AJP Practical Sample Solutions

1. **Write a program to demonstrate status of key on a Frame such as keyPressed, keyReleased, keyTyped.**

soln:

```
import java.awt.Frame;
import java.awt.Label;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;

public class keyFrame {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Frame fr=new Frame("Key Event");

        Label l=new Label("Label");
        fr.add(l);

        fr.addKeyListener(new KeyListener() {

            @Override
            public void keyTyped(KeyEvent e) {
                // TODO Auto-generated method stub
                char keychar=e.getKeyChar();
                if(keychar=='x') {
                    l.setText("x was pressed");
                }

            }

            @Override
            public void keyReleased(KeyEvent e) {
                // TODO Auto-generated method stub
                l.setText("Key released");
            }

            @Override
            public void keyPressed(KeyEvent e) {
                // TODO Auto-generated method stub
```

```
                l.setText("Key pressed");
            }
        });


        fr.setSize(500,500);
        fr.setVisible(true);

    }

}
```

2. **Write a program to create a frame using AWT. Implement mouseClicked, mouseEntered() and mouseExited() events. Frame should become invisible when the mouse exits it.**

soln:

```
import java.awt.Color;
import java.awt.Frame;
import java.awt.Label;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;

public class mouseFrame {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Frame f=new Frame("Mouse Event");

        Label la=new Label("Label");
        la.setBounds(100,100,200,80);
        f.setLayout(null);
        f.add(la);

        f.addMouseListener(new MouseListener() {

            @Override
            public void mouseReleased(MouseEvent e) {
                // TODO Auto-generated method stub

            }

            @Override
            public void mousePressed(MouseEvent e) {
                // TODO Auto-generated method stub


            }

            @Override
            public void mouseExited(MouseEvent e) {
```

```
                // TODO Auto-generated method stub
                f.setVisible(false);
            }

            @Override
            public void mouseEntered(MouseEvent e) {
                // TODO Auto-generated method stub
                la.setText("Mouse Entered frame");
                f.setBackground(Color.blue);
            }

            @Override
            public void mouseClicked(MouseEvent e) {
                // TODO Auto-generated method stub
                la.setText("Mouse Click detected");
                f.setBackground(Color.cyan);
            }
        });


        f.setSize(500,500);
        f.setVisible(true);

    }

}
```

3. **Develop a GUI which accepts the information regarding the marks for all the subjects of a student in the examination. Display the result for a student in a separate window.**

soln:

```
import java.awt.Button;
import java.awt.Frame;
import java.awt.Label;
import java.awt.TextField;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.text.Format;

public class guiDemo {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        Frame fr=new Frame("Student Marks");

        Label l1,l2,l3,l4,l5,l6;
        TextField tf1,tf2,tf3,tf4,tf5,tf6;
        Button b1,b2;

        l1=new Label("Name of Student: ");
        l1.setBounds(50, 50, 100, 30);

        tf1=new TextField();
        tf1.setBounds(150, 50, 130, 30);

        l2=new Label("Class: ");
        l2.setBounds(350, 50, 80, 30);

        tf2=new TextField();
        tf2.setBounds(450, 50, 130, 30);

        l3=new Label("Subject 1 Marks: ");
        l3.setBounds(50, 100, 100, 30);
```

```
        tf3=new TextField();
        tf3.setBounds(150, 100, 50, 30);

        l4=new Label("Subject 2 Marks: ");
        l4.setBounds(350, 100, 100, 30);

        tf4=new TextField();
        tf4.setBounds(450, 100, 50, 30);

        l5=new Label("Subject 3 Marks: ");
        l5.setBounds(50, 150, 100, 30);

        tf5=new TextField();
        tf5.setBounds(150, 150, 50, 30);

        l6=new Label("Subject 4 Marks: ");
        l6.setBounds(350, 150, 80, 30);

        tf6=new TextField();
        tf6.setBounds(450, 150, 50, 30);

        b1=new Button("Result");
        b1.setBounds(150, 200, 80, 30);
        b2=new Button("Close");
        b2.setBounds(350, 200, 80, 30);

        b1.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                int s1=Integer.parseInt(tf3.getText());
                int s2=Integer.parseInt(tf4.getText());
                int s3=Integer.parseInt(tf5.getText());
                int s4=Integer.parseInt(tf6.getText());

                int sum=s1+s2+s3+s4;
                String total=Integer.toString(sum);
                Label res=new Label("The result for " +
tf1.getText() + " is " + total);
```

```
                Frame fr2=new Frame("Result");
                fr2.add(res);
                fr2.setSize(500,200);
                fr2.setVisible(true);
            }
        });

        b2.addActionListener(new ActionListener() {

            @Override
            public void actionPerformed(ActionEvent e) {
                // TODO Auto-generated method stub
                fr.dispose();
            }
        });

        fr.add(l1);
        fr.add(tf1);
        fr.add(l2);
        fr.add(tf2);
        fr.add(l3);
        fr.add(tf3);
        fr.add(l4);
        fr.add(tf4);
        fr.add(l5);
        fr.add(tf5);
        fr.add(l6);
        fr.add(tf6);
        fr.add(b1);
        fr.add(b2);
        fr.setLayout(null);
        fr.setSize(800,400);
        fr.setVisible(true);

    }

}
```

4. **Write a program to insert and update the data in a database using JDBC.**

   Refer video on link: https://youtu.be/Yi4Yunh8tkc

5. **Develop an RMI application which accepts a string or a number and checks that string or number is palindrome or not.**

soln:

a. one.java

```
import java.rmi.*;



public interface one extends Remote

{

public int palin(String a) throws RemoteException;

}
```

b. two.java

```
import java.rmi.*;

import java.lang.*;

import java.rmi.server.*;

public class two extends UnicastRemoteObject implements one{
```

```
public two() throws RemoteException { }

public int palin(String a) throws RemoteException

{

System.out.println("Hello");

StringBuffer str = new StringBuffer(a);

String str1 = str.toString();

System.out.println("Print : " + str1.toString());

StringBuffer str2 = str.reverse();

System.out.println("Print : " + str2.toString());

int b = str1.compareTo(str2.toString());

System.out.println("Print : " + b);

if (b == 0)

return 1;

else



return 0;

}

}
```

c. rmiserver.java

```
mport java.io.*;

import java.lang.invoke.MethodHandles.Lookup;

import java.rmi.*;

import java.rmi.registry.*;

import java.rmi.server.*;



import java.net.*;

public class rmiserver

{

public static void main(String args[]) throws Exception

{

try

{

two twox = new two();



Registry registry = LocateRegistry.createRegistry(1880);



String name="palin";
```

```
registry.rebind(name, twox);

System.out.println("Object registered");



}



catch(Exception e)

{

System.out.println("Exception" + e);

}

}

}
```

d. rmiclient.java

```
import java.io.*;

import java.lang.invoke.MethodHandles.Lookup;// for printing reference of a

name in registry

import java.rmi.*;

import java.rmi.registry.*;
```

```
import java.net.*;

public class rmiclient

{

public static void main(String args[]) throws Exception

{

try

{

String name = "palin";

Registry registry = LocateRegistry.getRegistry(1880);

one pal = (one)registry.lookup(name);

int m = pal.palin("madam");

System.out.println("Print : " + m);

if (m == 1)



{

System.out.println("The given string is a Palindrome");

}

else

{
```

```
System.out.println("The given string is not a Palindrome");

}

}

catch (Exception e)

{

System.out.println("Exception" + e);

}

}

}
```

6. **Write a program to demonstrate the use of InetAddress class and its factory methods.**

soln: Refer lab manual

7. **Write the Servlet code (for server side and client side) to display the username and password accepted from the client.**

soln: Refer lab manual and video on link:
https://www.youtube.com/watch?v=v1M5BgYWays

Remember to add web.xml file and code

8. **Write a program to retrieve the data from a database using JDBC.**
   Refer video on link: https://youtu.be/Yi4Yunh8tkc

9. **Write a simple JSP page to display current Date and Time.**

soln: Refer lab manual

10. **Create a simple Calculator application using servlet.**

soln: Refer below code and video on link:
https://www.youtube.com/watch?v=v1M5BgYWays for procedure

a. CalculatorServlet.java

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

```
public void doGet(HttpServletRequest req,
HttpServletResponse res) throws ServletException {
int result = 0;
try {
String number1 = req.getParameter("num1");
String number2 = req.getParameter("num2");
String operator = req.getParameter("opr");
int x = Integer.parseInt(number1);
int y = Integer.parseInt(number2);
if(operator.equals("+")) {
result = x + y;
}
else if(operator.equals("-")) {
result = x - y;
}
else if(operator.equals("*")) {
result = x * y;
}
else if(operator.equals("/")) {
result = x/y;
}
PrintWriter p = res.getWriter();
p.println("<h1> Result= "+ result);
}
catch(Exception e) {}
}
}
```

b. Index.html

```
<html>
<head>
<meta charset="ISO-8859-1">
<title>Calculator Application Using Servlet</title>
</head>
<body>
<form method=get action="./MyURL" >
Enter First Number <input type="text" name="num1"><br>
```

```
Enter Second Number <input type="text" name="num2" ><br>
Select an Operation<input type="radio" name="opr"
value="+">
ADDTION <input type="radio" name="opr" value="-">
SUBSTRACTION <input type="radio" name="opr" value="*">
MULTIPLY <input type="radio" name="opr" value="/">
DIVIDE <br><input type="reset">
<input type="submit" value="Calculate" >
</form>
</body>
</html>
```

c. web.xml

```
<web-app>

 <servlet>
 <servlet-name>calc</servlet-name>
 <servlet-class>calc.CalculatorServlet</servlet-class>
 </servlet>

 <servlet-mapping>
 <servlet-name>calc</servlet-name>
 <url-pattern>/MyURL</url-pattern>
 </servlet-mapping>

</web-app>
```