#4B. SET INTERFACE

#code

```
import java.util.*;
public class SetDemo1
{
public static void main(String args[])
{
    Set<String> fruits = new HashSet<>();
    fruits.add("Apple");
    fruits.add("mango");
    fruits.add("banana");
    fruits.add("orange");
    fruits.add("Apple");
    System.out.println("Set of Fruits :");
    for (String fruit : fruits )
    {
        System.out.println(fruit);
    }
    System.out.println("\nSet Contains 'Mango' :"+fruits.contains("mango"));
fruits.remove("banana");
      System.out.println("\nSet after removing 'banana' :" +fruits);
    System.out.println("\nIterating using Iterator :");
    Iterator<String> iterator = fruits.iterator();
    while (iterator.hasNext())
    {
 System.out.println(iterator.next());
}
fruits.clear();
System.out.println("\nSet after clearing :" +fruits);

}



}
```

o/p

```
PS C:\Users\Admin\Desktop\SYCS-41\java> javac SetDemo1.java
PS C:\Users\Admin\Desktop\SYCS-41\java> java SetDemo1
Set of Fruits :
banana
orange
Apple
mango

Set Contains 'Mango' :true

Set after removing 'banana' :[orange, Apple, mango]

Iterating using Iterator :
orange
Apple
mango

Set after clearing :[]
```

##4C Map interface

#code

```
import java.util.*;
public class MapDemo
{
public static void main(String args[])
{
    Map<Integer,String> map = new HashMap<>();
    map.put(1, "Apple");
    map.put(2, "Banana");
    map.put(3, "Orange");
    map.put(4, "Mango");
    map.put(5, "Grapes");
    map.put(1, "cherry");
 System.out.println("Map of Fruits :");
for (Map.Entry<Integer, String> entry : map.entrySet())
{
System.out.println("Key :"+entry.getKey() + ", Value : " + entry.getValue());

}
System.out.println("\n value for key 2 : "+map.get(2));
map.remove(3);
System.out.println("\n after removing key 3 : ");
for (Map.Entry<Integer, String> entry : map.entrySet())
  {
    System.out.println("Key :"+entry.getKey() + ", Value : " +
entry.getValue());
  }
  System.out.println("\nMap contains key 4 : "+map.containsKey(4));
  System.out.println("map contains value 'Banana' :
"+map.containsValue("Banana"));
  System.out.println("\n Iterating using forEach method :");
  map.forEach((key, value) ->
  System.out.println("Key :"+key + ", Value :"+value));
  map.clear();
  System.out.println("\nMap after clearing :"+map);
}



}
```

o/p

```
PS C:\Users\Admin\Desktop\SYCS-41\java> javac MapDemo.java
PS C:\Users\Admin\Desktop\SYCS-41\java> java MapDemo
Map of Fruits :
Key :1, Value : cherry
Key :2, Value : Banana
Key :3, Value : Orange
Key :4, Value : Mango
Key :5, Value : Grapes

 value for key 2 : Banana

 after removing key 3 :
Key :1, Value : cherry
Key :2, Value : Banana
Key :4, Value : Mango
Key :5, Value : Grapes

Map contains key 4 : true
map contains value 'Banana' : true

 Iterating using forEach method :
Key :1, Value :cherry
Key :2, Value :Banana
Key :4, Value :Mango
Key :5, Value :Grapes

Map after clearing :{}
```

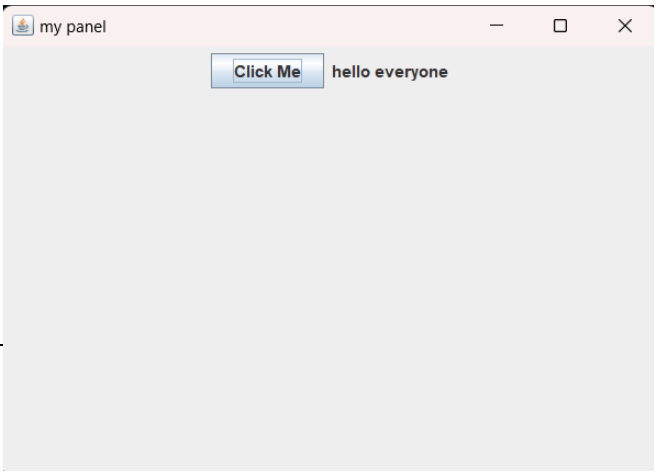
##practical-5

#code

practice code for jpanal in components

```
#code
import javax.swing.*;
public class myJpanel
{
  public static void main(String[] args)
  {
    JFrame fr = new JFrame("my panel");
    fr.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    fr.setSize(450,350);
    JPanel p= new JPanel();
    JButton bt= new JButton("Click Me");
    JLabel lb= new JLabel("hello everyone");
    p.add(bt);
    p.add(lb);
    fr.add(p);
    fr.setVisible(true);

  }
}
```

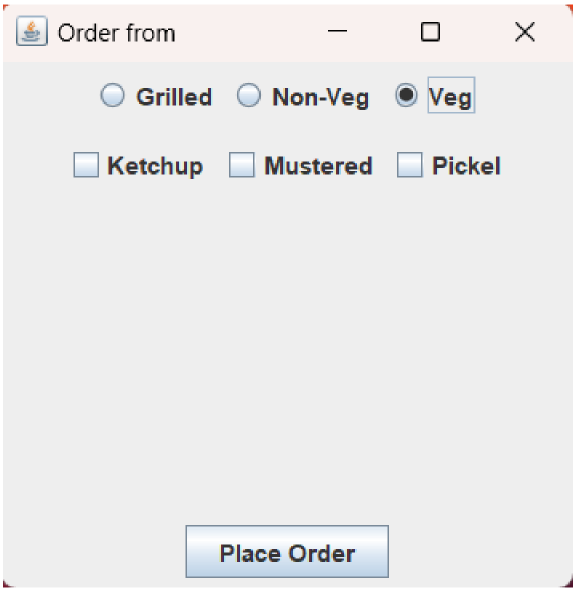


##practice JPanel

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

public class demo {
    public static void main(String[] args) {
        JFrame frame = new JFrame("Order from");
        frame.setSize(300, 300);
        frame.setLocation(200, 200);
        frame.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });

        JPanel p = new JPanel();
        final ButtonGroup bt = new ButtonGroup();
        JRadioButton rd;
        p.add(rd = new JRadioButton("Grilled"));
        rd.setActionCommand("Grilled");
        bt.add(rd);
        p.add(rd = new JRadioButton("Non-Veg"));
        rd.setActionCommand("Non-Veg");
        bt.add(rd);
        p.add(rd = new JRadioButton("Veg", true));
        rd.setActionCommand("Veg");
        bt.add(rd);
```



○ Grilled ○ Non-Veg ● Veg

□ Ketchup □ Mustered □ Pickel

```
        final JPanel c = new JPanel();
        c.add(new JCheckBox("Ketchup"));
        c.add(new JCheckBox("Mustered"));
        c.add(new JCheckBox("Pickel"));

        JPanel o = new JPanel();
        JButton or = new JButton("Place Order");
        o.add(or);

        Container con = frame.getContentPane();
        con.setLayout(new BorderLayout());
        con.add(p, BorderLayout.NORTH);
        con.add(c, BorderLayout.CENTER);
        con.add(o, BorderLayout.SOUTH);

        or.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String en = ((JRadioButton)
bt.getSelection()).getActionCommand();
                System.out.println(en + " sandwich");
                Component[] com = c.getComponents();
                for (int i = 0; i < com.length; i++) {
                    JCheckBox cb = (JCheckBox) com[i];
                    if (cb.isSelected())
                        System.out.println("with " + cb.getText());
                }
            }
        });

        frame.setVisible(true);
    }
}
```