**How java really Works**

* Java compiler, which checks syntax
* After the compiler compiles the Java File byte code generated
* Java interpreter, which understands only byte code and converts it to native code
* Human readable code with proper java syntax, i.e. the Java File
* We see output same in background is native code
* Only the byte code is platform independent

**JVM, or Java Virtual Machine**

Implementation of JVM provides the environment to run our Java Applications.

**JDK & JRE (JAVA DEVLOPEMENT KIT & JAVA RUNTIME ENVIRONMENT)**

* It’s usually used for developing various application in the java.
* JDK includes the Java Runtime Environment, an interpreter/ loader, a compiler, an archive (jar file), a documentation generator.
* When you install the java set up or .exe file of java JDK is installed automatically

**JAR File**

* JAR stands for Java Archive.
* It's a file format based on the popular ZIP file format
* It is used for aggregating many files into one.
* JAR can be used as a general archiving tool
* Primary motive is download the requisite component in one HTTP transaction**.**

What is a **class** in Java?

A class is group of object which have common properties.

It is logical entity. It contains fields, constructor, methods, blocks, nested class and interface.

**Object:** An entity that has **state**, **behavior** and **identity** is known as object

**State** represents the data (value) of an object.

**Behavior** is functionality of an object (ex. Deposit, withdraw),

**Identity** is typically implemented via a unique id. It is used internally by JVM to identify unique id.

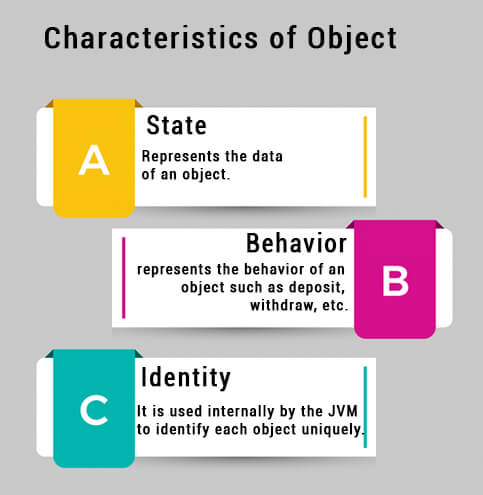
### Method in Java

In Java, a method is like a function which is used to expose the behavior of an object.

**Advantage of Method**

**Code Reusability**

**Code Optimization**



**Package**

A **package** in Java is a mechanism to contain classes, sub packages or interfaces.

We must have a proper folder structure from using of packages.

For the maintenance easily access of files.

**Method**

**Data types in java**

**Primitive and Non primitive**

**Primitive**

* **short**
* **int**
* **long**
* **float**
* **double**
* **char**
* **boolean**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **Type** | **Bit** | **Byte** | **Range** |
| **Integer value** | **Byte** | **8** | **1** | **-128 to 127** |
| **short** | **16** | **2** | **-32768 to 32767** |
| **int** | **32** | **4** | **2 31  to 2 31 -1** |
| **long** | **64** | **8** | **2 63  to 2 63 -1** |
| **Floating points** | **float** | **32** | **4** | **Stores fractional values sufficient for 5 to 6 decimal digits** |
| **double** | **64** | **8** | **Stores fractional values sufficient for 15 to 16 decimal digits** |
| **Unicode char** | **char** | **16** | **2** | **0 to 65536** |
| **Boolean** | **boolean** | **1** |  | **True or False** |

**Non Primitive Data Type**

* **Class**
* **Interface**
* **Array ( Object)**
* **String**

**Constructor**

Constructor is special type of method whose name is same as class name.

The main purpose of constructor is initialization of an object.

Every java class have constructor.

The constructor is automatically called at the time of object initialization.

The constructor never contains the return type including void.

We can write as much as constructor in the class with different parameters.

Whenever we call any method or variable in class constructor must be initialize.

When we call the parameterized constructor in class JVM cannot initialized default constructor.