Java SE7 Fundamentals

Duration: 5 Days

What you will learn

This Java SE7 Fundamentals course is an introduction to object-oriented programming using the Java language. Through hands-on exercises, you'll begin to build a baseline of knowledge to propel your career in development.

Learn To:
Use various Java programming language constructs to create several Java technology applications.
Use decision and looping constructs and methods to dictate program flow.
Perform basic error handling for your Java technology programs.
Implement intermediate Java programming and object-oriented (OO) concepts in Java technology programs.
Demonstrate knowledge of Java technology and the Java programming language.
Understand basic object oriented concepts such as inheritance, encapsulation, and abstraction.
Use and manipulate object references, and to write simple error handling code.

Benefits to You
This course teaches the significance of object-oriented programming, the keywords and constructs of the Java programming language. You'll walk away from this course with an understanding of the steps required to create simple Java technology programs.

Build a Solid Foundation in Java
You'll build a solid basis in the Java programming language upon which to base continued work and training. This course features the Java Platform, Standard Edition 7 (Java SE 7), and uses the Java SE Development Kit 7 (JDK 7) product.

Create Simple Java Classes

Learn how to create and use simple Java classes containing arrays, loops and conditional constructs. This course provides a solid understanding of what the Java SE7 platform is and how it is used in real world applications.

Application Developers
Developer
Portal Developer
Project Manager
System Administrator
Technical Administrator
Technical Consultant
Web Administrator

Course Objectives

Develop classes and describe how to declare a class

Analyze a business problem in order to recognize objects and operations that form the building blocks of the Java program design

Define the term

Demonstrate Java programming syntax

Write a simple Java program that compiles and runs successfully

Declare and initialize variables

List several primitive data types

Instantiate an object and effectively use object reference variables

Use operators, loops, and decision constructs

Declare and instantiate Arrays and ArrayLists and be able to iterate through them

Describe the benefits of using an Integrated Development Environment (IDE)

List and describe several key features of the Java technology

Declare a method with arguments and return values

Use inheritance to declare and define a subclass of an existing superclass

Describe how errors are handled in a Java program

Describe examples of how Java is used in applications, as well as consumer products

Course Topics

Introducing the Java Technology
Relating Java with other languages
Showing how to download, install, and configure the Java environment on a Windows system.
Describing the various Java technologies such as Java EE, JavaME, Embedded Java SE
Describing key features of the technology and the advantages of using Java
Using an Integrated Development Environment (IDE)

Thinking in Objects
Defining the problem domain
Identifying objects and recognizing the criteria for defining objects

Introducing the Java Language
Defining classes
Identifying the components of a class
Creating and using a test class

Compiling and executing a test program

Working with Primitive Variables
Declaring and initializing field variables
Describing primitive data types such as integral, floating point, textual, and logical
Declaring variables and assigning values
Using constants
Using arithmetic operators to modify values

Working with Objects
Declaring and initializing objects
Storing objects in memory
Using object references to manipulate data
Using JSE javadocs to look up the methods of a class
Working with String and StringBuilder objects

Using operators and decision constructs
Using relational and conditional operators
Testing equality between strings
Evaluating different conditions in a program and determining the algorithm
Creating if and if/else constructs
Nesting and chaining conditional statements
Using a switch statement

Creating and Using Arrays
Declaring, instantiating, and initializing a one-dimensional Array
Declaring, instantiating, and initializing a two-dimensional Array
Using a for loop to process an Array
Creating and initializing an ArrayList
Using the import statement to work with existing Java APIs
Accessing a value in an Array or and ArrayList
Using the args Array

Using Loop Constructs
Creating while loops and nested while loops
Developing a for loop
Using ArrayLists with for loops
Developing a do while loop
Understanding variable scope

Working with Methods and Method Overloading
Creating and Invoking a Method
Passing arguments and returning values
Creating static methods and variables
Using modifiers
Overloading a method

Using Encapsulation and Constructors
Creating constructors
Implementing encapsulation

Introducing Advanced Object Oriented Concepts

Using inheritance
Using types of polymorphism such as overloading, overriding, and dynamic binding
Working with superclasses and subclasses
Adding abstraction to your analysis and design
Understanding the purpose of Java interfaces
Creating and implementing a Java interface

Handling Errors
Understanding the different kinds of errors that can occur and how they are handled in Java
Understanding the different kinds of Exceptions in Java
Using Javadocs to research the Exceptions thrown by the methods of foundation classes
Writing code to handle Exceptions

The Big Picture
Creating packages and JAR files for deployment using java
Two and three tier architectures
Looking at some Java applications examples