TASK FOUR: PART ONE

1.Explain the difference between primitive and reference data types

Primitive data types	Reference data types
It can store exactly one value of its declared type at a time	Store the location of an object in the computer's memory
Primitive types are Boolean, byte, char, short, int, long, float and double	All other types are reference types, so classes, which specify the types of objects, are reference types

2.Define scope of a variable

Scoop of a variable is its lifetime in the program. This means that the scope of a variable is the block of code in the entire program where the variable is declared, used, and can be modified.

3.Why is initialization of variable required.

This is because local variables don't have a default value and the compiler won't let us use an uninitialized value.

4.Differentiate between static, instance and local variables.

Local variable	Instance variable	Static variable

Define within a method d or a code block	Defined outside a method at the class level	Defined outside a method at the class level
Remains in memory as long as the method executes	Remains in memory as long as the object is in memory	Remains in memory as long as program executes
Does not require any special keyword	Does not require any keyword but any access specifier (private or public) can be specified. Typically private or protected is used	Requires the static keyword to be specified. In addition any access specifier (private or public) can be specified. Typically, public is used
Remains in memory as long as the method executes	Remains in memory as long as the object is in memory	Remains in memory as long as program executes

5.Differentiate between widening and narrowing casting in java.

Widening casting involves the conversion of a smaller data type to the larger type size example byte_> short_> char_> int_> long_> float_> double while Narrowing casting this involves converting a larger data type to a smaller size type example double_> float_> long_> int_> char_> short_> byte

6.Fill in the table in the missing values

TYPE	SIZE(IN BYTES)	DEFAULT	RANGE
Boolean	1 bit		True, false
char	2		'\0000' to '\ffff'
byte		0	-2^7 to + 2^7-1

short		0	-2^15 to+2^15-1
int	4		-2^31 to+2^31-1
long		0L	
float	4	00.0F	
double	8		-1.8E+308 to+1.8E+308

7.Importance of using java packages

A package in java is used to group related classes. We use java packages to name conflicts, and write a better maintainable code.

- Java package provide access protection
- Java package is used to categorize the classes and interference so that they can be easily maintained
- Java package removes naming collision

8.Explain 3 controls used while creating GUI applications in java language

- Graphical components- that make up the Graphical User Interface
- Listener methods- that receives the events and respond to them
- Application methods- that do useful work for the user

9.Difference between containers and components as used in java

A container is a window-like component that can contain other components examples of containers are Jpanel, Jframe, JApplet while a

component is an object, like a button or a scroll bar examples of components are JLabel, JTextfield, JButton

10.Write a java program to reverse an array having five items of type int.

```
Public class Reverse Array {
   Public static void main (string [] args) {
    //Initialize array
      Int [] arr = new int [] {1, 2, 3, 4, 5};
      System.out, println (“Original array:”);
      For (int I = 0; I < arr. length; i++) {
         System.out. Print(arr[i]+ “”);
      }
      System.out. Println ();
      System.out. Println (“Array in reverse order:”);
       //loop through the array in reverse order
       For (int I = arr. length-1; I >= 0; i--) {
            System.out, print(arr[i] + “”)
       }
}
```

11.Expain what is meant by the term event, give at least two examples of events, and discuss how a program might respond to those events.

- An event is anything that can occur asynchronously, not under the control of the program, to which the program might want to respond. GUI programs are said to be “event-driven” because for the most part, such programs simply wait for events and respond to them when they occur. In many (but not all) cases, an event is the result of a user action, such as when the user clicks the mouse button, types a character or clicks a button. The program might respond to a mouse-click on a canvas by drawing a shape, to a typed character by adding the character to an input box, or to a click on a button by clearing a drawing. More generally, a programmer can set up any desired response to an event by writing an event-handling routine for that event.

12.Explain the difference between the following terms in java programming

- Polymorphism allows program code to have different meaning or functions while encapsulation is the process of keeping classes private so thy cannot be modified by external codes.
- Method overloading is a feature that allows a class to have more than one method with the same name, but with different parameters while method overriding allows a subclass to provide a specific implementation of a method that is already provided by one of its parent classes
- A class is a template used to create objects and to define object data types and methods while an interface I an abstract type that is used to describe a behavior that classes must implement
- Inheritance is concept that acquires the properties from one class to other classes while polymorphism refers to the ability of a class

to provide different implementations of a method, depending on the type of object that is passed to the method.

13.Using examples explain the two possible ways of implementing polymorphism. Show code in java

- ❖ Method overloading

Example in java

```
Class shapes {
    Public void area () {
       System.out. Println ("find
     area ");
   }
Public void area (int r) {
        System.out. Println ("circle
       Area = "+3.142*r*r);
    }
Public void area (double b,
double h) {
system. Out. println ("Triangle
area =" +0.5*b*h);
     }
Public void area (int l, int b)
{
```

```
System. Out. Println (“Rectangle
Area = “+l*b);
      }

}
   Class main {
         Public static void
    Main (string [] args) {
           Shapes myShape = new
     Shapes (); //create a shapes
      Object
          MyShape. Area ();
           MyShape. Area (5);
           MyShape. Area (6.0, 1.2);
           MyShape. Area (6.2);
       }
}
```

Output

Find area

Circle area =78.5

Triangle area =3.60

Rectangle =12

❖ Method overriding

Example in java

```
 class vehicle {
      //defining a method
      Void run ()
{system. Out. println (“vehicle
Is moving”);}
}
//creating a child class
Class car2 extends vehicle {
       //defining the same method
as in the parent class
        void run ()
{system. Out. Println (“car is
Running safely “);}

Public static void
Main (string args [])}
   Car2 obj = new
Car2() ;//creating object
    Obj. run () ;//calling method
    }
}
```

Output

Car is moving safely