KURSUS

TTTK1143 (REKABENTUK ATUCARA DAN PENYELESAIAN MASALAH)

SEMESTER 2 2020/2021

1CS5

TAJUK

TUTORIAL 5

GUI & EVENT HANDLING

PENSYARAH

PROF. DR. ZARINA SHUKUR

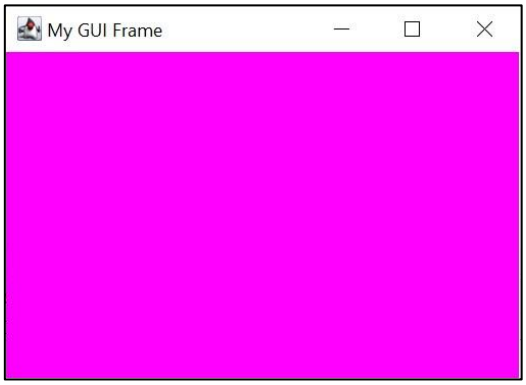
NAMA

NUR SYAFIQAH IZZATI BINTI SHA'ABAN(A179571)

TK1143 - Tutorial 5 Graphical User Interface (GUI)
Section A

Topic: JFrame and ContentPane.

1. Complete the following class (*L1-L3, L5-L6* and *L10, L12-L14*) to create a Java frame whose title is "My GUI Frame", width is 350, height is 250, and content pane's background color is magenta (as shown below).



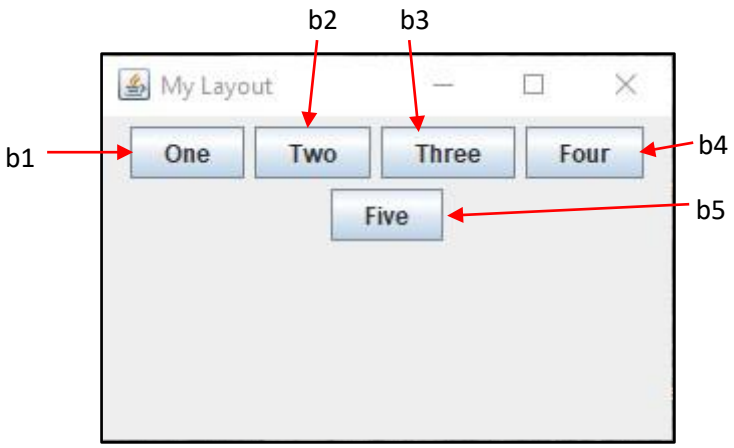

```
L1 import java.awt. *; //import package awt
L2 import java.swing.*; //import package swing
L3 class MyFrame extends JFrame {
L4      public  MyFrame () {
L5          Container pane = getContentPane();
L6          pane.setBackground(Color.Magenta); //set background
L7      }
L8      public static void main(String[] args)
L9      {
L10      MyFrame frame = new MyFrame();    //create frame
L11          frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
L12      frame.setTitle("My GUI Frame "); //set title
L13      frame.setSize(350,250);//set size
L14      frame.setVisible(True);//set visible
L15     }
L16 }
```

Topic: Layout Manager, GUI Component and JPanel

2. Name one Java Swing components that can be used for the following purposes:

Purpose	Component
a. Displays text on the Java form but cannot receive any input.	JLabel
b. User can choose only one from many options	JRadioButton
c. Displays lines of text that can be chosen by user	JList
d. A component that hides text from user.	JPasswordField
e. Display information, warning or input	JOptionPane
f. User can select more than one items from many items to choose.	JCheckBox
g. User can key in only one line of text inside it	JTextField
h. Provide a list of items from which the user can make a single selection.	JComboBox
i. It has a label and generates an event when pressed.	JButton

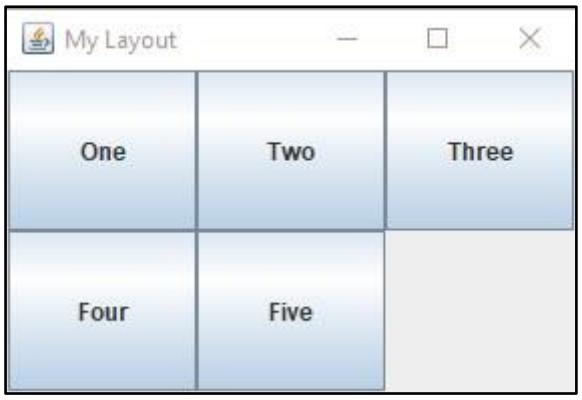
3. Complete the line of statement based on the following figure.




```
L1   Import java.awt*;
L2   Import java.swing.*;
L3
L4   class MyLayout extends JFrame {
L5   private JButton b1, b2, b3, b4, b5;//Declare component
L6
L7      public MyLayout() {
L8        Container pane = getContentPane();
L9      pane.setLayout(new FlowLayout()); //Set Layout
L10       b1=new JButton("One");  //create b1
L11       b2 = new JButton("Two"); //create b2
L12        b3 = new JButton("Three"); //create b3
L13       b4 = new JButton("Four");//create b4
L14      b5 = new JButton("Five"); //create b5
L15       pane.add(b1); //Add b1
L16        pane.add(b1);//Add b2
```

```
L17       pane.add(b3); //Add b3
L18       pane.add(b4); //Add b4
L19       pane.add(b5); //Add b5
L20     }
L21
L22     public static void main(String[] args) {
L23       MyLayout frame = new MyLayout();
L24       frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
L25         frame.setTitle("My Layout");
L26       frame.setSize(300, 200);
L27       frame.setVisible(true);
L28   }                                    L29 }
```

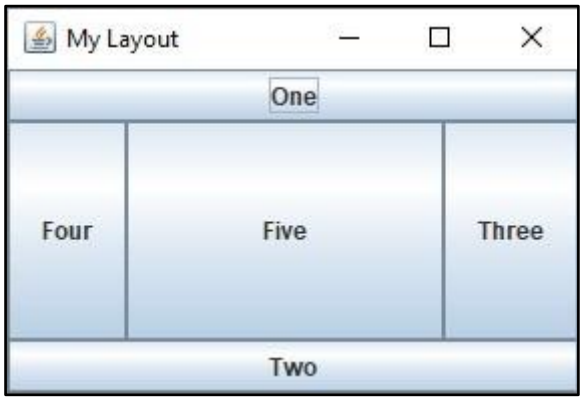
4. Modify the line of statement of *Question 3 in L9* to set this Layout in Container (as shown below).




```
import java.awt.*;
import javax.swing.*;
    class MyLayout extends JFrame {
    JButton b1, b2, b3, b4, b5; //Declare component
    public MyLayout() {
    Container pane = getContentPane();
    pane.setLayout(new GridLayout(2,3)); //Set Layout
          b1 = new JButton("One"); //create b1
          b2 = new JButton("Two"); //create b2
          b3 = new JButton("Three"); //create b3
          b4 = new JButton("Four"); //create b4
          b5 = new JButton("Five"); //create b5
          pane.add(b1); //Add b1
          pane.add(b2); //Add b2
          pane.add(b3); //Add b3
          pane.add(b4); //Add b4
          pane.add(b5); //Add b5
}
```

```
        public static void main(String[] args) {
        MyLayout frame = new MyLayout();
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setTitle("My Layout");
                frame.setSize(300, 200);
                frame.setVisible(true);
    }
}
```

5. a) Modify the line of statement of *Question 3 in L9, L15-L19* based on the following figure.




```
import java.awt.*;
import javax.swing.*;
class MyLayout extends JFrame {
       JButton b1, b2, b3, b4, b5; //Declare component
       public MyLayout() {
              Container pane = getContentPane();
              pane.setLayout(new BorderLayout()); //Set Layout
              b1 = new JButton("One"); //create b1
              b2 = new JButton("Two"); //create b2
              b3 = new JButton("Three"); //create b3
              b4 = new JButton("Four"); //create b4
              b5 = new JButton("Five"); //create b5
              pane.add((b1), BorderLayout.NORTH); //Add b1
              pane.add((b2), BorderLayout.SOUTH); //Add b2
              pane.add((b3), BorderLayout.EAST); //Add b3
              pane.add((b4), BorderLayout.WEST); //Add b4
              pane.add((b5), BorderLayout.CENTER); //Add b5
       }
       public static void main(String[] args) {
              MyLayout frame = new MyLayout();
              frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
              frame.setTitle("My Layout");
              frame.setSize(300, 200);
              frame.setVisible(true);
       }
                     }
```

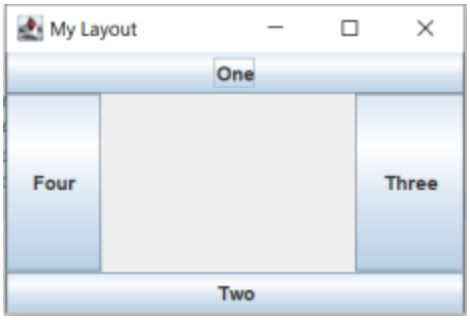
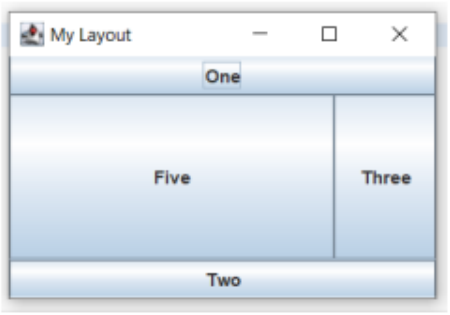
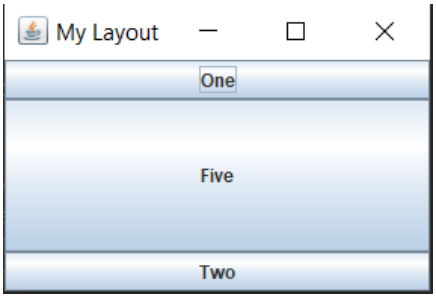
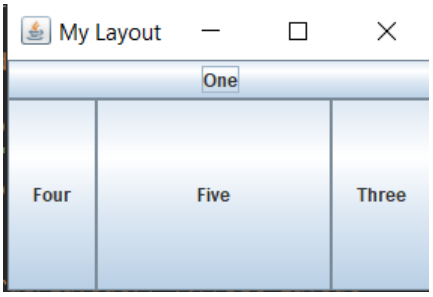
```
L9     _____________________________ //Set Layout

L15 _____________________________ //Add b1

L16 _____________________________ //Add b2

L17 _____________________________ //Add b3

L18 _____________________________ //Add b4

L19 _____________________________ //Add b5
```

b) Based on *Question 5a)* sketch the output if we remove or comment statement(s)

(i) L19

(ii) L18

(iii) L17 & L18 (iv) L6

Remove/comment L19	Remove/comment L18
	
Remove/comment L17 & L18	**Remove/comment L16**
	

6. Consider the following statements.

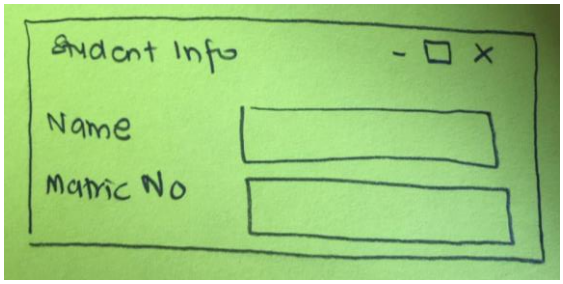
```
L1    import javax.swing.*;
L2    import java.awt.*;
L3
L4    public class TextField extends JFrame {
L5     private JLabel name, matric;
L6     private JTextField textName, textMatric;
L7
L8     public TextField()
L9     {
L10          Container pane = getContentPane();
L11          pane.setLayout(new GridLayout(2,1));
L12

L13          name = new JLabel("Name");
L14          textName = new JTextField(20);
L15          matric = new JLabel("Matric No");
L16          textMatric = new JTextField(20);
L17          pane.add(name);
L18          pane.add(textName);
L19          pane.add(matric);
L20          pane.add(textMatric);
L21      }
L22
L23      public static void main(String [] args) {
L24          TextField frame = new TextField();
L25          frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
L26          frame.setTitle("Student Info");
L27          frame.setSize(300, 100);
L28          frame.setVisible(true);
L29      }
L30   }
```

a) How many components display in the frame? List the name and type of components involve based on the following statements.

Four:

- name: JLabel
- matric: JLabel
- textName: JtextField
- textMatric: JTextField

b) Sketch the expecting output



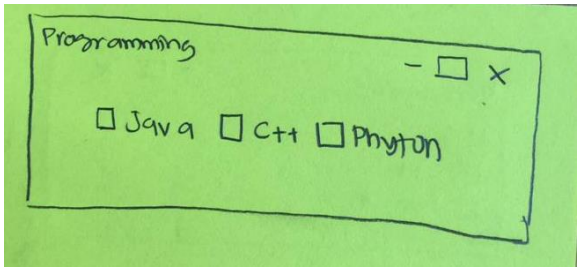
7. Consider the following statements.

```
L1 import javax.swing.*;
L2 import java.awt.*;
L3
L4 public class CheckBox extends JFrame {
L5      private JCheckbox java, c, python;
L6
L7      public CheckBox()
L8      {
L9            Container pane = getContentPane();
L10           pane.setLayout(new FlowLayout());
L11
L12           java = new JCheckBox("Java");
L13           c = new JCheckBox("C++");
L14           python = new JCheckBox("Python");
L15           pane.add(java);
L16           pane.add(C);
L17           pane.add(python);
L18     }
L19
L20     public static void main (String [] args) {
L21          CheckBox Jframe = new CheckBox ();
L22          frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
L23          frame.setTitle("Programming");
L24          frame.setSize(300, 100);
L25          frame.setVisible();
L26     }
L27     }
```

a) Trace and fix if have any error(s) in the above statements.

```
import javax.swing.*;
import java.awt. *;
public class CheckBox extends JFrame {
      private JCheckBox java, c, python; //JCheckbox
      public CheckBox()
      {
            Container pane = getContentPane();
            pane.setLayout(new FlowLayout());
            java = new JCheckBox("Java");
            c = new JCheckBox("C++");
            python = new JCheckBox("Python");
            pane.add(java);
            pane.add(c); // capital C
            pane.add(python);
      }
      public static void main(String [] args) {
            CheckBox frame = new CheckBox(); //Jframe
            frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            frame.setTitle("Programming");
            frame.setSize(300, 100);
            frame.setVisible(true); //no true
      }
}
```

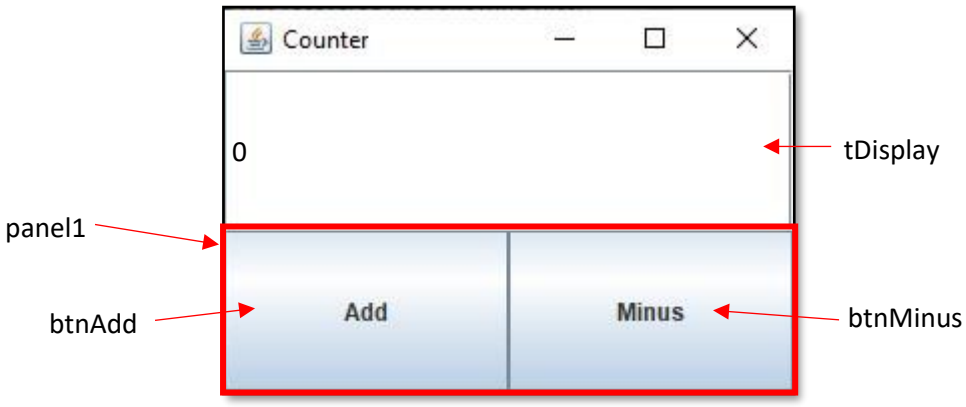
b) Sketch the expecting output



c) What the difference between checkbox and radio button?

CheckBox	Radio button
A CheckBox is a component that represents an item which shows a state of selected or unselected. We can change this state by clicking on the checkbox of the component.	A Radio Button is a component that represents an item with a state selected or unselected. Usually, a group of radio buttons is created to provide options to the user, but only one option can be selected at a time.
A standard CheckBox component contains a checkbox and a label that describes the purpose of the checkbox.	Radio Button will generate an ActionListener, ChangeListener and ItemListener interfaces
A CheckBox can generate either ItemListener or ActionListener interfaces	The radio buttons are often used in a group to display multiple options; therefore, they are used with Button Group class. The Button Group has a property that only one button in a group is selected at a given time and it does not have the visual appearance.

8. Consider the following figure.



a) What is the purpose of panel1?

A component that allows placing multiple components on it. It provides space in which an application can attach any other component.

b) Explain the similar and different between frame and panel?

Frame	Panel
A subclass of Container	A subclass of Window
Does not have title bar	Have a title bar
Does not have a border	Has a border

c) Complete the line of following statements.

```
L1 import javax.swing.*;
L2 import java.awt.*;
L3
L4 public class Counter extends JFrame
L5 {
L6 _____________________ //declare panel1
L7 _____________________ //declare tDisplay
L8 _____________________ //declare btnAdd and btnMinus
L9
L10 public Counter()
L11 {
L12 Container pane = getContentPane(); //get content pane
L13 ____________________________________ //set layout pane
L14 ____________________________________ //create tDisplay
```

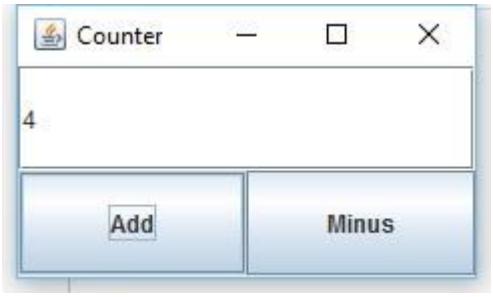
```
L15
L16 _____________________________ //create panel1
L17 _____________________________ //set layout panel1
L18 _____________________________ //create btnAdd
L19 _____________________________ //create btnMinus
L20 ____________________________ //add btnAdd in panel1
L21 ____________________________ //add btnMinus in panel1
L22
L23
L24 ____________________________ //add tDisplay in pane
L25 ____________________________ //add panel in pane
L26 }
L27
L28 public static void main (String[] args)
L29 {
L30 Counter frame = new Counter();
L31 frame.setTitle("Counter");
L32 frame.setSize(300, 200);
L33 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
L34 frame.setVisible(true);
L35 }
L36 }
```

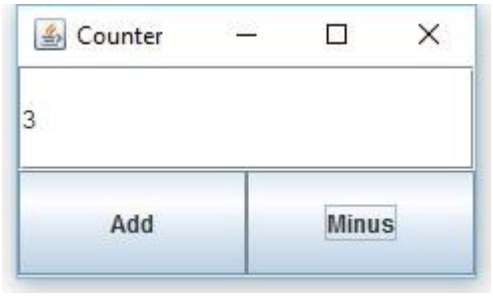
```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Counter extends JFrame implements ActionListener
{
	private JPanel panel1;//declare panel1
	private JTextField tDisplay;//declare tDisplay
	private JButton btnAdd, btnMinus; //declare btnAdd and btnMinus
	public Counter()
	{
		Container pane = getContentPane(); //get content pane
		pane.setLayout(new GridLayout(2,1)); //set layout pane
		tDisplay = new JTextField("0");//create tDisplay
		panel1 = new JPanel();//create panel1
		panel1.setLayout(new GridLayout());//set layout panel1
		btnAdd = new JButton("Add"); //create btnAdd
		btnMinus = new JButton("Minus"); //create btnMinus
		panel1.add(btnAdd); //add btnAdd in panel1
		panel1.add(btnMinus);//add btnMinus in panel1
		btnAdd.addActionListener(this); //Register event listener
		btnMinus.addActionListener(this); // Register event listener
		pane.add(tDisplay);//add tDisplay in pane
		pane.add(panel1);//add panel in pane
	}
	public void actionPerformed(ActionEvent e) {
		Object obj = e.getSource();
		int value = Integer.parseInt(tDisplay.getText());
		if (obj == btnAdd){
			value = value + 2;
			tDisplay.setText(value +"");
		}
		else
		{
			value = value - 1;
```

```
                tDisplay.setText(value +"");
            }
        }
        public static void main (String[] args)
        {
            Counter frame = new Counter();
            frame.setTitle("Counter");
            frame.setSize(300, 200);
            frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            frame.setVisible(true);
        }
}
```

Topic: Event Handling.

9. Based on previous question *(Question 8)*, write the code statements to handle event so that when the Add button in the GUI is clicked, the number in the box will increased by 2, meanwhile if Minus button is clicked, the number in the box will decrease by 1 as shown in the following figures.





Step 1: The previous class Counter should import event package (*add in L3*) and implement an event listener (*modify L4*).

```
L3 ____________________________
L4 public class Counter extends JFrame ____________________________
```

Step 2: Register event listener with the appropriate event source (component in the GUI) in the previous constructor (write statement in L22-L23).

```
L22 ____________________________________;
L23 ____________________________________;
```

Step 3: Write the code to handle event as shown in the above figures (insert the statement before main method).

```
public void actionPerformed(ActionEvent e) {

 Object obj = e.getSource();
 ____________________________________________;

 if (obj == btnAdd){        ____________________________;
          ____________________________;
 }
```

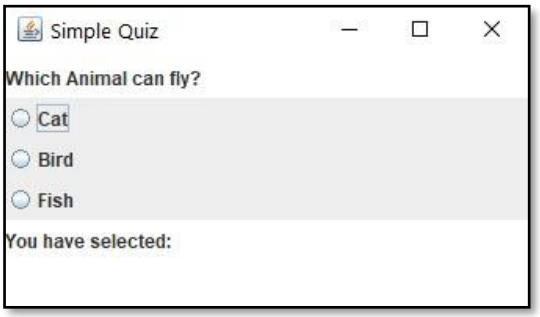
```
        else
        {
                _____________________________;
                _____________________________;
        }
      }
```

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Counter2 extends JFrame implements ActionListener
{
	private JPanel panel1;//declare panel1
	private JTextField tDisplay;//declare tDisplay
	private JButton btnAdd, btnMinus; //declare btnAdd and btnMinus
	public Counter2()
	{
		Container pane = getContentPane(); //get content pane
		pane.setLayout(new GridLayout(2,1)); //set layout pane
		tDisplay = new JTextField("0");//create tDisplay
		panel1 = new JPanel();//create panel1
		panel1.setLayout(new GridLayout());//set layout panel1
		btnAdd = new JButton("Add"); //create btnAdd
		btnMinus = new JButton("Minus"); //create btnMinus
		panel1.add(btnAdd); //add btnAdd in panel1
		panel1.add(btnMinus);//add btnMinus in panel1
		btnAdd.addActionListener(this); //Register event listener
		btnMinus.addActionListener(this); // Register event listener
		pane.add(tDisplay);//add tDisplay in pane
		pane.add(panel1);//add panel in pane
	}
	public void actionPerformed(ActionEvent e) {
		Object obj = e.getSource();
		int value = Integer.parseInt(tDisplay.getText());
		if (obj == btnAdd){
			value = value + 2;
			tDisplay.setText(value +"");
		}
		else
		{
			value = value - 1;
			tDisplay.setText(value +"");
		}
	}
	public static void main (String[] args)
	{
		Counter2 frame = new Counter2();
		frame.setTitle("Counter");
		frame.setSize(300, 200);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.setVisible(true);
	}
   }
```

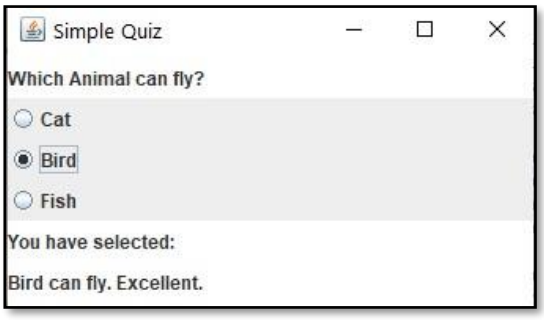
Section B Practice - Question 1

Write the GUI code statements and event for the following figure. User can choose only one answer from the radio button. The selected answer will be display in the text field such as the following figure. (Set frame size 350, 200).

Before user click radio button



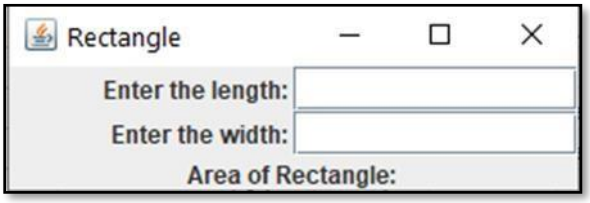
After user click radio button



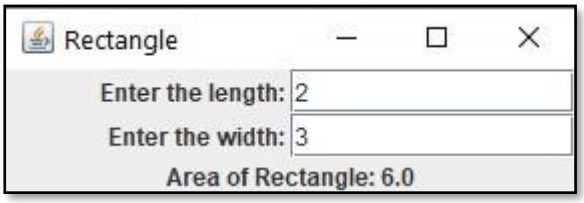
Practice - Question 2

Write the GUI code statements and event for the following figure. The output of rectangle area should be display such as the following figure after user insert the data and click “enter”. (Set frame size 300, 100).

Before user insert the data :



After user insert the data and click enter:



Practice - Question 3

Write the GUI code statements and event for the following figure. Customer can choose from the list of "Lovely Drink". The selected item will be display in the text field such as the following figure. (Set frame size 300, 200).

Before selection drink

After selection drink

Assignment – Question 4

A company named 'Create Your Own Pizza' is a small and famous homemade pizza in rural area at Kelantan. Since the company doesn't have enough staff and the pizza is hot selling due to very cheap and tasty, the company only limited one pizza for only one customer. The company has a very traditional way on ordering pizza. Only one staff will handle the order and the company doesn't have any system users to order the pizza. You as a freelance programmer try to help them to develop a simple order system for Create Your Own Pizza. The system should have some graphical user interface and related event for ordering the pizza. Figure 1 below show the GUI of the system.

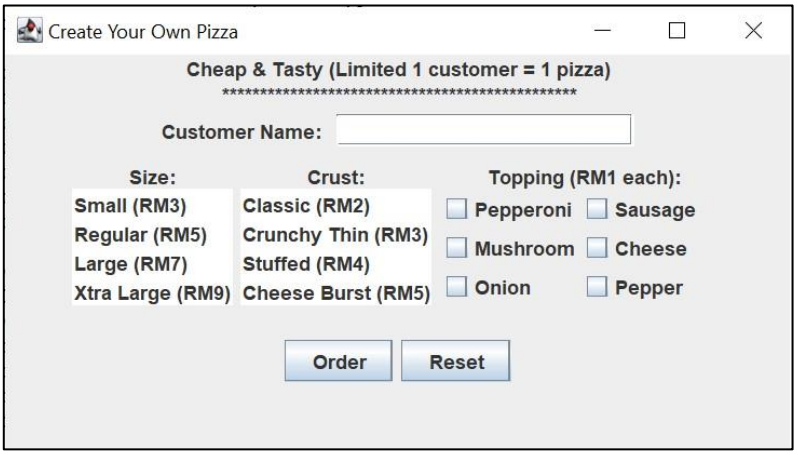


Figure 1

Customer must enter name, select size, crust and topping of pizza. The figure 2 below show the GUI when the form completed and order button clicked.

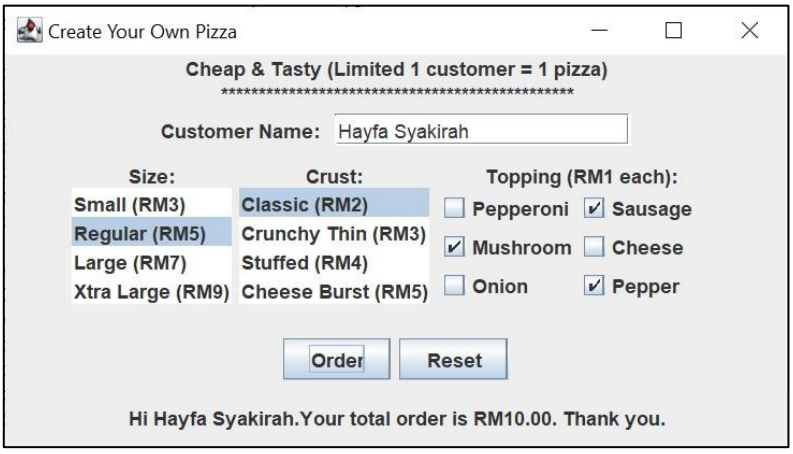


Figure 2

If the reset button clicked, all the data selected and the name filled before will be reset and it will produce the following GUI.

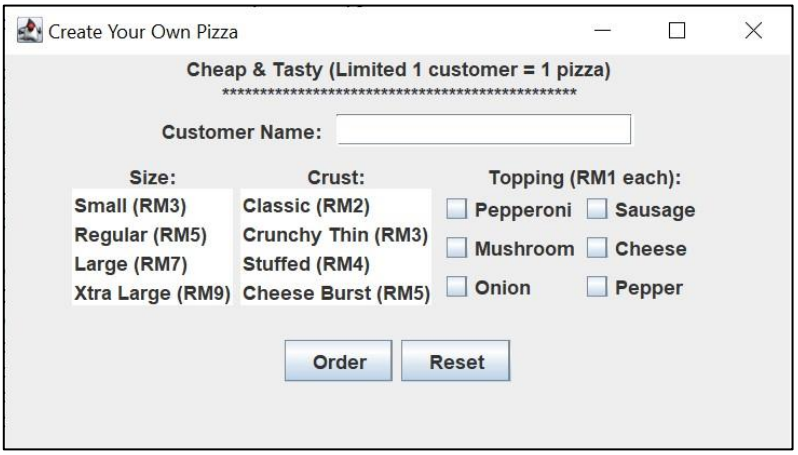
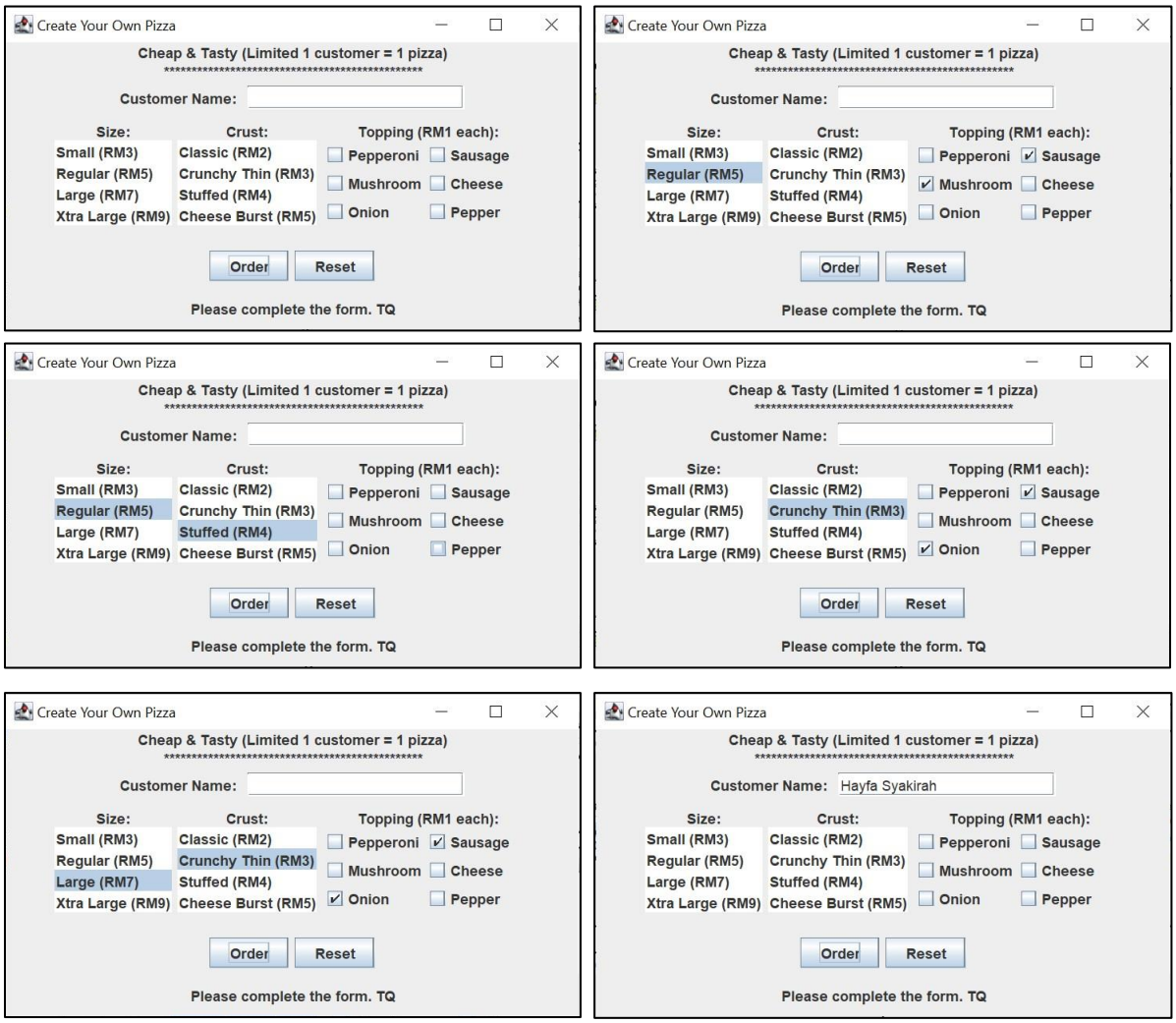


Figure 3

If the form is not completed such as the following figure and the order button is clicked, the feedback will display at the bottom with the message “Please complete the form. TQ”



Note:

- Pizza Size and Crust only can be selected by ONE selection.
- Topping can be selected by multiple selections.