/\* Steps to add javadoc for JRE System Library

javadoc path = D:\jdk-9.0.4\_doc-all.zip

Path for the javadoc zip file: http://www.oracle.com/technetwork/java/javase/documentation/jdk9-doc-downloads-3850606.html

Go to Window option in Eclipse->

Preferences->

Java ->

Installed JRES->

Select the JRE and click Edit->

Select one of the option in JRE system libraries->

Click Javadoc Location->

Select Javadoc in archive->

In Archive path select the path of zip file e.g. \jdk-9.0.4\_doc-all.zip->

In Path within archive Select docs/api->

Click Validate

\*/

# What is Java?

Java is a programming language and a platform.

Java is a high level, robust, secured and object-oriented programming language.

Platform: Any hardware or software environment in which a program runs, is known as a platform. Since Java has its own runtime environment (JRE) and API, it is called platform.

Java provides software-based platform.

# Features of Java



* + 1. Simple
    2. Object-Oriented: We organize our software as a combination of different types of objects that incorporates both data and behavior.

Basic concepts of OOPs are:

* + - * 1. Object
        2. Class
        3. Inheritance
        4. Polymorphism
        5. Abstraction
        6. Encapsulation
    1. Portable
    2. Platform independent



* + 1. Secured
       - 1. No explicit pointer
         2. Java Programs run inside virtual machine sandbox
    2. Robust:

Robust simply means strong. Java uses strong memory management. There are no pointers hence avoids security problem. There is automatic garbage collection in java. There is exception handling and type checking mechanism in java. All these points make java robust.

* + 1. Architecture neutral:

There are no implementation dependent features e.g. size of primitive types is fixed.

In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. But in java, it occupies 4 bytes of memory for both 32 and 64 bit architectures.

* + 1. Interpreted
    2. High Performance:

Java is faster than traditional interpretation since byte code is "close" to native code still somewhat slower than a compiled language (e.g., C++)

* + 1. Multithreaded

A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications etc.

# What happens at compile time?

At compile time, java file is compiled by Java Compiler (It does not interact with OS) and converts the java code into bytecode.



1. **How to set Permanent Path of JDK in Windows**

The path is required to be set for using tools such as javac, java etc.

If you are saving the java source file inside the jdk/bin directory, path is not required to be set because all the tools will be available in the current directory.

But if you are having your java file outside the jdk/bin folder, it is necessary to set path of JDK.

For setting the permanent path of JDK, you need to follow these steps:

Go to My Computer properties -> advanced tab -> environment variables -> System Variable ->

Select Variable = Path and click Edit. Append the bin directory path of JDK at the end

e.g. C:\Program Files\Java\jdk1.8.0\_151\bin

-> ok -> ok -> ok

To set Temporary Path of JDK in Windows run below in command prompt

**set path= C:\Program Files\Java\jdk1.8.0\_151\bin**

1. **How to set JAVA\_HOME path**

C:\Program Files\Java\jdk1.8.0\_151

1. **Difference between JDK, JRE and JVM**



**What is JVM**

It’s a specification where working of Java Virtual Machine is specified. But implementation provider is independent to choose the algorithm. Its implementation has been provided by Sun and other companies.

Its implementation is known as JRE (Java Runtime Environment).

Runtime Instance Whenever you write java command on the command prompt to run the java class, an instance of JVM is created.

What it does

1. The JVM performs following operation:
2. Loads code
3. Verifies code
4. Executes code
5. Provides runtime environment

JVM provides definitions for the:

1. Memory area
2. Class file format
3. Register set
4. Garbage-collected heap
5. Fatal error reporting etc.
6. **Variables and Data Types in Java**



1. Local Variable: A variable which is declared inside the method is called local variable.
2. Instance Variable: A variable which is declared inside the class but outside the method, is called instance variable. It is not declared as static
3. Static variable: A variable that is declared as static is called static variable. It cannot be local.

Example

**class** A {

**int** data = 50;// instance variable

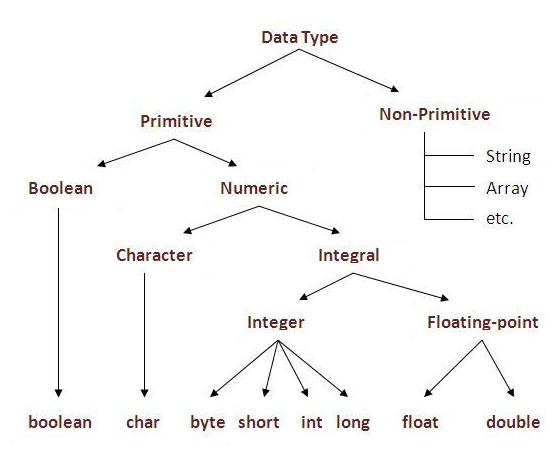
**static** **int** *m* = 100;// static variable

**void** method() {

**int** n = 90;// local variable

}

} // end of class



Solve Exercise1 for to understand DataType Conversion.