**Java programming**

**Exercise 1 of 16**

**Instructions:**

All programs should be written, and linked to an online repository like GitHub.

A video to get you started with GitHub has been posted on Moodle.

After completing your assignment, post the link on the link on Moodle. An instructor will follow the posted link to access and grade your work.

Note that: Your program should always be well-commented. At the top of your source code file, you should write a short description of what your program does and add other comments to help in explaining your code.

All of your variables should be given a deceptive name. Avoid giving your variables names like a, b, I, x, y etc.

In case you copy your friend's work, you both get a Zero (0).

**Section 1:**

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

Primitive data type stores the actual value making the memory efficient and faster access while reference store the address enabling them to reference complex object and share data across different program part

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

Local variables: These are declared within a method, constructor, or block, and can only be accessed within that method or block. Global (Instance) variables: These are declared within a class but outside methods. They can be accessed by any method in the class. If they are static, they belong to the class itself and not instances.

1. Why is initialization of variables required.

Initialization of variables is required to provide an initial value to a variable before it is used. This prevents undefined behavior and errors that could arise from trying to use variables that contain garbage values.

1. Differentiate between static, instance and local variables.

Static variables

These variables are shared across all instances of a class and are available to all methods. They are created at the start of the program and destroyed at the end. Static variables are similar to global variables.

Instance variables

These variables are specific to each object of a class. They are created and destroyed along with the objects. Instance variables are stored in separate copies within each object.

Local variables

These variables are used for temporary storage within methods or blocks. They are created when a function starts and deleted when the function is completed. Local variables are only recognized inside their functions.

1. Differentiate between widening and narrowing casting in java.

Widening casting safely converts smaller types to large types while narrowing requires explicit conversion from large to smaller types which can lead to data loss

1. 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 | -128 to 127 |
| Short | 2 | 0 | -215 to +215-1 |
| Int | 4 | 0 | -231 to (231)-1 |
| Long | 8 | 0L | -263to (263)-1 |
| Float | 4 | 00.0f | -3.4e-38 to +3.4e+38 |
| Double | 8 | 0.0d | -1.8E+308 to +1.8E+308 |

1. Define class as used in OOP

A class in Object-Oriented Programming (OOP) is a blueprint or template for creating objects. It defines properties (variables) and behaviors (methods) that objects created from the class will have.

1. Explain the importance of classes in Java programming.

Classes allow for modular, reusable, and organized code by grouping data and methods together. They enable the use of OOP principles like encapsulation, inheritance, and polymorphism, which make code more maintainable and scalable.