

Introduction to Programming with Java

Concepts

- What is Java?
- Hello World Program in Java
- Arithmetic Operations in Java
 - Addition
 - Subtraction
 - Multiplication
 - Division
 - Modulo Operation
- Variables
 - Declaring a Variable
 - Initializing a Variable
 - Defining a Variable
 - Type of the variable

1. Introduction

In this course, you will be introduced to a new programming language called **Java**. The following sessions assume that you understand the fundamental concepts of programming and are comfortable writing code in Python. This is a fast-paced course compared to **Programming Foundations with Python** where the fundamentals are explained in much more detail. So, please revise our Python course once, before proceeding further.

1.1 What is Java?

Java is a popular procedural and object-oriented programming language. It was created in 1995 and is owned by Oracle.

Java is a secure, fast, and powerful programming language. Java works on different platforms (Windows, Mac, Linux, etc.). It has huge community support (tens of millions of developers).

1.2 Why learn another Programming Language?

Python is an extremely good language to start learning programming for various reasons. It has a simple syntax and a powerful standard library. This simplicity and power of Python have attracted a huge developer community who have built a wide variety of third-party libraries that simplify a lot of programming tasks.

So this brings us to the question, why learn another programming language?

Different programming languages have different strengths. Some languages are good for quickly writing small scripts, some are better suited for working on a large-scale project involving multiple developers. The same feature of the programming language which brings certain advantages in some aspects may also bring disadvantages in other aspects.

Python - A Dynamic Programming Language

Python belongs to a class of programming languages called dynamic languages. The syntax of this category of languages is considered a bit informal. Informal in the sense that a data type of a variable or return type of a function/method, etc., is never explicit or formally pre-defined. You can directly assign a value to a variable and later assign a value of a different type to the same variable. This informality simplifies the syntax and makes it easy to write Python code. But this also means the programmer has to remember certain things. It is up to the developer to ensure that the right type of value exists in a variable when a particular method is called. If the developer misses these types of things, the code will run into exceptions during runtime.

Java - A Static Programming Language

On the other hand, we have programming languages like C, C++, and Java that are more formal. These are often referred to as static programming languages. These languages protect you by forcing you to be upfront and formal about the kind of object each variable is going to contain in its lifetime. As the developer mentions the data types of the variables upfront, before using them, these languages could use this knowledge to

- report some of the possible errors to the developer before the code is executed.
- optimize functionalities to different data types and thereby increase performance and speed of execution.

You may even notice a performance improvement of 5x - 10x when the same logic is written in Java or C++ instead of Python.

As a developer, you should understand the strengths of different languages and should be able to use the appropriate language as per the needs of the project at hand. There are certain features like variables, conditionals, loops, functions, etc. that most programming

languages have in common. So when learning a new programming language you should focus on differences in syntax. You should try to compare different aspects of the new programming language with different programming languages that you already know to understand what is common among them and what are the key differences.

1.3 Why learn Java instead of C/C++?

For a Python programmer, it is easier to learn Java when compared to C++

- Java includes a larger standard library than C/C++
- The syntax of Java is less complicated than the syntax of C++

Java is usually considered to be halfway between C/C++ and Python in terms of performance, and ease of programming. This balance has made Java an attractive option for many enterprises. Java was considered the most popular programming language for more than 20 years (until the early 2020s).

With Java, it is possible to easily develop Cross-Platform applications. The same Java application can be run on different Operating Systems like Windows, Ubuntu, etc. which is not possible with C/C++. (You will learn more about this later.)

1.4 Applications & Career Opportunities with Java

Java can be used to develop a wide variety of applications.

- Mobile Applications
- Web-based and Desktop Applications
- Server-Based Applications, etc.

As Java has been the primary language of choice for many enterprises for quite some time, there are multiple job roles that require developers who can program in Java.

- Java Developer
- Java Architect
- Java Web Developer
- Java Android Developer, etc.

2. Hello World Program in Java

Below is the code to display the message **"Hello World!"** in Java.

Code

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

JAVA

Output

```
Hello World!
```

Java is a programming language that incorporates the concepts of Object-Oriented programming to the fullest. Every Java program must define a new class. The execution of the program starts from a method named

`main` in this class. You can directly use other predefined classes or import classes from the standard library. Now let us dive into the above Java code.

- In the above Hello World program,

`HelloWorld` is the name of the class. You can define a new class using the `class` keyword.

- **Syntax**

```
1 class NameOfTheClass {
2
3 }
```

JAVA

- **Naming Convention:** In both Java and Python, the programmers generally follow *pascal case* notation for naming classes.

- We define the method and attributes of that class within the curly braces

`{}` .

- **Note:** Just as we use indentation in Python to mention that the following code belongs to class definition or method definition.

We use curly braces

`{}` in Java.

- In the line,

`public static void main(String[] args)` we are defining a method with the name `main`. Every Java application must consist of a method called `main`. It serves as an entry point for the application (The code execution begins from the `main` method). The next set of curly braces contains the implementation of the `main` method.

We will learn the meanings of the keywords

`public`, `static`, `void`, and `String[]` later in the course.

- In Java,

`System.out.println()` statement is used to print a line of output to the console.

- `System` is a class from the core library provided by Java.
- `out` is an object that controls the output.
- `println()` is a method associated with that object that receives a single argument.

3. Arithmetic Operations in Java

3.1 Addition

Addition is denoted by

`+` sign. It gives the sum of two numbers.

Code

JAVA

```
1 class Addition {  
2     public static void  
3         System.out.  
4         System.out.  
5     }  
6 }
```

Output

```
7  
2.5
```

3.2 Subtraction

Subtraction is denoted by

- sign. It gives the difference between the two numbers.

Code

JAVA

```
1 class Subtraction {  
2     public static void  
3         System.out.
```



```
4 }  
5 }
```

Output

```
3
```

3.3 Multiplication

Multiplication is denoted by

`*` sign. It gives the product of two numbers.

Code

```
1 class Multiplicatio  
2     public static v  
3         System.out.  
4         System.out.  
5     }  
6 }
```

JAVA

Output

```
10
2.5
```

3.4 Division

Division is denoted by

/ sign. It returns the quotient as a result.

In contrast to Python which returns a

float, Java returns an integer. During division, Python internally converts the integer values to float data type. The conversion of one data type to another data type is called **Type Conversion**.

We will learn more about **Type Conversions** later in the course.

Code

```
1 class Division {
2     public static void
3         System.out.
4         System.out.
5     }
6 }
```

JAVA

Output

```
2  
2
```

► Comparison with Python

3.5 Modulo Operation

Modulo is denoted by

`%` sign. It returns the modulus (remainder) as a result.

Code

```
1 class Modulus {  
2     public static void main(String[] args) {  
3         System.out.println("Modulo operation");  
4         System.out.println("2 % 2 = " + (2 % 2));  
5     }  
6 }
```

JAVA

Output

```
1
```

4. Variables

Variables are like containers for storing information.

4.1 Declaring a Variable

A Variable is created when it is declared, and to declare a variable, we must specify the type of data first, followed by the variable name.

Syntax

```
1 dataType variableName;
```

JAVA

- **dataType:** It specifies the type of information we are storing in the variable.
- **variableName:** It will say about the name given to the variable for storing that information.

The Java programming language is statically-typed, which means that all variables must first be declared before they can be used.

For example, to store information of an integer we need to declare a variable using the keyword

int

Later in this module, we will learn different data types that Java supports to store information

Naming Convention: In Java, the programmers generally follow camel case notation. Whereas in Python, snake case notation is followed.

► Comparison with Python

4.2 Initializing a Variable

Assigning a value to a variable for the first time is called **Initialization**. Memory is allocated when a variable is initialized.

Code

```
1 class Main {
2     public static \
3         int points;
4         points = 90;
5     }
6 }
```

JAVA

A variable is only a name given to a memory location. All the operations done on the variable affect that memory location

4.3 Defining a Variable

Declaring and initializing a variable in a single statement is called Defining a variable.

Code

JAVA

```
1 class Main {  
2     public static  
3         int points;  
4         points = 90;  
5         int totalPoints;  
6  
7         System.out.println("Points: " + points);  
8         System.out.println("Total Points: " + totalPoints);  
9     }  
10 }
```

Output

```
90  
100
```

► Comparison with Python

Java executes the code line-by-line

Code

```
1 class Main {  
2     public static v  
3         System.out.  
4         int points  
5     }  
6 }
```

Output

```
file.java:3: error: c  
        System.out.pr  
        ^  
        symbol:   variable  
        location: class Mai  
1 error
```

- Variable `points` is not created by the time we tried to print.
- As Java programming language is a statically typed language. So before using the `points` variable we must declare it.

4.4 Type of the variable

The type of a variable cannot be changed once a variable is declared.

Code

```
1 class Main {  
2     public static v  
3         int points =  
4         String point  
5  
6         System.out.p  
7     }  
8 }
```

Output

```
Main.java:4: error: v  
    String points  
    ^  
1 error
```

As Java programming language is a statically typed language, we cannot change the type of the variable once it is declared.

► Comparison with Python

Summary

- In Java,

`System.out.println()` statement is used to print a line of output to the console.

- **Arithmetic Operations**

- Addition is denoted by `+` sign.
 - Subtraction is denoted by `-` sign.
 - Multiplication is denoted by `*` sign.
 - Division is denoted by `/` sign.
 - Modulo is denoted by `%` sign.
- To declare a variable, we must specify the type of data first, followed by the variable name.