

RETNAA MAXWELL DAYOK

Explain briefly, the Java compilation process in your own words.

When you write Java code, it's written in a human-readable format that the computer can't directly understand. So, before it can be executed, the code needs to be translated into a machine-readable format. That's where the Java compiler comes in.

The Java compilation process can be broken down into a few simple steps:

1. **Writing Java code:** You write Java code in a text editor or an integrated development environment (IDE).
2. **Compiling Java code:** When you're ready to run your Java code, you compile it using the Java compiler. The compiler takes your Java code and turns it into a class file that contains bytecode, which is a low-level representation of your code that can be understood by the Java Virtual Machine (JVM).
3. **Running Java code:** Once your code has been compiled, you can run it using the JVM. The JVM interprets the bytecode and executes your program.

That's it! So, to summarize, you write Java code, compile it using the Java compiler, and then run it using the JVM

Difference between JDK and JRE

The difference between the JDK (Java Development Kit) and JRE (Java Runtime Environment) is that the JDK includes everything you need to develop and run Java programs, while the JRE only includes the tools necessary to run Java programs. If you're just running Java programs, the JRE is all you need. But if you're developing Java programs, you'll need the JDK.

I the computer is so smart, why do we need a compiler

The computer only understands machine language which is the language of 0s and 1s. However this machine language isn't what humans or programmers can understand easily because it consists of long strings of binary code.

To make programming easier for humans, we develop programming languages just like Java, Python and so on. These programming languages allow us to write code that is easier to read and write, but it still needs to be translated into machine language before the computer can execute it.

That's where a compiler comes in. A compiler is a software tool that translates the code we write in a programming language into machine language that the computer can understand and execute. So when we write a program in a high-level language like Java, the compiler takes that code and turns it into a program that the computer can understand or run.

SUMMARY OF CHAPTER ONE

Java is one of the world's most used programming languages. For many organizations, the preferred language for meeting their enterprise programming needs is Java. Java is also widely used for implementing Internet-based applications and software for devices that communicate over a network.

In recent years, there have been several developments in the field of computing that have transformed the way we interact with technology. One of the most significant developments has been the evolution of programming languages, such as Java.

Java is a popular programming language that comes in different editions, including Standard, Enterprise, and Micro. Each edition has specific use cases, with Standard being used for desktop applications, Enterprise for web applications, and Micro for mobile devices and other embedded systems.

Along with programming languages, there have been significant developments in computer hardware and software. Computers are made up of several logical units, including the input, output, arithmetic and logical, memory, central processing, and secondary storage units. Each unit has a specific function, such as processing data, storing data, or interacting with users.

Data processed by a computer forms a hierarchy, from bits to characters to fields to records to files to databases to big data. This hierarchy is essential in understanding how data is stored, processed, and retrieved by computers.

Programming languages play a crucial role in writing instructions that computers can understand. Machine language is the language that computers understand on their own, while assembly and high-level languages require intermediate translation steps through the assembler and compiler, respectively.

Introduction to object-oriented programming is best understood by using an automobile as an example. Different concepts in OOP, such as classes, methods, instantiation, reuse, attributes, encapsulation, inheritance, interfaces, object-oriented analysis and design, modeling, and the use of the unified modeling language, were discussed.

Several operating systems are used for personal computers, including Windows and MacOS, which are proprietary operating systems, and Linux, which is open source. For other devices such as healthcare devices, game consoles, NASA satellites, and smartwatches, Android and IOS are the commonly used operating systems.

There are various programming languages available, and each has its strengths and weaknesses. Some languages, such as Java, are object-oriented, while others are used for other paradigms of programming, such as Functional programming, structured programming, and procedural programming.

Java language is based on object-oriented programming language developed by James Gosling that writes programs that will run on a great variety of computer systems and computer-controlled devices. The Java development environment is where we create and execute Java applications. This process occurs in five phases through specific commands; they include edit, compile, load, verify, and execute. Only through these processes can the Java application be executed