EXPERIMENT 1

// Introduction to Java

About Java

Java is a **high-level**, **general-purpose programming language** that supports the **object-oriented programming paradigm**. Also, the programs written in Java are **platform independent** and thus, **portable**.

- **High-level -** This signifies that Java is a more human friendly programming language and so, it is easy to learn and use.
- **General-Purpose -** This means that Java can be used to write programs for a wide variety of applications. For example, Java is used for mobile applications, web applications, etc.
- **Object-Oriented -** This indicates that Java enables programmers to develop object-oriented programs.
  Object-oriented programming paradigm divides a computational task into sub-tasks, implements these sub-tasks as **modules** or **objects**, and then integrates these objects as an entire system by making them communicate with each other.
- **Platform Independent -** By platform independency it is meant that Java programs can be executed on any platform (computer architecture and operating system).
  This is possible because Java has a **Java Virtual Machine** mechanism that converts compiled Java bytecode from any platform into machine specific code for the target platform.
- **Portable -** This means that since Java programs are platform independent, they can be easily ported from one platform to another.

Development Environment for Java

- For programming in Java, a **Java Development Kit (JDK)** must be installed on the system. This is a software that provides all functionalities required for developing applications and applets in Java.
  - JDK has a very important component known as **Java Runtime Environment (JRE or Java RTE)**, whose specific purpose is to support execution of Java programs on a system. So, JRE contains functionalities needed by a Java program at the run time for its successful execution.
    - For instance, JRE has a **Java Virtual Machine (JVM)** that brings in platform independency for Java. It is a software that converts a compiled Java bytecode (from a Java program or any other language) from any platform into machine specific code for the target platform.

★ JDK, JRE and JVM, all three are platform dependent i.e., they are built and customised for each platform supporting Java. However, they work in a manner that a programmer can develop platform independent Java programs.

Java Execution Mechanism

- A source file in Java is created with the '**.java**' extension.
- This file gets executed in the system in two phases:

- It is first compiled by the **Java compiler** into a **bytecode file** with the extension '**.class**'. This bytecode is not the target machine code to be executed by the system. In fact, it is an intermediate code generated to be handled by the **Java Virtual Machine (JVM)**.
- The Java Virtual Machine is an **interpreter** accompanying Java, that interprets this bytecode into machine specific code, and this machine code can finally be executed by the target system.

Software used to learn Java in this course -

- JDK 21 from Java's Standard Edition
- IDE like Visual Studio Code or Eclipse or IntelliJ Idea, etc.