



# Programming Mindset

by Sophia



## WHAT'S COVERED

In this lesson, you will learn how to think like a programmer, and you will understand what's involved in the development process. You will also learn reasons why studying programming is important, along with the advantages of learning to program in Java. Specifically, this lesson covers:

- [\*\*1. Introduction to Java Programming\*\*](#)
- [\*\*2. Why Study Programming?\*\*](#)
  - [\*\*2a. Gain New Skills\*\*](#)
  - [\*\*2b. Advantages of Learning Java\*\*](#)

## 1. Introduction to Java Programming



Introduction to Java Programming will provide you with a comprehensive overview of object-oriented programming using the Java language. When considering the purpose of this course, some important questions might come to mind: Why study programming? Why study Java? What is object-oriented programming? This tutorial will address these questions. First, you will learn about the reasons why someone would choose to study programming. You will examine the advantages of the Java programming language. Finally, you will explore object-oriented programming and why Java is considered an object-oriented programming language.

Programming is a creative and rewarding activity. You can write programs for many reasons, from making a living to solving a difficult data analysis problem that helps someone else solve a problem for fun. This course assumes that everyone can learn how to program, and that once you know how to program, you will figure out what you want to do with your newfound skills.



### BIG IDEA

Throughout the rest of this course, you will explore what it means to think like a programmer. You will learn to foster curiosity and creativity along the way. In the end, you will be a *programmer*—perhaps not a professional programmer, but you will have the skills to analyze a problem and develop a program to solve it.

## 2. Why Study Programming?

Computer programming is the art and science of designing and writing computer programs. A computer **program** is a set of instructions that directs the computer's behavior. It might be easiest to understand what a program is by thinking about a problem that a program might solve.

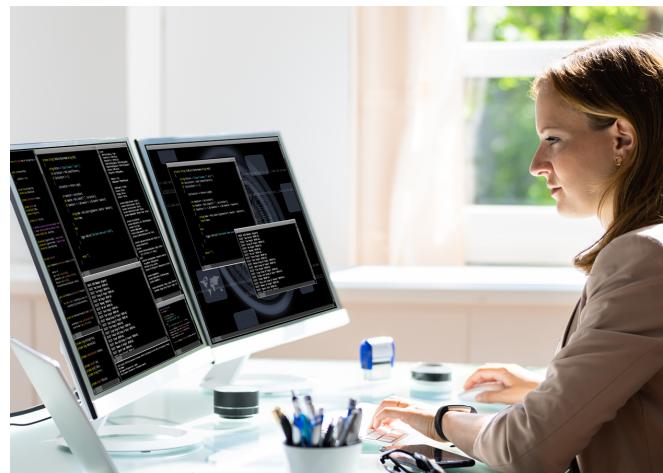


### THINK ABOUT IT

Look at the first three paragraphs of this tutorial and determine the most commonly used word and how many times the word is used. While you were able to read and understand the words in a few seconds, counting them is almost painful because it is not the kind of problem that human minds are designed to solve. For a computer, the opposite is true. Reading and understanding text from a piece of paper is hard for a computer to do, but counting the words and telling you how many times the most used word was used is very easy for the computer.

Early computers required users to understand how to issue commands in a specific programming language. It was widely believed that entrance into the computer age would require practically everyone to learn how to program, but this did not prove to be true. Modern computers no longer require us to learn programming skills to perform everyday tasks.

Today, a person may choose to study programming in order to enter into a career as a computer scientist. However, there are many computer scientists who do little or no programming as part of their everyday work. If you plan to major or minor in computer science, you will certainly learn to program, but good careers in the computing field are available to programmers and nonprogrammers alike.



### DID YOU KNOW

One of the best reasons to study programming is because it is potentially a creative and enjoyable problem-solving activity. This course will teach you how to develop well-designed solutions to a range of interesting problems. You will see and experience your solutions as running programs.

As many learners have indicated, there's really nothing like the feeling you get from seeing your program solve a problem you've been struggling with. Designing and building well-written programs provides a powerful sense of accomplishment and satisfaction. What's more, Java is a language that makes programming even more fun, because once they're finished, many Java programs can be posted on the World Wide Web (WWW) for all the world to see!

## 2a. Gain New Skills

You need two skills to be a programmer:

- First, you need to become fluent in the vocabulary and sentence structure of the programming language—in this case, using Java. Learning to program is very similar to learning a foreign language. It's important to

know the vocabulary, which in this case, refers to using Java keywords and method names, and the grammar, or the syntax, of this new language.

- Second, you need to think like a programmer. Programmers are curious and creative problem solvers.

Once you learn one programming language, like Java, you will find it much easier to learn another programming language, like Python, C++ or C#. Each programming language has a different vocabulary and grammar, but the problem-solving skills will be the same across all programming languages.

The “vocabulary” and “sentences” of Java should come fairly quickly to you. However, it will take longer for you to write a coherent program to solve a brand-new problem. Programming is often taught like writing. We start by reading and explaining programs, then write simple programs, and then write increasingly complex programs over time. At some point, you “get your muse” and see the patterns on your own and can see more naturally how to take a problem and write a program that solves that problem. Once you get to that point, programming becomes a very pleasant and creative process.

Thinking like a programmer involves the breaking down of a problem, or the solution to a problem, into small, unambiguous steps that a machine can carry out. It also involves understanding how the vocabulary and syntax of programming language can be used to map out and implement those steps. Different programming languages bring different strengths and tools to the process. This means that programmers will typically need working knowledge of two or more programming languages to be efficient. Learning Java as one of those languages will help you.



Thinking about your goals will help you prioritize which languages you choose to study. For example, Java is a popular choice among programmers interested in web development and Android-specific app development. Python is often preferred by programmers working in data science, back-end development, and machine learning. If you are interested in learning more about Python, consider earning credits toward your degree with Sophia's Introduction to Python Programming course.

## 2b. Advantages of Learning Java

Java is a relatively young programming language. It was initially designed by Sun Microsystems in 1991 as a language for embedding programs into electronic consumer devices, such as microwave ovens and home security systems. Java has since become one of the most important languages for Web and Internet applications. Because of its originally intended role as a language for programming microprocessors embedded in consumer appliances, Java has been designed with a number of interesting features. These include:

- Designing Java from scratch as an object-oriented language. **Object-oriented programming languages** divide programs into separate modules, called objects, that encapsulate the program’s data and operations. Thus, object-oriented programming (OOP) and object-oriented design (OOD) refer to a particular way of organizing programs.
- Designing Java to be robust. Meaning that errors in Java programs don’t cause system crashes as often as errors in other programming languages. Certain features of the language enable many potential errors to be detected before a program is run.

- Ensuring that Java is **platform independent**. A platform, in this context, is just a particular kind of computer system, such as a MacOS or Windows system. Java's trademark is "Write once, run anywhere." This means that a Java program can be run without changes on different kinds of computers. This is not true for other high-level programming languages. This portability—the ability to run on virtually any platform—is one reason that Java is well suited for internet applications.
- Developing Java with a **distributed language**, which means that its programs can be designed to run on computer networks. In addition to the language itself, Java comes with an extensive collection of code libraries—software that has been designed to be used directly for particular types of applications—that make it particularly easy to build software systems for the Internet. This is one of the reasons why Java is so well-suited for supporting applications on corporate networks.
- Designing Java as a secure language. Designed to be used on networks, Java contains features that protect against untrusted code—code that might introduce a virus or corrupt your system in some way.

Additionally, Java is a good programming language to learn because it is a widely used, high-level, general purpose programming language. It can be used to develop a wide range of programs from desktop programs to mobile and distributed applications. Java is also a cross-platform language. Thanks to the Java Virtual Machine (JVM), Java allows development and compiling on one platform for deployment on all platforms on which Java is supported.

Let's say you are doing social computing research on Facebook posts and you are interested in the most frequently used word in a series of posts. You could print out the stream of Facebook posts and pore over the text looking for the most common word, but that would take a long time and be very mistake-prone. You would be smart to write a Java program to handle the task quickly and accurately so you can spend the weekend doing something fun.

Java is a way for us to exchange useful sequences of instructions (i.e., programs) in a common format that can be used by anyone who has Java installed on his or her computer. So, neither of us is talking to *Java*; instead, we are communicating with each other *through Java*.



### BIG IDEA

There are few other languages in which a beginning programmer can write a computer game or a graphically based application that can be distributed on a Web page to just about any computer in the world. The simplicity of Java's design and its easily accessible libraries bring such accomplishments within reach of the most novice programmers.



### TERMS TO KNOW

#### Program

A sequence of computer language statements that have been crafted to do something.

## **Object-Oriented Programming Language (OOP)**

A particular way of organizing programs by dividing them into separate modules, called objects, that encapsulate the program's data and operations.

## **Platform Independent**

A Java program can be run on virtually any computer system platform including MacOS and Windows.

## **Distributed Language**

A programming language, like Java, that is designed to run on computer networks.



### SUMMARY

In this lesson, you were **introduced to programming with Java**. You learned why the **study of programming** is important—such as **gaining new skills**—and the **advantages of learning Java programming** over other languages.

Source: This content and supplemental material has been adapted from Java, Java, Java: Object-Oriented Problem Solving. Source [cs.trincoll.edu/~ram/jjj/jjj-os-20170625.pdf](http://cs.trincoll.edu/~ram/jjj/jjj-os-20170625.pdf)

It has also been adapted from “Python for Everybody” By Dr. Charles R. Severance. Source [py4e.com/html3/](http://py4e.com/html3/)



### TERMS TO KNOW

#### **Distributed Language**

A programming language, like Java, that is designed to run on computer networks.

#### **Object-oriented Programming Language (OOP)**

A particular way of organizing programs by dividing them into separate modules, called objects, that encapsulate the program's data and operations.

#### **Platform Independent**

A Java program can be run on virtually any computer system platform including MacOS and Windows.

#### **Program**

A sequence of computer language statements that have been crafted to do something.