**Assignment 1**

1. What is JDK? JRE? JVM?

* The Java Development Kit (JDK) is a distribution of Java Technology by Oracle Corporation. It implements the Java Language Specification and the Java Virtual Machine Specification and provides the Standard Edition of the Java Application Programming Interface
* A Java runtime environment (JRE) is a set of components to create and run a Java application. A JRE is part of a Java development kit (JDK). A JRE is made up of a Java virtual machine (JVM), Java class libraries, and the Java class loader. JDKs are used to develop Java software; JREs provide programming tools and deployment technologies; and JVMs execute Java programs.
* A Java virtual machine (JVM) is a virtual machine that enables a computer to run Java programs as well as programs written in other languages that are also compiled to Java bytecode. The JVM is detailed by a specification that formally describes what is required in a JVM implementation.

1. What is java compiler?

Java Compiler and Interpreter are the most fundamental tools in Java language that programmers use during programming.

A compiler in Java is a computer program that is used for compiling Java programs. It is platform-independent. It converts (translates) source code (.java file) into bytecode (.class file).

In other words, the compiler (javac.exe) generates bytecode during the compilation process.

A bytecode is a binary code that is understood and interpreted by Java Virtual Machine (JVM) on the underlying operating system. It is not like machine code.

It is unreadable by humans because it is composed of numbers that are the only language that computers understand.

1. Why is java platform independent?

Java is platform-independent because it uses a virtual machine. The Java programming language and all APIs are compiled into bytecodes. Bytecodes are effectively platform-independent. The virtual machine takes care of the differences between the bytecodes for the different platforms.

1. What is IDE? Why is it important for developers?

An integrated development environment is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of at least a source code editor, build automation tools and a debugger.

An integrated development environment combines several of these development-related technologies into a single framework. When all utilities are represented on the same workbench, developers don't have to spend hours learning how to operate them separately. This is also handy for new developers who may use an IDE to learn about a team's standard tools and practices.

1. Is java case sensitive?

Java is a case-sensitive language, which means that the upper or lower case of letters in your Java programs matter.

1. What do the following key words do?  
   static, final, public, private, void, null, package, Class, new

* The static keyword in Java is mainly used for memory management. The static keyword in Java is used to share the same variable or method of a given class. The users can apply static keywords with variables, methods, blocks, and nested classes. The static keyword belongs to the class than an instance of the class.
* When a variable is declared with final keyword, its value can’t be modified, essentially, a constant. This also means that you must initialize a final variable. If the final variable is a reference, this means that the variable cannot be re-bound to reference another object, but the internal state of the object pointed by that reference variable can be changed i.e. you can add or remove elements from the final array or final collection.

When a class is declared with final keyword, it is called a final class. A final class cannot be extended(inherited).

When a method is declared with final keyword, it is called a final method. A final method cannot be overridden. The Object class does this—a number of its methods are final. We must declare methods with the final keyword for which we are required to follow the same implementation throughout all the derived classes.

* public is a Java keyword which declares a member's access as public. Public members are visible to all other classes. This means that any other class can access a public field or method. Further, other classes can modify public fields unless the field is declared as final.
* The private keyword is an access modifier used for attributes, methods and constructors, making them only accessible within the declared class.
* The void keyword specifies that a method should not have a return value.
* null indicates that a reference does not refer to anything
* Package in Java is a mechanism to encapsulate a group of classes, sub packages and interfaces.
* The class keyword is used to create a class. Every line of code that runs in Java must be inside a class. A class should always start with an uppercase first letter, and the name of the java file must match the class name.
* The new keyword creates new objects.

1. What is primitive type and reference type?

* Primitive Data Types. The eight primitives defined in Java are int, byte, short, long, float, double, boolean and char. These aren't considered objects and represent raw values. They're stored directly on the stack
* In Java, non-primitive data types are known as reference types. In other words, a variable of class type is called reference data type. It contains the address (or reference) of dynamically created objects.

1. Is parameter passed by value or reference?

In java, parameter is always passed by value.

1. What is the output: System.out.println(1 > 0 : “A”:”B”);

A

1. How to define constants in java?

To define a variable as a constant, we just need to add the keyword “final” in front of the variable declaration.

1. What is String? Is it primitive type?

In Java, string is an object that represents a sequence of characters. The java.lang.String class is used to create a string object.

The string data type is a non-primitive data type, but it is predefined in java, some people also call it a special ninth primitive data type.

1. How to check if a String is representing a number?

The easiest way of checking if a String is a numeric or not is by using one of the following built-in Java methods:

* Integer.parseInt()
* Integer.valueOf()
* Double.parseDouble()
* Float.parseFloat()
* Long.parseLong()

These methods convert a given String into its numeric equivalent. If they can't convert it, a NumberFormatException is thrown, indicating that the String wasn't numeric.

NumberUtils.isParsable() This method accepts a String and checks if it's a parsable number or not, we can use this method instead of catching an exception when calling one of the methods we mentioned earlier.

1. Write a program to implement the following activity diagram:

图示

描述已自动生成文本

描述已自动生成

文本

描述已自动生成

1. Write a program to merge two array of int.

文本

描述已自动生成

1. Write a program to find the second largest number inside an array of int.

文本

中度可信度描述已自动生成