



---

## OBJECT ORIENTED PROGRAMMING WITH JAVA

### Assignment9

TYPE OF QUESTION: MCQ

Number of questions: 15

Total mark:  $15 \times 1 = 15$

---

#### **QUESTION 1:**

Which package provides many methods for graphics programming?

- a. java.awt
- b. java.Applet
- c. java.Graphics
- d. java.io

**Correct Answer: a**

#### **Detailed Solution:**

There is no package like java.Graphics and java.Applet. The Graphics class and other classes, which are necessary for GUI programming is defined in java.awt package.

---

#### **QUESTION 2:**

In which of the following package, the class Applet is defined?

- a. java.awt
- b. java.applet
- c. javax.swing
- d. javax.jws

**Correct Answer: b**

#### **Detailed Solution:**

The Applet class is defined in java.applet package.

---

#### **QUESTION 3:**

Which of the following steps are must for any kind of event handling?

- a. Implement the listener method(s) to receive and process event related notifications



- b. Adding mouse listener in the init() method.
- c. Register the source(s) to receive notifications about specific type of event(s).
- d. Adding mouse motion listener in the main() method .

**Correct Answer:a, c**

**Detailed Solution:**

There are two basic steps so far the event handling is concerned: 1) Register all the sources of events and 2) implement the listener methods related to the event(s). The (b) and (d) are related to dealing with handling mouse event.

---

**QUESTION 4:**

Which of the following is/ are not related with keyboard event?

- a. ClassKeyEvent is used to notify if any key pressed occurs or not.
- b. The KeyListener should be added either in init() or the main method.
- c. RequestFocus should be added either in init() or the main method.
- d. The class InputEvent should be imported explicitly in the program.

**Correct Answer:d**

**Detailed Solution:**

The InputEvent class is the super class of all the sub classes dealing with events from different sources. When, we import java.awt.\*, we import it automatically. No need to import it explicitly.

---

**QUESTION 5:**

Which of the following package supports event handling routine, when we use Swing components in GUI programs?

- a. Javax.swing
- b. Java.awt
- c. Java.util
- d. Java.applet

**Correct Answer:b**

**Detailed Solution:**

To support GUI programming, Java provides two packages: java.awt and javax.swing. Whatever be the GUI components, event handling classes are defined in java.awt.

---

### **QUESTION 6:**

Which of the following displays components row-by-row in the order in which they were added to the JFrame?

- a. CardLayout
- b. FlowLayout
- c. BorderLayout
- d. GridLayout

**Correct Answer: b**

### **Detailed Solution:**

Flow layout adds the components serially in the order in which they were added.

---

### **QUESTION 7:**

Using a FlowLayout manager, which is the correct way to add elements to a container?

- a. add (component);
- b. add("Center", component);
- c. add(x, y, component);
- d. set(component);

**Correct Answer:a**

### **Detailed Solution:**

add() - Other options are not valid in the context of FlowLayout manager.

---

### **QUESTION 8:**

What will be the output produced by the following piece of code?

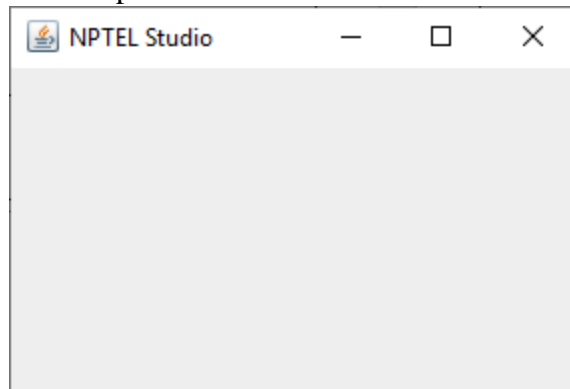
```
public class MyWindow {
    public static void main(String args[]){
        JFrame myWindow = new JFrame();
        myWindow.setSize(300, 200);
        myWindow.setTitle("NPTEL Studio");
        JLabel myLabel;
        JButton myButton = new JButton("Exit");
        myWindow.setVisible(true);
    }
}
```

- a. A window with a title and a button.
- b. Only a window.
- c. Only a button.
- d. Only a window with a title and without any button.

**Correct Answer: d**

**Detailed Solution:**

Below is the screen-shot of the output.



**QUESTION 9:**

Which layout manager places components in one of five regions: north, south, east, west, and center?

- a. CardLayout
- b. GridLayout
- c. BorderLayout
- d. FlowLayout

**Correct Answer:c**

**Detailed Solution:**

This is the layout that the BorderLayout does.

**QUESTION 10:**



The `MouseListener` interface specifies the following methods: `mouseClicked`, `mouseEntered`, `mouseExited`, `mousePressed`, and `mouseReleased`. In order to use a `MouseListener` in your code, you must provide an implementation for which of the following?

- a. None of the `MouseListener` methods.
- b. All `MouseListener` methods.
- c. At least the `mouseClicked` method.
- d. Any one `MouseListener` method.

**Correct Answer:b**

**Detailed Solution:**

You should implement all the methods, otherwise, the corresponding event for which the method(s) is (are) not implemented will get un-notified.

---

**QUESTION 11:**

Which class is used to create a pop-up list of items from which the user may choose?

- a. `JList`
- b. `JChoice`
- c. `JLabels`
- d. `JCheckbox`

**Correct Answer: b**

**Detailed Solution:**

The `JChoice` component creates a pop-up list.

---

**QUESTION 12:**

Consider a subclass of `JFrame` that implements `MouseListener`. Assume that the class has five instance data, `int x1, x2, y1, y2`, and `boolean inside`. The four `int` values represent the two end-points of a box (`x1, y1` is the upper left hand point and `x2, y2` is the lower right hand point). Which of the following properly defines code that will determine whenever the mouse button is clicked if the mouse is currently inside this box or not. If the mouse is inside the box, `inside` is set to `true`, otherwise it is set to `false`.

- a. 

```
public void mouseMoved(MouseEvent me){  
    if(me.getX() >= x1 && me.getX() <= x2 && me.getY() >= y1 && me.getY() <= y2)  
        inside = true;  
}
```

```
        else
            inside = false;
        }
    b. public void mousePressed(MouseEvent me) {
        if(me.getX()>=x1 &&me.getX()<=x2 &&me.getY()>=y1 &&me.getY()<=y2)
            inside = true;
        else
            inside = false;
        }
    c. public void mouseReleased(MouseEvent me){
        if(me.getX()>=x1 &&me.getX()<=x2 &&me.getY()<=y1 &&me.getY()>=y2)
            inside = true;
        else
            inside = false;
        }
    d. public void mouseEntered(MouseEvent me){
        if(me.getX()>=x1 &&me.getX()<=x2 &&me.getY()>=y1 &&me.getY()<=y2)
            inside = true;
        else
            inside = false;
        }
    }
```

**Correct Answer: c**

**Detailed Solution:**

Once the MouseReleased event takes place, then only it will notified the event and hence its position where the event takes place.

---

**QUESTION 13:**

How many frames will be displayed according to the following program?

```
import javax.swing.*;
public class Test{
    public static void main(String[] args){
        JFrame f1 = new JFrame("My Frame");
        JFrame f2 = f1;
        JFrame f3 = f2;
        f1.setVisible(true);
        f2.setVisible(true);
        f3.setVisible(true);
    }
}
```

- a. 0
- b. 1
- c. 2
- d. 3



---

**Correct Answer: b**

**Detailed Solution:**

Although there are three frames declared, eventually they refer to only one frame.

---

**QUESTION 14:**

To set a FlowLayout in a panel, sayjp, which of the following method(s) that you can use?

- a. `jp.setLayout(new FlowLayout());`
- b. `jp.setLayout(new FlowLayout(FlowLayout.CENTER));`
- c. `jp.setLayout(new FlowLayout(FlowLayout.center));`
- d. `jp.setLayout(FlowLayout());`

**Correct Answer: a, b**

**Detailed Solution:**

(c) and (d) are not valid according to the syntax.

---

**QUESTION 15:**

Analyze the following code.

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

public class Test {
    public static void main(String[] args) {
        JFrame frame = new JFrame("My Frame");
        frame.add(new JButton("OK"));
        frame.add(new JButton("Cancel"));
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setSize(200, 200);
        frame.setVisible(true);
    }
}
```

Which button will be displayed?

- a. OK
- b. Cancel
- c. both
- d. None

**Correct Answer: b**



---

**Detailed Solution:**

By default, the layout of the content pane in a JFrame is BorderLayout. Button OK is placed in the center of content pane, then button Cancel is placed in the same place. So you only can see button Cancel.

---

\*\*\*\*\*END\*\*\*\*\*