Java Programming: Solving Problems with Software

Duke University

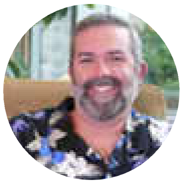
About this Course

Learn to code in Java and improve your programming and problem-solving skills. You will learn to design algorithms as well as develop and debug programs. Using custom open-source classes, you will write programs that access and transform images, websites, and other types of data. At the end of the course you will build a program that determines the popularity of different baby names in the US over time by analyzing comma separated value (CSV) files.

After completing this course you will be able to:
1. Edit, compile, and run a Java program;
2. Use conditionals and loops in a Java program;
3. Use Java API documentation in writing programs.
4. Debug a Java program using the scientific method;
5. Write a Java method to solve a specific problem;
6. Develop a set of test cases as part of developing a program;
7. Create a class with multiple methods that work together to solve a problem; and
8. Use divide-and-conquer design techniques for a program that uses multiple methods.

⌃ Show less

Taught by: Owen Astrachan, Professor of the Practice
Computer Science



Taught by: Robert Duvall, Lecturer
Computer Science

Taught by: Andrew D. Hilton, Assistant Professor of the Practice

Electrical and Computer Engineering

Taught by: Susan H. Rodger, Professor of the Practice

Computer Science

Basic Info	Course 1 of 4 in the Object Oriented Programming in Java Specialization
Level	Beginner
Commitment	4 weeks of study, 4-8 hours/week
Language	English, **Subtitles:** Turkish Volunteer to translate subtitles for this course
How To Pass	Pass all graded assignments to complete the course.
User Ratings	★★★★⯪ 4.5 stars

Syllabus

WEEK 1

Introduction to the Course

Welcome to “Java Programming: Solving Problems with Software”! We are excited that you are starting our course to learn how to write programs in Java, one of the most popular programming languages in the world. In this introductory module, you will get to meet the instructor team from Duke University and have an overview of the course. Have fun!

5 videos, 1 reading

1. **Video:** Introduction to the Course
2. **Reading:** Programming Resources
3. **Video:** Resources to Help You Succeed
4. **Video:** Tips for Learning Programming
5. **Video:** Using Forums: How to Ask for Help Effectively
6. **Video:** Object Oriented Programming with Java Specialization

Show less

Fundamental Java Syntax and Semantics

In this module, you will learn to write and run your first Java programs, including one program that prints “Hello!” in various countries’ languages and another where you will analyze the perimeters and other information of shapes. To accomplish these tasks, you will learn the basics of Java syntax and how to design stepwise solutions with programs. By the end of this module, you will be able to: (1) Download and run BlueJ, the Java programming environment for this course; (2) Access the documentation for the Java libraries specially designed for this course; (3) Edit, compile, and run a Java program; (4) Construct methods, variables, if else statements, and for each loops in Java; and (5) Use Iterables (like DirectoryResource) to run a program that iterates over multiples lines in a document or webpage or multiple files in a directory.

^Show less

17 videos, 4 readings, 4 practice quizzes

1. **Reading:** Module Learning Outcomes
2. **Video:** Why Use Java?
3. **Reading:** Download BlueJ and Open Your First BlueJ Project
4. **Video:** Using BlueJ to Program in Java
5. **Practice Quiz:** Getting Started with BlueJ
6. **Video:** Shapes: Collections of Points
7. **Video:** Why Semantics: Motivation to Read Code
8. **Video:** Variables
9. **Video:** Mathematical Operators
10. **Practice Quiz:** Variables and Mathematical Operators
11. **Video:** Functions
12. **Video:** Conditionals

13. **Practice Quiz:** Functions and Conditionals
14. **Video:** Classes
15. **Video:** New
16. **Video:** Methods
17. **Video:** Types
18. **Video:** For Each Loops
19. **Practice Quiz:** Classes, Types, and For Each Loops
20. **Video:** Solving Programming: A Seven Step Approach
21. **Video:** Seven Steps in Action: Developing an Algorithm
22. **Video:** Seven Steps in Action: Testing the Algorithm
23. **Video:** Seven Steps in Action: Translating to Code
24. **Reading:** Calculating the Perimeter of a Shape
25. **Reading:** End of Module Survey

Show less

Graded: Calculating the Perimeter of a Shape

WEEK 2

Strings in Java

This module begins with a short presentation from Raluca Gordân, an assistant professor in Duke University's Center for Genomic and Computational Biology, about an important problem genomics scientists encounter regularly: how to identify genes in a strand of DNA. To tackle this problem, you will need to understand strings: series of characters such as letters, digits, punctuation, etc. After learning about Java methods that work with strings, you will be able to find genes within a DNA string as well as tackle other string related problems, such as finding all of the links in a web page. By the end of this module, you will be able to: (1) Use important methods for the Java String class; (2) Use conditionals, for loops, and while loops appropriately in a Java program; (3) Find patterns in the data represented by strings to help develop the algorithm for your program; (4) Understand the importance of designing programs that keep different data processing steps separate; (5) Use the StorageResource iterable for this course to store some data for further processing; and (6) Rely on Java documentation to better understand how to use different Java packages and classes.

Show less

21 videos, 4 readings, 5 practice quizzes

1. **Video:** What is a String
2. **Video:** Understanding Strings
3. **Video:** Developing an Algorithm
4. **Video:** Positions in Strings
5. **Video:** Translating into Code
6. **Video:** Java Math
7. **Reading:** Programming Exercise: Finding a Gene and Web Links
8. **Practice Quiz:** Finding a Gene in DNA
9. **Video:** Introduction
10. **Video:** Conceptual Understanding
11. **Video:** While Loops
12. **Video:** While Loop Syntax and Semantics
13. **Video:** Coding While Loops
14. **Video:** Three Stop Codons
15. **Video:** Coding Three Stop Codons - Part I
16. **Video:** Coding Three Stop Codons - Part II
17. **Video:** Logical And / Or
18. **Video:** Coding And / Or
19. **Video:** Finding Multiple Genes
20. **Video:** Translating to Code
21. **Reading:** Programming Exercise: Finding Many Genes
22. **Practice Quiz:** Finding All Genes in DNA
23. **Practice Quiz:** Debugging: Part 1
24. **Practice Quiz:** Debugging: Part 2
25. **Discussion Prompt:** Debugging First Steps
26. **Video:** Separation of Concerns
27. **Video:** StorageResource Class
28. **Video:** Coding StorageResource Class
29. **Reading:** Programming Exercise: Storing All Genes
30. **Practice Quiz:** Using StorageResource
31. **Reading:** End of Module Survey

Show less

Graded: Strings in Java

WEEK 3

CSV Files and Basic Statistics in Java

A common format for storing tabular data (any data organized into columns and rows) is in comma separated values (CSV) files. In this module, you will learn how to analyze and manipulate data from multiple CSV data files using a powerful open-source software package: Apache Commons CSV. Using this library will empower you to solve problems that could prove too complex to solve with a spreadsheet. By the end of this module, you will be able to: (1) Use the open-source Apache Commons CSV package in your own Java programs; (2) Access data from one or many CSV files using Java; (3) Convert strings into numbers; (4) Understand how to use “null” in Java programs (when you want to represent “nothing”); (5) Devise an algorithm (and implement in Java) to answer questions about CSV data; and (6) Analyze CSV data across multiple CSV files (for example, find maximums, minimums, averages, and other simple statistical results).

^Show less

14 videos, 4 readings, 2 practice quizzes

1. **Reading:** Module Learning Outcomes
2. **Video:** CSV Data: Comma Separated Values
3. **Video:** Using CSV Libraries
4. **Video:** Which Countries Export...? Developing an Algorithm
5. **Video:** Which Countries Export...? Translating into Code
6. **Video:** CSVExport: Summary
7. **Reading:** Programming Exercise: Parsing Export Data
8. **Practice Quiz:** Which Countries Export...?
9. **Video:** Hottest Day in a Year: Comma Separated Values
10. **Video:** Converting Strings to Numbers
11. **Video:** Maximum Temperature: Developing an Algorithm
12. **Video:** Java for Nothing—null: When You Don't Have an Object
13. **Video:** Maximum Temperature: Translating into Code
14. **Video:** Maximum Temperature: Testing Code

15. **Video:** Maximum Temperature from Multiple Datasets
16. **Video:** Maximum Temperature Refactored
17. **Video:** CSVMax: Summary
18. **Reading:** Programming Exercise: Parsing Weather Data
19. **Practice Quiz:** Weather Data
20. **Reading:** End of Module Survey

Show less

Graded: CSV Files and Basic Statistics in Java

WEEK 4

MiniProject: Baby Names

This module wraps up the course with a mini project that ties together the different practices, skills, and libraries you have gained across the course! Using data on the popularity of different baby names in the United States from the past several decades, you will be able to compare different names' popularity over time. While the data we have collected for this course is from the United States, we welcome you to share data from other countries in the course discussion forums. Good luck with the mini project!

9 videos, 4 readings

1. **Video:** Baby Names MiniProject: Overview
2. **Video:** Baby Names MiniProject: Data Overview
3. **Video:** Baby Names MiniProject: Total Births
4. **Reading:** MiniProject Exercise Guide
5. **Reading:** Extend Your Program
6. **Reading:** End of Module Survey
7. **Video:** Batch Grayscale: Converting Many Files
8. **Video:** Grayscale Algorithm: Seven Step Approach
9. **Video:** Image Iterable in BlueJ: Grayscale
10. **Video:** Batch Processing Grayscale
11. **Video:** Saving Images with New Names
12. **Video:** Batch Grayscale Summary: Converting Many Files

13. **Reading:** Programming Exercise: Batch Grayscale and Image Inversion

Show less

Graded: Baby Names

Graded: Batch Grayscale Images

View Less

How It Works

GENERAL

How do I pass the course?

To earn your Course Certificate, you'll need to earn a passing grade on each of the required assignments—these can be quizzes, peer-graded assignments, or programming assignments. Videos, readings, and practice exercises are there to help you prepare for the graded assignments.

What do start dates and end dates mean?

Most courses have sessions that run multiple times a year — each with a specific start and end date. Once you enroll, you'll have access to all videos, readings, quizzes, and programming assignments (if applicable). Peer-graded assignments can only be submitted and reviewed once your session has begun. If you choose to explore the course without purchasing, you may not be able to access certain assignments. If you don't finish all graded assignments before the end of the session, you can enroll in the next session. Your progress will be saved and you'll be able to pick up where you left off when the next session begins.

What are due dates? Is there a penalty for submitting my work after a due date?

Within each session there are suggested due dates to help you manage your schedule and keep coursework from piling up. Quizzes and programming assignments can be submitted late without consequence. However, it is possible that you won't receive a grade if you submit your peer-graded assignment too late because classmates usually review assignment within three days of the assignment deadline.

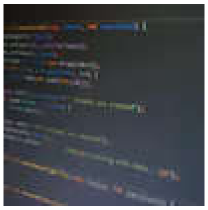
Can I re-attempt an assignment?

Yes. If you want to improve your grade, you can always try again. If you're re-attempting a peer-graded assignment, re-submit your work as soon as you can to make sure there's enough time for your classmates to review your work. In some cases you may need to wait before re-submitting a programming assignment or quiz. We encourage you to review course material during this delay.

^ Show less

Course 1 of Specialization

Grow Your Portfolio as a Software Engineer. Learn about Object Oriented Design in four project-based courses.

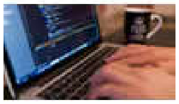


Object Oriented Programming in Java

University of California, San Diego, Duke University

View the course in catalog

Related Courses

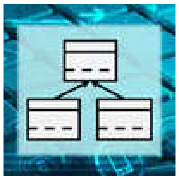


Java Programming: Arrays, Lists, and Structured Data

Duke University

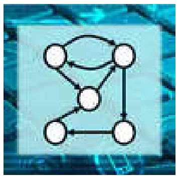
Java Programming: Principles of Software Design

Duke University



Object Oriented Programming in Java

University of California, San Diego



Advanced Data Structures in Java

University of California, San Diego

Data Structures and Performance

University of California, San Diego