Java SE 8 Fundamentals Ed 1

This Java SE 8 Fundamentals training enables those with little or no programming experience to learn object-oriented programming using the Java language. It provides a solid foundation to build upon throughout your software development career.

Objectives

- Write Java code that uses variables, arrays, conditional and loop constructs
- Manipulate primitive numeric data and string data using Java operators
- Create Java classes and use object references
- Access the fields and methods of an object
- Manipulate text data using the methods of the String and StringBuilder classes
- Use casting without losing precision or causing errors
- Declare, override, and invoke methods
- Access and create static fields and methods
- Use classes from the java.time and java.time.format packages to format and print the local date and time
- Encapsulate a class using access modifiers and overloaded constructors
- Define and implement a simple class hierarchy
- Demonstrate polymorphism by implementing a Java Interface
- Use a Predicate Lambda expression as the argument to a method

Topics

- Introduction to Computer Programs
- Key Features of the Java Language
- The Java Technology and Development
- Environment Running/testing a Java program
- Java Classes
- The main Method
- Introducing variables
- Working with Strings
- Working with numbers
- Manipulating numeric data
- Working with Conditions
- Working with a List of Items
- Processing a list of items
- Working with objects and classes
- Defining fields and methods
- Declaring, Instantiating, and Initializing Objects

- Working with Object References
- Doing more with Arrays
- Introducing the NetBeans IDE
- Introducing the Soccer League Use Case
- Using the String Class
- Using the Java API Docs
- Using the StringBuilder Class
- More about primitive data types
- The remaining numeric operators
- Promoting and casting variables
- Using methods
- Method arguments and return values
- Static methods and variables
- How Arguments are Passed to a Method
- Overloading a method
- Access Control
- Encapsulation
- Overloading constructors
- Relational and conditional operators
- More ways to use if/else constructs
- Using Switch Statements
- Using the NetBeans Debugger
- Working with Dates
- Parsing the args Array
- Two-dimensional Arrays
- Alternate Looping Constructs
- Nesting Loops
- The ArrayList class
- Overview of inheritance
- Working with subclasses and superclasses
- Overriding methods in the superclass
- Introducing polymorphism
- Creating and extending abstract classes
- Polymorphism in the JDK foundation classes
- Using Interfaces
- Using the List Interface
- Introducing Lambda expressions
- Handling Exceptions: An overview
- Propagation of exceptions
- Catching and throwing exceptions
- Handling multiple exceptions and errors