("Red");
        myList.add("Green");
        myList.add("Blue");
        myList.add("Brown");
        myList.add("Pink");
        myList.add("Purple");

        ListIterator<String>list_it = myList.listIterator(0);
        System.out.println("Elements in the LinkedList:");
        while(list_it.hasNext())
            System.out.print(list_it.next() + " ");
    }
}
```

**Q.45)Write a java program to scroll the text from left to right continuously.**

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

public class ScrollingText extends Applet implements Runnable
{
    String msg="Welcome to Java Programming Language ..... ";
    Thread t=null;
    public void init()
    {
        setBackground(Color.cyan);
        setForeground(Color.red);
        t=new Thread(this);
        t.start();
    }
}
```

```

public void run()
{
    char ch;
    for(;;)
    {
        try
        {
            repaint();
            Thread.sleep(400);
            ch=msg.charAt(0);
            msg=msg.substring(1,msg.length());
            msg+=ch;
        }
        catch(InterruptedException e)
        {}
    }
}

    public void paint(Graphics g)
    {
        g.drawString(msg,10,10);
    }
}

```

HTML File

```
<APPLET CODE=ScrollingText.class WIDTH=400 HEIGHT=200 > </APPLET>
```



**Q.46) Write a JSP script to accept username and password from user, if they are same then display “Login Successfully” message in Login.html file, otherwise display “Login Failed” Message in Error.html file.**

### **Index.html**

```
<html>
<head>
<title>Login Page</title>
</head>
<body>
<form action="checkdetails.jsp">
<legend>Enter User Id and Password...!!!</legend>
UserId: <input type="text" name="id" /> <br><br>
Password: <input type="text" name="pass" /> <br><br>
<input type="submit" value="Sign In!!"/>
</div>
</form>
</body>
</html>
```

### **Checkdetails.jsp**

```
<html>
<head>
<title>Check Credentials</title>
</head>
<body>
<%
    String uid=request.getParameter("id");
    String password=request.getParameter("pass");
    session.setAttribute("session-uid", uid);
    if(uid.equals("Sofiya") && password.equals("Shaikh"))
    {
        response.sendRedirect("success.jsp");
    }
    else
    {
        response.sendRedirect("failed.jsp");
    }
%>
</body>
</html>
```

### Success.jsp

```
<html>
<head><title>Success Page</title></head>
<body>
<%
    String data=(String)session.getAttribute("session-uid");
    out.println(" Login Successfully...!!!");
%>
</body>
</html>
```

### Failed.jsp

```
<html>
<head><title>Sign-in Failed Page</title></head>
<body>
<%
    String data2=(String)session.getAttribute("session-uid");
    out.println(" User Id and Password are wrong. Please try Again.");
%>
</body>
</html>
```

**Q.47)Write a JSP program to accept Name and Age of Voter and check whether he is eligible for voting or not.**

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
</head>
<body>
<form action="Slip29.jsp" method="post">
Name : <input type="text" name="name">

Age : <input type="text" name="age">

<input type="submit" value="Check">
</form>
</body>
```

</html>

JSP FILE:

```
<% @page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<%
String name = request.getParameter("name");
int age = Integer.parseInt(request.getParameter("age"));
if(age >=18)
{
out.println(name + "\nAllowed to vote");
}
else
{
out.println(name + "\nNot allowed to vote");
}
%>
```

**Q.48)Write a Java program to delete the details of given employee (ENo EName Salary). Accept employee ID through command line. (Use PreparedStatement Interface)**

```
import java.sql.*;
class Slip27_1
{
public static void main(String a[])
{
Connection con;
PreparedStatement ps;
ResultSet rs;
try
{
// load a driver
Class.forName("org.postgresql.Driver");
// Establish Connection
conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90",
"");if(con==null)
{
System.out.println("Connection Failed....");
System.exit(1);
}
System.out.println("Connection Established...");
ps=con.prepareStatement("select * from employee where eid=?");
int id = Integer.parseInt(a[0]);
ps.setInt(1,id);
```

```

rs=ps.executeQuery();
System.out.println("eno\t"+"ename\t"+"department\t"+"sal"); while(rs.next())
{
System.out.println("\n"+rs.getInt(1)+"\t"+rs.getString(2)+"\t"+rs.getString(3)+"\t"+rs.ge tInt(4));
}
con.close();
}
catch(Exception e)
{
System.out.println(e);
}
}
}

```

**Q.49)Write a Java Program to display the details of College (CID, CName, address, Year) on JTable.**

```

import java.io.*;
import java.sql.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class College extends JFrame implements ActionListener
{
    JLabel lblid, lblname, lbladdr, lblyr;
    JTextField txtid, txtname, txtaddr, txtyr;
    JButton btninsert, btnclr, btnexit;
    College()
    {
        setLayout(null);
        lblid=new JLabel("College id");
        lblname=new JLabel("College Name");
        lbladdr=new JLabel("College Address");
        lblyr=new JLabel("Year");
        txtid=new JTextField();
        txtname=new JTextField();
        txtaddr=new JTextField();
        txtyr=new JTextField();
        btninsert=new JButton("Insert");
        btnclr=new JButton("Clear");
        btnexit=new JButton("Exit");
        lblid.setBounds(20,30,100,20);
        lblname.setBounds(20,70,150,30);

```

```

lbladdr.setBounds(20,110,150,30);
lblyr.setBounds(20,150,150,30);
txtid.setBounds(120,30,150,30);
txtname.setBounds(120,70,150,30);
14    txtaddr.setBounds(120,110,150,30);
txtyr.setBounds(120,150,150,30);
btninsert.setBounds(10,200,100,50);
btnclear.setBounds(120,200,100,50);
btnexit.setBounds(230,200,100,50);
btninsert.addActionListener(this);
btnclear.addActionListener(this);
btnexit.addActionListener(this);
add(lblid); add(txtid);
add(lblname); add(txtname);
add(lbladdr); add(txtaddr);
add(lblyr); add(txtyr);
add(btninsert);
add(btnclear);
add(btnexit);
setSize(500,400);
}
public void actionPerformed(ActionEvent a)
{
    try    {
        if(a.getSource()==btninsert)
        {
            int id,yr;
            String nm,add;
            // load a driver
            Class.forName("org.postgresql.Driver");
            // Establish Connection
            conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90", "");
            PreparedStatement pst=con.prepareStatement("insert into College values(?,?,?,?)");
            id=Integer.parseInt(txtid.getText());
            nm=txtname.getText();
            add=txtaddr.getText();
            15        yr=Integer.parseInt(txtyr.getText());
            pst.setInt(1,id);
            pst.setString(2,nm);
            pst.setString(3,add);
            pst.setInt(4,yr);
            pst.executeUpdate();
            //int k=        JOptionPane.showMessageDialog(null,"Record Inserted Successfully");
            con.close();
            /*if(k>0){System.out.println("Record Inserted..!!!");}else{System.out.println("Error..!!!");}*/
        }
    }
}

```

```

        if(a.getSource()==btnclear)
        {
            txtid.setText("");
            txtname.setText("");
            txtaddr.setText("");
            txttyr.setText("");
        }
        if(a.getSource()==btnexit)
        {
            System.exit(0);
        }
    }
    catch(Exception e)
    {
        System.out.println("Error is :"+e);
    }
}
public static void main(String args[])
16 {
    new College().show();
}
}

```

**Q.50)Write a SERVLET program to change inactive time interval of session.**

```

import java.util.*;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class MyServlet extends HttpServlet
{
    public void doGet(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException
    {
        res.setContentType("text/html");
        PrintWriter out = res.getWriter();
        // Get the current session object, create one if necessary
        HttpSession session = req.getSession();
        out.println("<HTML><HEAD><TITLE>SessionTimer</TITLE></HEAD>");
        out.println("<BODY><H1>Session Timer<</H1>");
        // Display the previous timeout
        out.println("The previous timeout was " +
            session.getMaxInactiveInterval());
        out.println("<BR>");
        // Set the new timeout
    }
}

```

```

session.setMaxInactiveInterval(2*60*60); // two hours
// Display the new timeout
out.println("The newly assigned timeout is " + session.getMaxInactiveInterval());
out.println("</BODY></HTML>");
}
}

```

//xml code

```

<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
"http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">
<web-app>
<servlet><servlet-name>MyServletName</servlet-name>
<servlet-class>MyServlet</servlet-class>
</servlet>
<servlet-mapping><servlet-name>MyServletName</servlet-name>
<url-pattern>/index.html</url-pattern>
</servlet-mapping>
</web-app>
build.xml <project name="MyProject" default="compile" basedir=".">
<property name="sourcedir" value="${basedir}/src"/>
<property name="webdir" value="${basedir}/build"/> <property name="javaSourcedir"
value="${sourcedir}/WEB-INF/classes"/>
<property name="webClassdir" value="${webdir}/WEB-INF/classes"/>
<property name="webClassLib" value="${webdir}/WEB-INF/lib"/>
<property name="compileLibDir" value="${basedir}/lib"/>
<path id="libraries">
<fileset dir="${compileLibDir}">
<include name="*.jar"/>
</fileset>
</path>
<target name="clean">
<delete dir="${webdir}"/>
<mkdir dir="${webClassdir}"/>
</target>
<target name="copy">
<copy todir="${webdir}">
<fileset dir="${sourcedir}">
<exclude name="**/*.java"/>
</fileset>
</copy>
</target>
<target name="compile" depends="clean, copy">
<javac srcdir="${javaSourcedir}"
destdir="${webClassdir}"

```

```

classpathref="libraries"/>

<!-- one for deployment, another one for reference -->
<war warfile="demo.war" basedir="${webdir}" webxml="${webdir}/WEB-INF/web.xml">
<exclude name="WEB-INF/web.xml"/>
</war>
<war warfile="../demo.war" basedir="${webdir}" webxml="${webdir}/WEB-INF/web.xml">
<exclude name="WEB-INF/web.xml"/>
</war>
<delete dir="${webdir}"/>
</target>
</project>

```

**Q.51)Write a JSP script to accept a String from a user and display it in reverse order.**

Ans) Reference Servlet & JSP Assignment 4 Go through it.

**Q.52)Write a java program to display name of currently executing Thread in multithreading.**

Ans) Multithreading Assignment 2 go through it

**Q.53)Write a java program for the implementation of synchronization.**

```

class Table{
    synchronized void printTable(int n){//synchronized method
        for(int i=1;i<=5;i++){
            System.out.println(n*i);
            try{
                Thread.sleep(400);
            }catch(Exception e){System.out.println(e);}
        }
    }
}

class MyThread1 extends Thread{
    Table t;
    MyThread1(Table t){
        this.t=t;
    }
    public void run(){
        t.printTable(5);
    }
}

```



```

}
class MyThread2 extends Thread{
    Table t;
    MyThread2(Table t){
        this.t=t;
    }
    public void run(){
        t.printTable(100);
    }
}

public class TestSynchronization2{
    public static void main(String args[]){
        Table obj = new Table();//only one object
        MyThread1 t1=new MyThread1(obj);
        MyThread2 t2=new MyThread2(obj);
        t1.start();
        t2.start();
    }
}

```

**Q.54)Write a Java Program for the implementation of scrollable ResultSet. Assume Teacher table with attributes (TID, TName, Salary) is already created.**

Ans )Refer below program and change database name.

```

import java.io.*;
import java.sql.*;
import java.util.*;
class Slip25_2
{
    public static void main(String args[])
    {
        Connection conn= null;
        Statement stmt = null;
        ResultSet rs = null;
        int ch;
        Scanner s=new Scanner(System.in);
        try
        {
            // load a driver
            Class.forName("org.postgresql.Driver");
            // Establish Connection

```

```

conn = DriverManager.getConnection("jdbc:postgresql://192.168.1.21:5432/ty90", "ty90", "");
stmt = conn.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
ResultSet.CONCUR_UPDATABLE);
rs = stmt.executeQuery("select * from employee");
int count=0;
while(rs.next())
count++;
System.out.println("Which Record u want");
System.out.println("Records are = "+count);
do
{ System.out.println("1 First \n2 last \n3 next \n4 prev \n0 Exit");
ch=s.nextInt();
switch(ch)
{
case 1: rs.first();
System.out.println("Roll :"+rs.getInt(1)+" Name :"+rs.getString(2)); break;
case 2: rs.last();
System.out.println("Roll :"+rs.getInt(1)+" Name :"+rs.getString(2)); break;
case 3 : rs.next();
if(rs.isAfterLast())
System.out.println("can't move forward");
else
System.out.println("Roll :"+rs.getInt(1)+" Name :"+rs.getString(2));
break;
case 4 : rs.previous();
NR CLASSES, PUNE (8796064387/90)
if(rs.isBeforeFirst())
System.out.println("can't move backward");
else
System.out.println("Roll :"+rs.getInt(1)+" Name :"+rs.getString(2));
break;
case 0 : break;
default: System.out.println("Enter valid operation");
} //switch
} while(ch!=0);
} //end of try
catch(Exception e)
{
System.out.println(e);
}
} //main
} //class

```