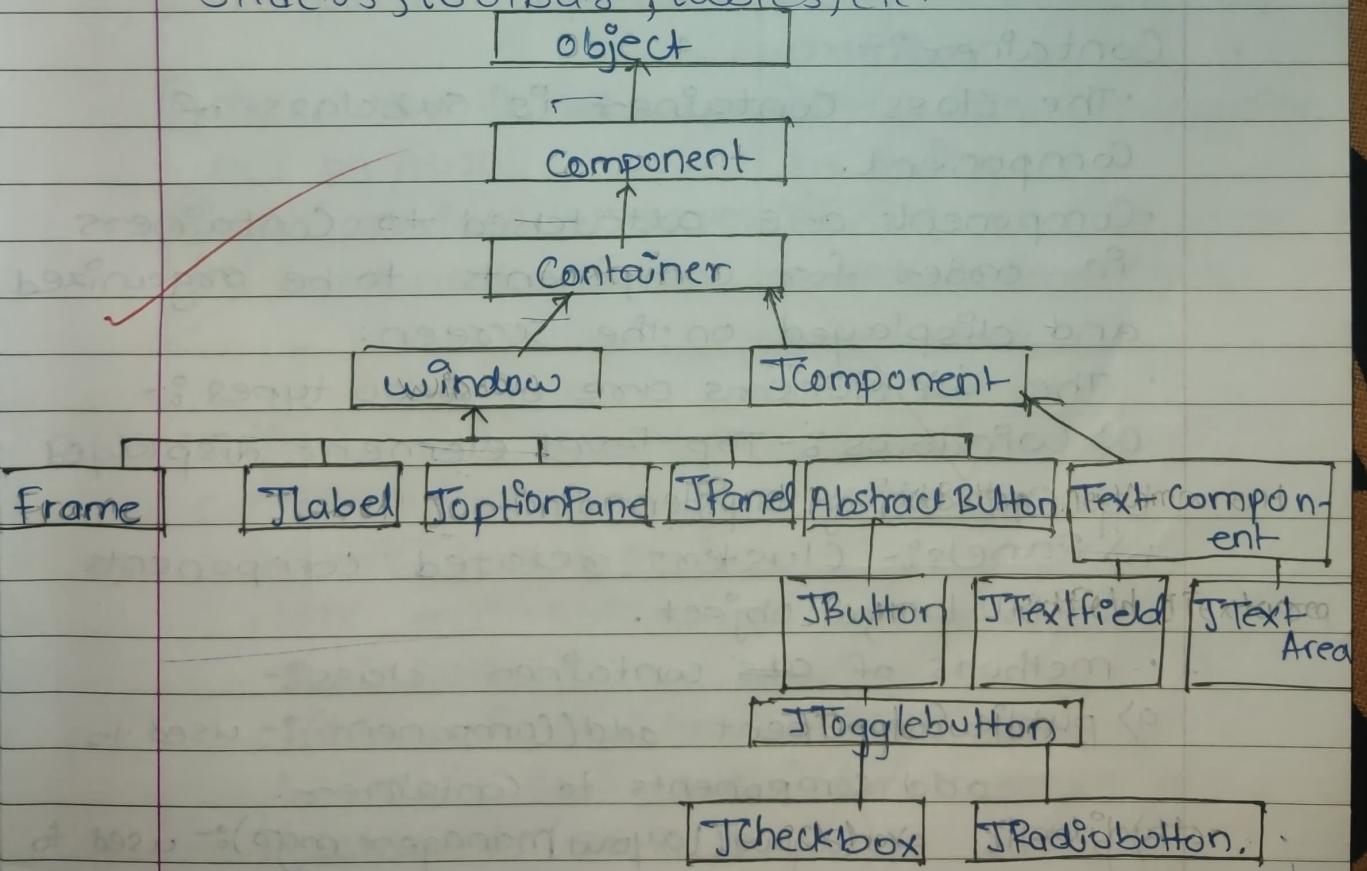


Assignment no. 2

Q.1) What is AWT and Swing? State its hierarchy?

- • **AWT :-** AWT refers to Abstract Window Toolkit. It refers to a collection of application program interfaces (API) that are utilized by Java programmers for creation of Graphical user Interface (GUI) objects. The objects are in the form of scrollbars, windows, buttons, etc.
- **Swing :-** Swing in Java refers to graphical user interface in the form of lightweight widget toolkit; the toolkit is packaged with widgets with rich functionality. The built-in controls in swing comprise of images, buttons, trees, tabbed panes, color choosers, sliders, toolbar, tables, etc.



Q.2) what is meant by component and Container?

→ Component :-

- The Component is a subclass of class Object, which declares many attributes and behaviour common to GUI components.
- The class component implements useful public methods, which are inherited by subclasses, such as :-

- public void setSize(int width, int height); - used to set size of component.
- public void setBackground(Color c); - used to set background color of component.
- public void setVisible(boolean b); - used to set visibility of a component.

Container :-

- The class Container is subclass of Component.
- Components are attached to Containers in order for components to be organized and displayed on the screen.
- The Containers are of two types :-
- a) Windows :- Top level element displayed by operating system.
- b) Panels :- Clusters related components within large object.
- methods of ~~the~~ container class :-
- a) public Component add(Component); - used to add components to container.
- b) public void setLayout(LayoutManager mg); - used to set layout of container.

Q.3)
→

Differentiate between Java AWT & Java swing.

Java AWT

Java swing

- | | |
|---|--|
| ① Java AWT is an API to develop GUI applications in Java. | ① swing is a part of Java Foundation Classes and is used to create graphical applications. |
| ② Components of Java AWT are heavy weighted. | ② Components of Java Swing are light weighted. |
| ③ AWT has less functionality as compared to swing. | ③ swing has more functionality as compared to AWT. |
| ④ Execution time of AWT is more than swing. | ④ Execution time of swing is less than AWT. |
| ⑤ Components of Java AWT are platform dependent. | ⑤ Components of Java swing are platform independent. |
| ⑥ MVC pattern is not supported by AWT. | ⑥ MVC pattern is supported by swing. |
| ⑦ AWT provides less powerful components. | ⑦ swing provides more powerful components. |

Q.4)

Write down a simple Java program containing at least four AWT components.

→

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

```
class mainAWT { public static void main(String args[]) {
    mainAWT();
}

Frame f = new Frame("Employee Login");
Label l1 = new Label("Employee ID");
l1.setBounds(20, 80, 80, 30);
```

```
textfield tf1 = new textfield();  
tf1.setBounds(20, 100, 80, 30);  
label l2 = new label("Password");  
l2.setBounds(30, 120, 80, 30);  
textfield tf2 = new textfield();  
tf2.setBounds(40, 130, 80, 30);  
button b = new button("Login");  
b.setBounds(100, 100, 80, 30);  
f.add(l1);  
f.add(tf1);  
f.add(l2);  
f.add(tf2);  
f.add(b);  
f.setSize(400, 300);  
f.setLayout(null);  
f.setVisible(true);  
}  
}
```

Output :-

~~Employee Login~~ - | P | x

Employee Login

- | | X

Employee ID:

EMP

Password:

Q.5) Write down a simple Java program containing at least 4 swing components.

→

```

import javax.swing.*;
import java.awt.*;
public class Swing {
    public static void main (String args[]) {
        JFrame f = new JFrame ("Swing Frame");
    }
}
```

~~JLabel l1 = new JLabel ("Name");~~

~~JLabel l2 = new JLabel ("Roll no.");~~

~~l1.setBounds (10, 120, 200, 300);~~

~~l2.setBounds (20, 130, 200, 300);~~

~~JTextField tf1 = new JTextField ();~~

~~tf1.setBounds (30, 140, 200, 300);~~

~~JTextField tf2 = new JTextField ();~~

~~tf2.setBounds (40, 150, 200, 300);~~

~~JButton b = new JButton ("Present");~~

~~b.setBounds (100, 160, 100, 100);~~

~~f.add (l1); f.add (l2); f.add (tf1);~~

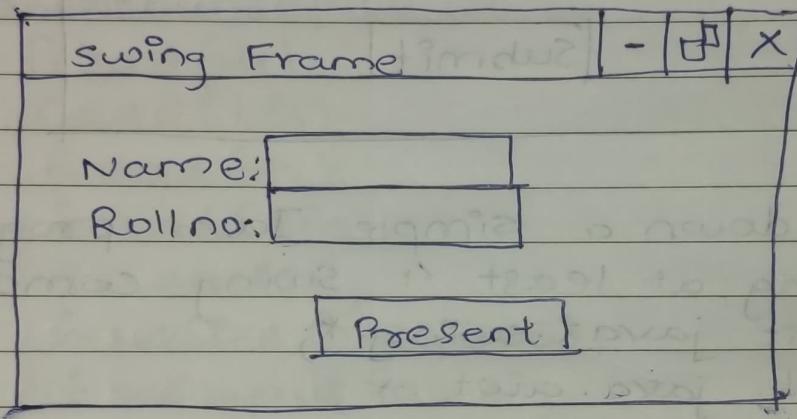
~~f.add (tf2); f.add (b);~~

```

f.setSize(300,300);
f.setLayout(null);
f.setVisible(true);
    }
}

```

Output :-



Q.6) What are different types of containers and components in swing?

→ Java Swing Components & containers :-

- 1) A component is an independent visual control.
- 2) Swing framework contains a large set of components which provide rich functionalities and allows high level of customization.
- 3) They all are derived from JComponent.
- 4) All these components are lightweight components.
- 5) This class provides some common functionality like pluggable look & feel.

support for accessibility, drag layout, etc.

6) A container holds a group components. It provides a space where a component can be managed and displayed.

7) Containers are two types:-

- Top level containers:-

- a) It inherits components and containers of AWT.

- b) It cannot be contained within other containers.

- c) Heavyweight.

- d) Ex. JFrame, JDialog, JApplet.

- Lightweight Containers:-

- a) It inherits JPanel class.

- b) It is a general purpose container.

- c) It can be used to organize related components together.

- d) Ex. JPanel.

Q. 7) Write down a Java program for GUI based Input/Output with JOptionPane.

→ import javax.swing.JOptionPane;

public class Addition

{

public static void main (String args []) {

String firstNumber =

JOptionPane.showInputDialog ("Enter first number");

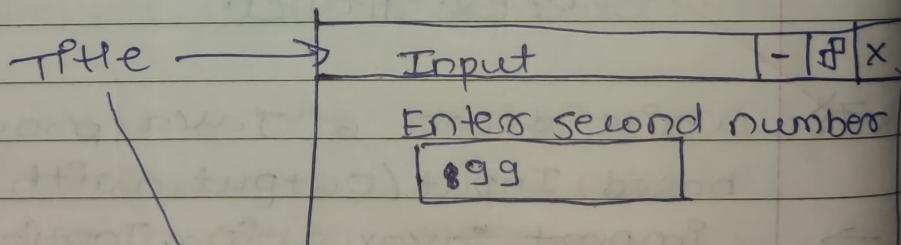
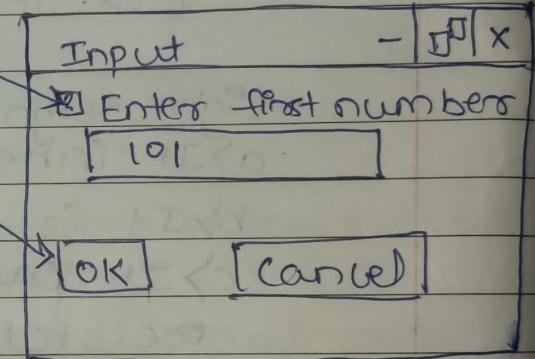
String secondNumber =

```
JOptionPane.showInputDialog ("Enter second number")
int number1 = Integer.parseInt (FirstNumber);
int number2 = Integer.parseInt (SecondNumber);
int sum = number1 + number2;
```

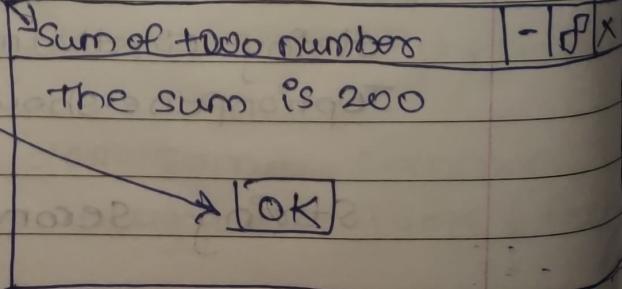
```
JOptionPane.showMessageDialog (null, "The sum is",
sum, "Sum of two integers", JOptionPane
PLAIN_MESSAGE);
}
```

Output :-

Prompt to user
when OK clicked showInputDialog
Dialog returns to program
101 typed by user as string.
The program must convert
string into an int.



When user clicks OK,
message dialog is
dismissed



Q. 8) How to display text and messages in a window? Write down a simple Java program to enter text and image in window.

-
- JLabel is a class of java.swing.JLabel is used to display a short string or image icon.
 - JLabel can be display text, image or both.
 - JLabel is only a display of text or image and it cannot get focus.
 - JLabel is inactive to input events such as mouse focus or keyboard focus.
 - By default Label are vertically centered but the user can change alignment of label.
 - Constructors of the class are:-

JLabel :

JLabel (String s)

JLabel (Icon i)

JLabel (String s, Icon i, int align).

- methods of the class are:-

getIcon ()

setIcon (Icon i)

getText ()

setText (String s)

- program:-

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

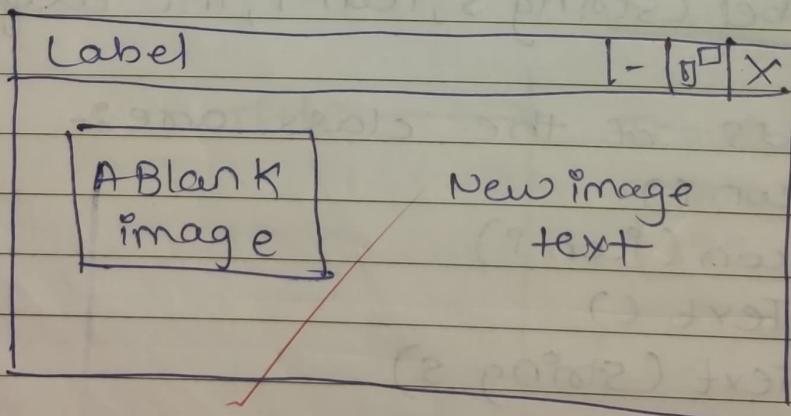
```
import java.awt.*;
```

```
import javax.swing.*;
```

```
class text extends JFrame {
```

```
Static JFrame f;
Static JLabel l;
text()
{
}
public static void main (String args[])
{
    f = new JFrame ("Label");
    ImageIcon i = new ImageIcon ("f:/image.png");
    Constant.HORIZONTAL;
    JPanel p = new JPanel ();
    p.add (i);
    f.add (p);
    f.setSize (800,800);
    f.show ();
}
```

Output:-



g) What is event handling in Java? How Event handling works?

- • Event handling is the mechanism that controls the event and decides what should happen if an event occurs. This mechanism has a code which is known as an event handler, that is executed when event occurs.
- Java uses Delegation Event model to handle events. This model defines the standard mechanism to generate and handle events.
- Delegation Event model has following key participants:-

a) ~~Source~~- The source is an object on which event occurs. Source is responsible for providing information of the occurred event to its handler. Java provides us with classes for source object.

~~b) Listener :-~~

b) ~~Listener :-~~ The listener is responsible for generating response to an event. Listener waits till it receives an event.

Once event is received, the listener processes event and then returns.

• Steps involved in Event Handling:-

Step 1 :- The user clicked on mousebutton and then event generated.

Step 2 :- The object of concerned event class is created automatically and information about source and event get populated within same object.

Step 3 :- Event object is forwarded to the method of registered Listener class.

Step 4 :- The methods get executed and returns.

Q.10) What are the common GUI types?

→ There are two types of GUI elements:-

1) Components :- Components are elementary GUI entities, such as Button, Label, and Textfield.

2) Containers :- Containers, such as Frame and Panel, are used to hold components in a specific layout. A container can also hold sub-containers.

• A Frame is a top-level containers of an AWT program. A Frame has title bar (containing an icon, title, and minimize/maximize/close buttons), an optional menu bar and content display area.

• A Panel is a rectangular area used to group related GUI components in a certain layout.

~~There are five components :-~~

~~1) Label :- provides~~

• In GUI program, a component must be kept in containers. You need to identify a container to hold component. Every comp container has method called add(Component c).

• GUI components are also called controls

which allow users to interact with the application.

Q.11) Define / describe various event listeners used in swing.



i) ActionListeners:-

This interface is used for receiving the option event.

ii) ComponentListeners:-

Used for receiving component event.

iii) ItemListeners:-

used for receiving item events.

iv) KeyListeners:-

used for receiving key events.

v) MouseListeners:-

(Time) used for receiving mouse events.

vi) WindowListeners:-

used for receiving window event.

vii) AdjustmentListeners:-

used for receiving adjustment event

viii) ContainerListeners:-

used for receiving container event.

ix) MouseMotionListeners:-

used for receiving mouse motion events.

x) FocusListeners:-

used for receiving focus events.