

Architecture-neutral

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

Portable

We may carry the java bytecode to any platform.

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++)

Distributed

We can create distributed applications in java. RMI and EJB are used for creating distributed applications. We may access files by calling the methods from any machine on the internet.

Multi-threaded

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 shares the same memory. Threads are important for multi-media, Web applications etc.

Simple Program of Java

Creating hello java example

Let's create the hello java program:

```
1.      class Simple{
2.          public static void main(String args[]){
3.              System.out.println("Hello Java");
4.          }
5.      }
```

6. save this file as Simple.java

7.

To compile: javac Simple.java

To execute: java Simple

8. **Output:** Hello Java

Understanding first java program

Meaning of class, public, static, void, main, String[], System.out.println().

- **class** keyword is used to declare a class in java.
- **public** keyword is an access modifier which represents visibility, it means it is visible to all.
- **static** is a keyword, if we declare any method as static, it is known as static method. The core advantage of static method is that there is no need to create object to invoke the static method. The main method is executed by the JVM, so it doesn't require to create object to invoke the main method. So it saves memory.
- **void** is the return type of the method, it means it doesn't return any value.
- **main** represents startup of the program.
- **String[] args** is used for command line argument. We will learn it later.
- **System.out.println()** is used print statement. We will learn about the internal working of System.out.println statement later.

1) By changing sequence of the modifiers, method prototype is not changed.

static public void main(String args[])

2) subscript notation in java array can be used after type, before variable or after variable.

Different codes to write the main method.

1. **public static void** main(String[] args)
2. **public static void** main(String []args)
3. **public static void** main(String args[])

Valid java main method signature

1. **public static void** main(String[] args)
2. **public static void** main(String []args)
3. **public static void** main(String args[])
4. **public static void** main(String... args)
5. **static public void** main(String[] args)

Operators in java

There are many types of operators in java such as unary operator, arithmetic operator, relational operator, shift operator, bitwise operator, ternary operator and assignment operator.

Operators	Precedence
Postfix	<code>expr++ expr--</code>
Unary	<code>++expr --expr +expr -expr ~ !</code>
Multiplicative	<code>* / %</code>
Additive	<code>+ -</code>
Shift	<code><< >> >>></code>
Relational	<code>< <= > >= instanceof</code>
Equality	<code>== !=</code>
bitwise AND	<code>&</code>
bitwise exclusive OR	<code>^</code>
bitwise inclusive OR	<code> </code>
logical AND	<code>&&</code>
logical OR	<code> </code>
Ternary	<code>? :</code>
Assignment	<code>= += -= *= /= %= &= ^= = <<= >>= >>>=</code>