Intro to Java programming

Learn Java language basics and master the constructs for building and deploying real-world applications

J Steven Perry

Learn the structure, syntax, and programming paradigm of the Java platform and language. Start by mastering the essentials of object-oriented programming on the Java platform, and progress incrementally to the more-sophisticated syntax and libraries that you need in order to develop complex, real-world Java applications.

- **Skill level**: Beginner
- **Cost**: Free
- **Time commitment**: 15 - 30 minutes per unit (10 to 12 hours, total)
- **Get started**: Unit 1

This developerWorks learning path is for software developers who are new to Java™ technology and want to become productive Java programmers. Work through the 23-unit sequence, including videos and quizzes, to get up and running with object-oriented programming (OOP) and real-world application development using the Java language and platform.

Step-by-step instructions and video demos help you create and build on a simple Java object, developing it into a full application that — in the final unit — you deploy as a web app in the cloud. All along the way, you can test your newly acquired Java expertise with short quizzes and programming challenges.

To view this video, **Author's introduction**, please access the online version of the article. If this article is in the developerWorks archives, the video is no longer accessible.

**Skills you'll gain**

- Gain an understanding of the basics of OOP on the Java platform
- Have a fully functional Java development environment that uses the Eclipse IDE
- Become familiar with Java syntax and essential libraries
- Be ready to learn more-complex programming Java techniques
- Know where to find curated resources for bolstering your Java programming knowledge
System requirements

• A system supporting Java SE 8 with at least 2GB of memory. Java 8 is supported on Linux®, Windows®, Solaris®, and Mac OS X
• At least 200MB of disk space to install the software components and examples

You also need to install and set up a development environment consisting of JDK 8 from Oracle and the Eclipse IDE. Follow the download and installation instructions provided in the learning path.

Units in this learning path

Unit 1: Java platform overview

Learn the function of each of the Java platform's constituent components, see how the Java language is structured, and become familiar with navigating the Java API documentation.

Start Unit 1

Unit 2: Setting up your Java development environment

Install the Java Development Kit and the Eclipse IDE, become familiar with the main Eclipse components, and create a new Java project.

Start Unit 2

Unit 3: Object-oriented programming concepts and principles

Get an introduction to OOP concepts and understand the benefits of the OOP paradigm.
Unit 4: Getting started with the Java language

Recognize the reserved words in the Java language and learn the function and syntax for each construct within a Java class.

Unit 5: Your first Java class

Create a package and declare a class, add variables and methods to your class, use the Eclipse code generator, and test your class with the JUnit framework from within Eclipse.

Unit 6: Adding behavior to a Java class

Learn the syntax for accessor method declarations and method calls.
Unit 7: Strings and operators
Learn how to instantiate and manipulate strings, do string concatenation and method chaining, and explore arithmetic operators.

Start Unit 7

Unit 8: Conditional operators and control statements
Use relational operators, conditional operators, and control statements for decision making.

Start Unit 8

Unit 9: Loops
Iterate over code or execute it repeatedly.

Start Unit 9

Unit 10: Java collections
Create and manage collections of objects.

Start Unit 10
Unit 11: Archiving Java code
Import other developers' code and share yours.

Start Unit 11

Unit 12: Writing good Java code
Learn best practices for writing clean, easily maintainable Java code.

Start Unit 12

Unit 13: Next steps with objects
Enhance Java classes via method overloading, method overriding, and more.

Start Unit 13

Unit 14: Exceptions
Use built-in Java platform mechanisms (checked and unchecked exceptions) to handle errors in your code.

Start Unit 14
Unit 15: Building Java applications
Create applications from collections of objects.
Start Unit 15

Unit 16: Inheritance
Enhance code reuse by deriving classes from other classes.
Start Unit 16

Unit 17: Interfaces
Learn the purpose of an interface, how to use one, and how to implement one.
Start Unit 17

Unit 18: Nested classes
Learn how to define tightly coupled classes, and consider the advantages and side effects.
Start Unit 18
Unit 19: Regular expressions

Describe and search for string patterns in your Java code.

Start Unit 19

Unit 20: Generics

Facilitate code reuse by defining classes with abstract type parameters.

Start Unit 20

Unit 21: I/O

Collect and manipulate external data in your Java programs.

Start Unit 21

Unit 22: Java serialization

Store object state in binary format for object remoting or object persistence.

Start Unit 22
Related topic

- IBM Code: Java