
Reg.No

SNS COLLEGE OF TECHNOLOGY

(Autonomous)

B

MCA- Internal Assessment –II (July 2023)

Academic Year 2022-2023(Even) / Second Semester

19CAT608 – Java Programming – Answer key

Time: 1 1/2 Hours **Maximum Marks: 50**

Answer All Questions

PART - A (5 x 2 = 10 Marks)

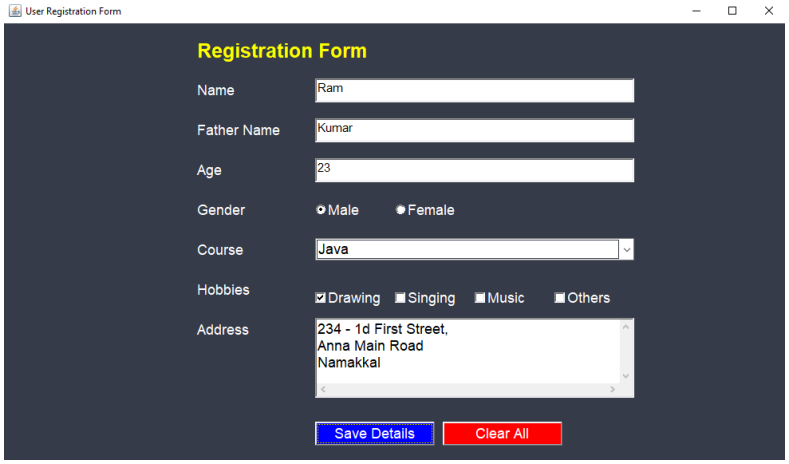
1	**Explain any two methods under WindowListener Interface.** 1. windowActivated (WindowEvent e) 2. windowOpened (WindowEvent e)	CO2	Und
2	**Compare final and finalize.** 1. The final keyword is used to declare a variable or method that cannot be modified or overridden. 2. The finalize method is a special method that is called by the garbage collector when an object is about to be destroyed. The finalize() method can be used to perform any necessary cleanup operations before the object is destroyed.	CO2	Ana

3	**Write syntax used to send and receive data via Datagramsocket.**
	Send:
	DatagramPacket dp = new DatagramPacket(str.getBytes(), str.length(), ip, 3000);
	ds.send(dp);
	Receive:
	DatagramPacket dp = new DatagramPacket(buf, 1024);
	ds.receive(dp);
		CO3	Rem

4	**List the class and interface of thread.** Class: Thread Interface: Runnable	CO3	Ana
5	**Write about the need of inner classes in java.** To access the private members and methods	CO3	Und

PART - B (2x13=26, 1x14=14 Marks)

6 (a) **Design a registration form using AWT components and event handling methods.**



CO2 App

```
import java.awt.*;
class MyApp extends Frame {
Label
```

```
lblTitle,lblName,lblFather,lblAge,lblGender,lblCourse,lblHobbies,lbl
Address;        TextField txtName,txtFather,txtAge;
        TextArea txtAddress;   Checkbox checkMale,
checkFemale,Hobbies1,Hobbies2,Hobbies3,Hobbies4;
        CheckboxGroup cbg; Choice Course; Button btnSave,btnClear;
        public MyApp() {
                super("User Registration Form");
                setSize(1000, 600);// w,h
                setLayout(null);
                setVisible(true);
                Color formColor = new Color(53, 59, 72);
                setBackground(formColor);

                Font titleFont = new Font("arial", Font.BOLD, 25);
                Font labelFont =new  Font("arial", Font.PLAIN, 18);
                Font textFont =new  Font("arial", Font.PLAIN, 15);

                lblTitle=new Label("Registration Form");
                lblTitle.setBounds(250,40,300,50);
                lblTitle.setFont(titleFont);
                lblTitle.setForeground(Color.YELLOW);
                add(lblTitle);

                lblName=new Label("Name");
                lblName.setBounds(250,100,150,30);
                lblName.setFont(labelFont);
                lblName.setForeground(Color.WHITE);
                add(lblName);

                txtName=new TextField();
                txtName.setBounds(400,100,400,30);
                txtName.setFont(textFont);
                add(txtName);

                lblFather=new Label("Father Name");
                lblFather.setBounds(250,150,150,30);
                lblFather.setFont(labelFont);
                lblFather.setForeground(Color.WHITE);
                add(lblFather);

                txtFather=new TextField();
                txtFather.setBounds(400,150,400,30);
                txtFather.setFont(textFont);
                add(txtFather);

                lblAge=new Label("Age");
                lblAge.setBounds(250,200,150,30);
                lblAge.setFont(labelFont);
                lblAge.setForeground(Color.WHITE);
                add(lblAge);

                txtAge=new TextField();
                txtAge.setBounds(400,200,400,30);
                txtAge.setFont(textFont);
                add(txtAge);


                lblGender=new Label("Gender");
                lblGender.setBounds(250,250,150,30);
                lblGender.setFont(labelFont);
                lblGender.setForeground(Color.WHITE);
                add(lblGender);

                cbg = new CheckboxGroup();
```

```
checkMale = new Checkbox("Male",cbg,true);
checkMale.setBounds(400,250, 100, 30);
checkMale.setFont(labelFont);
checkMale.setForeground(Color.WHITE);
add(checkMale);

checkFemale = new Checkbox("Female",cbg,false);
checkFemale.setBounds(500,250, 100, 30);
checkFemale.setFont(labelFont);
checkFemale.setForeground(Color.WHITE);
add(checkFemale);

lblCourse=new Label("Course");
lblCourse.setBounds(250,300,150,30);
lblCourse.setFont(labelFont);
lblCourse.setForeground(Color.WHITE);
add(lblCourse);

Course= new Choice();
Course.setFont(labelFont);
Course.setBounds(400, 300, 400, 50);
Course.add("C");
Course.add("C++");
Course.add("Java");
Course.add("C#");
Course.add("Python");
add(Course);

lblHobbies=new Label("Hobbies");
lblHobbies.setBounds(250,350,150,30);
lblHobbies.setFont(labelFont);
lblHobbies.setForeground(Color.WHITE);
add(lblHobbies);

Hobbies1=new Checkbox("Drawing");
Hobbies1.setBounds(400, 350, 100, 50);
Hobbies1.setFont(labelFont);
Hobbies1.setForeground(Color.WHITE);
add(Hobbies1);

Hobbies2=new Checkbox("Singing");
Hobbies2.setBounds(500, 350, 100, 50);
Hobbies2.setFont(labelFont);
Hobbies2.setForeground(Color.WHITE);
add(Hobbies2);

Hobbies3=new Checkbox("Music");
Hobbies3.setBounds(600, 350, 100, 50);
Hobbies3.setFont(labelFont);
Hobbies3.setForeground(Color.WHITE);
add(Hobbies3);

Hobbies4=new Checkbox("Others");
Hobbies4.setBounds(700, 350, 100, 50);
Hobbies4.setFont(labelFont);
Hobbies4.setForeground(Color.WHITE);
add(Hobbies4);

lblAddress=new Label("Address");
lblAddress.setBounds(250,400,150,30);
lblAddress.setFont(labelFont);
lblAddress.setForeground(Color.WHITE);
add(lblAddress);
```

```
                txtAddress=new TextArea(10,30);
                txtAddress.setBounds(400,400,400,100);
                txtAddress.setFont(labelFont);
                add(txtAddress);


                btnSave=new Button("Save Details");
                btnSave.setBounds(400,530,150,30);
                btnSave.setFont(labelFont);
                btnSave.setBackground(Color.BLUE);
                btnSave.setForeground(Color.WHITE);
                add(btnSave);

                btnClear=new Button("Clear All");
                btnClear.setBounds(560,530,150,30);
                btnClear.setFont(labelFont);
                btnClear.setBackground(Color.RED);
                btnClear.setForeground(Color.WHITE);
                add(btnClear);


                // Close Button Code
                this.addWindowListener(new WindowAdapter() {
                        public void windowClosing(WindowEvent we) {
                                System.exit(0);
                        }
                });
        }

}

public class app {
        public static void main(String[] args) {
                MyApp frm = new MyApp();
        }

}
```

(Or)

(b) **Narrate Garbage Collection with a suitable example.** CO2 Rem

Garbage Collection is process of reclaiming the runtime unused memory automatically. In other words, it is a way to destroy the unused objects.

Advantage:

1. Memory Efficient
2. Automatically done

Difference ways:

- By nulling the reference

Employee e=new Employee();
e=null;

- By assigning a reference to another

Employee e1=new Employee();
Employee e2=new Employee();
e1=e2;

- By anonymous object etc.

new Employee();

Two methods:

1. finalize()
2. gc()

Example

```
1. public class TestGarbage1{
2.  public void finalize(){System.out.println("object is garbage collected");}
3.  public static void main(String args[]){
4.   TestGarbage1 s1=new TestGarbage1();
5.   TestGarbage1 s2=new TestGarbage1();
6.   s1=null;   s2=null;   System.gc();
7.  } }
```

7 (a) Give suitable examples to read from and write bytes to a file. CO3 App

Java I/O (Input and Output) is used to process the input and produce the output. Java uses the concept of a stream to make I/O operation fast. The java.io package contains all the classes required for input and output operations. We can perform file handling in Java by Java I/O API.

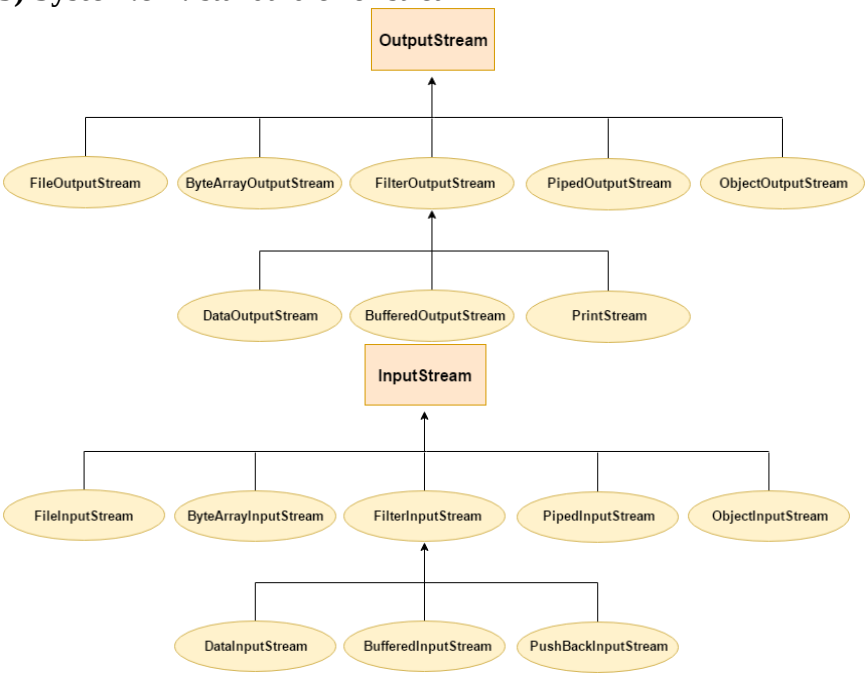
Stream:

It is a sequence of data

1) System.out: standard output stream

2) System.in: standard input stream

3) System.err: standard error stream




```
import java.io.FileOutputStream;
public class FileOutputStreamExample {
public static void main(String args[]){
try{
FileOutputStream fout=new FileOutputStream("D:\\testout.txt");
fout.write(65);   fout.close();   System.out.println("success...");
}catch(Exception e){System.out.println(e);}
}   }
```

```
import java.io.FileInputStream;
public class DataStreamExample {
public static void main(String args[]){
try{
FileInputStream fin=new FileInputStream("D:\\testout.txt");
```

```
    int i=fin.read();  System.out.print((char)i);    fin.close();
    }catch(Exception e){System.out.println(e);}
}   }
```

(Or)

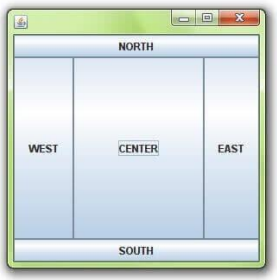
(b) Illustrate Commonly used methods in TCP/IP. CO3 Und

- DatagramPacket
- DatagramSocket
- DatagramSocketImpl
- InterfaceAddress
- JarURLConnection
- MulticastSocket
- InetSocketAddress
- InetAddress etc..

8 (a) Develop a Java program to demonstrate Layout Manager.

The LayoutManagers are used to arrange components in a particular manner. The **Java LayoutManagers** facilitates us to control the positioning and size of the components in GUI forms.

1. java.awt.BorderLayout
2. java.awt.FlowLayout
3. java.awt.GridLayout
4. java.awt.CardLayout
5. java.awt.GridBagLayout
6. javax.swing.BoxLayout
7. javax.swing.GroupLayout
8. javax.swing.ScrollPaneLayout
9. javax.swing.SpringLayout etc.



CO2 App

```
import java.awt.*;
import javax.swing.*;
public class Border
{
JFrame f;
Border()
{
   f = new JFrame();
   JButton b1 = new JButton("NORTH");
   JButton b2 = new JButton("SOUTH");
   JButton b3 = new JButton("EAST");
   JButton b4 = new JButton("WEST");
   JButton b5 = new JButton("CENTER");
   f.add(b1, BorderLayout.NORTH); // b1 will be placed in the North Direction
   f.add(b2, BorderLayout.SOUTH);  // b2 will be placed in the South Direction
   f.add(b3, BorderLayout.EAST);  // b2 will be placed in the East Direction
   f.add(b4, BorderLayout.WEST);  // b2 will be placed in the West Direction
   f.add(b5, BorderLayout.CENTER);  // b2 will be placed in the Center
   f.setSize(300, 300);
   f.setVisible(true);
}
public static void main(String[] args) {
   new Border();    }    }
```

(Or)

(b) Create a class A with two nested classes B and C. Define B as a member of A and C as local inner class and define three methods inA(),inB() and inC() in these three classes respectively. Create a class demo to invoke all three methods. CO3 Eva

Sample:

```
class  A
 {
 A();
 class B
  B();
  {
        Local inner class C
        {
              C()

        }
  }
}
class Innersam
  public static void main(String args[])
  {
  C c=new C();
```