21/05093 – MUMO DENNIS

Task 3 Part 1

1. Explain the differences between primitive and reference data types.

- ➔ Primitive data types specify the size and type of variable values while reference data types specify the reference/address of the variable values.
- ➔ Primitive data types are considered to be storage locations declared in a memory, while reference data types are considered to be objects.

2. Define the scope of a variable (hint: local and global variable)

A scope is a region of the program. It is where variables can be declared: Inside a function or a block which is called *local variables*, In the definition of function parameters which is called formal parameters. Outside of all functions which are called *global variables*.

3. Why is initialization of variables required?

- ➔ To avoid run-time errors.

4. Differentiate between static, instance and local variables.

Static variable - is accessible in the whole class

Instance variable - is also accessible in the whole class

Local variable - is only accessible in the method where it is created.

5. Differentiate between widening and narrowing casting in java.

Widening conversions preserve the source value but can change its representation while narrowing conversion changes a value to a data type that might not be able to hold some of the possible values.

6. The following table shows data type, its size, default value and the range. Filling in the missing values.

TYPE	SIZE (IN BYTES)	DEFAULT RANGE
boolean	1 bit	false true, false
Char	2	'\u0000' ‘\0000’ to ‘\ffff’
Byte	1	0 -27 to +27-1
Short	2	0 -215 to +215-1

Int	4	0 -231 to +231-1
Long	8	0L -$2^{63}$to 2^{63}-1
Float	4	00.0f 3.4E-38 to 3.4E+38

Double 8 0.0d -1.8E+308 to +1.8E+308

7. Explain the importance of using Java packages

- To group related classes.
- To avoid name conflicts.
- To write a better maintainable code.

8. Explain three controls used when creating GUI applications in Java language.

- Label - Is used to provide a descriptive text string that cannot be changed directly by the user.
- TextField - Used to get text input from the user into the program for processing.
- Button - Used to execute blocks of code in a program when clicked by the user.

9. Explain the difference between containers and components as used in Java.

A Container is a component that can have other components or other containers while a component cannot have other components or containers in it.

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

```
import java.util.*;
import java.util.stream.*;
public class Main {
    public static void main(String[] args) {
    //creating my array
    Integer[] myArray = {11,20,34,45,59}; //the five items in the
    array

    //print the array starting from first element
    System.out.println("Original array:");
    for(int i=0;i<myArray.length;i++) {
         System.out.print(myArray[i] + " ");
    }
    System.out.println();

    //print the array starting from last element
    System.out.println("Printing the array in reverse
         order:"); for(int
         i=myArray.length-1;i>=0;i--) {
         System.out.print(myArray[i] + " ");}
    }
}
```

11. Programs written for a graphical user interface have to deal

with “events.” Explain what is meant by the term event.

Give at least two different examples of events, and discuss how a program might respond to those events.

An event Is a change in the state of an object by performing actions triggered by the user when running the program.

Examples:

When a button is clicked (action), close the application (event).

When a key is pressed on the keyboard (action), close a dialog box (event).

12. Explain the difference between the following terms as used in Java programming.

➔ Method overloading and method overriding - Method overloading happens at compile-time while method overriding happens at runtime.

➔ Polymorphism and encapsulation - Polymorphism allows program code to have different functions while encapsulation is the process of keeping classes private so they cannot be modified by external codes.

➔ Class and interface - A class doesn't support multiple inheritance while interface supports multiple inheritance.

➔ Inheritance and polymorphism - Inheritance is one in which a new class is created (derived class) that inherits the features from the already existing class(Parent class). Whereas polymorphism is that which can be defined in multiple forms.