**Data-structure**

Way to store and organise our data in the memory in an efficient way so that retrieval and manipulation of data becomes easy is called data-structure.

**Algorithm**

Steps to solve a problem is called algorithm.

**Pseudo-code**

**Pseudo-code** is a simplified, informal way of describing an algorithm or program logic without the strict syntax of a programming language. It combines plain language with programming constructs to represent the steps of an algorithm clearly and concisely.

**Difference between library and framework**

**Difference Between Library and Framework**

| **Feature** | **Library** | **Framework** |
| --- | --- | --- |
| **Definition** | A collection of pre-written code that performs specific tasks. | A structured platform that provides built-in functionalities and controls the application's flow. |
| **Control** | The developer calls and controls the library. | The framework controls the flow and calls the developer's code (Inversion of Control - IoC). |
| **Flexibility** | You can use specific parts of a library as needed. | You must follow the framework's structure and rules. |
| **Dependency** | Can be used independently in any project. | Works as a complete structure; difficult to use partially. |
| **Examples** | JDBC (for database connections), Hibernate (ORM), Apache Commons (utility functions). | Spring, Spring Boot, Angular, Django. |

**LOOSE-COUPLING**

Loose coupling means two codes(or components) would be minimally dependent on each other. Changes on one part won’t heavily impact the other part, making the system flexible(can add new features, modify existing ones, or fix bugs without affecting other parts of the system).

**NULL POINTER EXCEPTION**

When we try to access an object which is not present then we will have NULL POINTER EXCEPTION (it points to nothing or null).

== operator will check value in case of primitive datatype

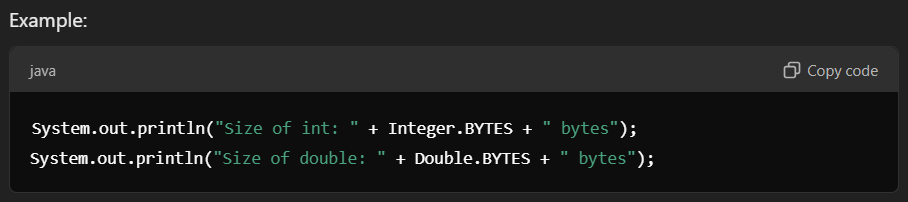
== operator will check address in case of non-primitive datatype(because reference variable contains address only so it will check whatever present inside the variable whether it is value or address).

equals() method will check address in case of non-primitive datatype(if equal method is not overridden) because in object class equals method call hashcode() method which return address of object and based on the object address equals method will return true or false.

But in all predefined non-primitive datatype(like Integer, Character, String) in these classes equals() and hashcode() is over-ridden that why in case of these equals() method will check value.

* In java you cannot print the size of variable like in c or c++ ( sizeOf() method)

Alternative is



* In java if we want to print address of any variable you can call toString() method – it will give you fullyqualified class name@hexadecimal address

Or you can call hashCode() method it will give you integer representation of hexadecimal address of that variable.

Thread- safe

thread-safe objects may allow multiple threads to operate on them simultaneously, ensuring the object's state remains consistent and free of race conditions.

How to wrap to your array

Reach to index where ever you want to go % arraysize ….. will give respective index in the array

Difference between for loop and while loop

for loop is used when we know the exact number of times the loop will run.

While loop is used when we don’t know the exact number of times the loop will run, it depends on some condition.