

Java 8 Programming Essentials Using Eclipse (Custom for Boeing)

Course Summary

Description

This course will introduce the student to the Java programming language using the Java EE Indigo Eclipse. Through hands-on exercises, the student will become familiar with the Java syntax and the object oriented approach that this language utilizes. It is based on the new Java 1.8.

Objectives

By the end of this course, participants will be able to:

- Use the desired Java EE Eclipse proficiently
- Understand the Java language syntax
- Create, compile, run and debug Java applications
- Manage the Java files
- Translate a class diagram into Java code (and vice-versa)
- Use the Collections framework
- Familiar with Using the Java API

Topics

- Basic Java syntax (data types, operators, control statement)
- Object-oriented programming in Java
- UML notation - Class Diagram
- Using Collections Framework
- Using Exceptions
- Navigating Sun's API documentation
- Generics and Collections Framework
- Varargs
- Enumeration
- Default Methods

Audience

This course is designed for new Java programmers who have not previously programmed in an object-oriented language. The course can also run in Rad, Java EE Indigo Eclipse, WebLogic Workshop and JDeveloper.

Prerequisites

The student must be familiar object oriented programming and with basic programming concepts with prior programming experience in at least one language (COBOL, Visual Basic, C, etc.).

Duration

Five days

Java 8 Programming Essentials Using Eclipse (Custom for Boeing)

Course Outline

- I. Overview of the Course**
 - A. Java keyword: package
 - B. The Lab Disk
 - C. The Labs
 - D. File Hierarchy of Eclipse Based IDE
 - E. Mapping the Lab Disk with the IDE
 - F. Notations
- II. Overview of Eclipse**
 - A. What is the Eclipse Platform?
 - B. Eclipse Architecture
 - C. Getting Eclipse
 - D. Verify the Installed JRE
- III. First Java Application**
 - A. Class Declaration Syntax
 - B. Java Application
 - C. First Java Application
 - D. Using the Source Code
 - E. Comments
- IV. Variables**
 - A. Primitive Data Types
 - 1. Integers
 - 2. Floating Point
 - 3. char
 - 4. boolean
 - B. Variables
 - C. Declaring Variables
 - D. Legal Identifiers
 - E. Reserved Words
 - F. Declaring Variables
 - G. Assigning a Value to a Variable
 - H. Underscores in Numeric Literals
 - I. Terminologies
 - J. Primitive Data Types
 - K. Assignment and Conversion
 - L. Strings
- V. Operators**
 - A. Arithmetic Operators
 - B. Return Type
 - C. String Operators
 - D. Assignment Operator
 - E. Relational Operators
 - F. Increment/Decrement Operators
 - G. Logical Operators
 - H. Boolean Operators
 - I. Logical Not
- VI. Flow of Control**
 - A. Organizing Statements
 - B. Control Flow Statements
 - C. The if-then Statement
 - D. The if-then-else Statement
 - E. The if-then-if else Statement
 - F. Ternary Operator
 - G. The while Statement
 - H. The do-while Statement
 - I. The for Statement
 - J. The break keyword
 - K. The switch statement
 - L. The continue keyword
- VII. Classes vs Objects**
 - A. What is an Object?
 - B. What is a Class?
 - C. Classes
 - D. Revisiting: Class Declaration Syntax
 - 1. What are the Attributes of a Class?
 - 2. Attribute Definition Syntax
 - 3. Attributes of Part-time Class
 - E. Revisiting: Class Declaration Syntax
 - 1. Methods of a Class
 - 2. Method Definition Syntax
 - 3. Signature of the Method
 - 4. Overloading a Method
 - 5. Methods of Part-time Class
 - F. Classes vs Object
 - G. The this keyword
 - 1. In Variable
 - 2. In Method
 - H. The static keyword
 - 1. In Variable
 - 2. In Method
- VIII. Constructors**
 - A. Revisiting: Class Declaration Syntax
 - B. Constructors
 - C. Constructor Declaration Syntax
 - D. Constructors of Part-time Class
 - E. Invoking the Constructors
 - F. Revisiting: The this keyword
 - G. In Constructors
- IX. Arrays**
 - A. Arrays Defined
 - B. Array Variable Syntax
 - 1. Primitive Data Types
 - 2. Object Data Types
 - 3. Shortcut
 - C. Referencing the Array
 - D. Exceptions in Arrays
 - E. Arrays of Array

Java 8 Programming Essentials Using Eclipse (Custom for Boeing)

Course Outline

- X. Inheritance**
 - A. Inheritance
 - B. Revisiting: Class Declaration Syntax
 - C. The extends Keyword
 - D. The super keyword
 - 1. In Constructors
 - 2. In a Variable
 - 3. In a Method
 - E. The final keyword
 - 1. In a Variable
 - 2. In a Method
 - 3. In a Class
- XI. Polymorphism**
 - A. Polymorphism
 - B. Revisiting: Reference Variables
- XII. Encapsulation**
 - A. Encapsulation
 - B. Applying Encapsulation
 - C. Access Modifiers
 - D. The package keyword
 - 1. Using the package keyword
 - E. The import keyword
- XIII. Abstraction**
 - A. Abstraction
 - B. The abstract Keyword
 - 1. In Method
 - 2. In Class
 - C. Interface
 - 1. Interface Declaration Syntax
 - 2. Revisiting: Class Declaration Syntax
 - 3. Using an Interface
 - D. Improvement in Java 8
 - 1. Default methods
 - 2. Multiple Defaults
- XIV. Exception Handling**
 - A. What is an Exception?
 - B. How it Works
 - C. Exception Hierarchy
 - D. Error and Exception
 - E. Exception Class
 - F. Types of Exception
 - G. How to Handle Exception
 - H. Constructing an Exception Handler
 - 1. The try block
 - 2. The catch block
 - 3. The finally block
 - I. Throwing an Exception
 - J. Create and Use Your Own Exception
 - K. Chained Exception
 - L. Improvements in Java 7
- XV. Enumeration**
 - A. Enumeration
 - B. Enumeration Declaration Syntax
 - C. Using an Enumeration
 - D. Advantages of Enumeration
 - E. Predefined Methods
- XVI. Navigating the Java API (under development)**
 - A. Java API
 - 1. Instance Method
 - 2. Class Method
 - B. The String Class
 - C. The Math Class
 - D. The StringBuilder Class
 - E. The Wrapper Classes
 - F. Boxing and Autoboxing
- XVII. Generics and Collection Framework**
 - A. Generics
 - B. Why Generics
 - C. Class Declaration Syntax: Using Generics
 - D. Using Parameterized Type Class
 - E. Collections
 - F. What is the Collection Framework?
 - G. The import keyword
 - H. To Traverse a Collection
 - I. Revisiting: For Each Loop
 - J. Collection Interfaces
 - K. Collection Classes
 - 1. Priority Queue
 - 2. Vector
 - 3. ArrayList
 - 4. LinkedList
 - 5. Iterator
 - 6. HashSet
 - 7. TreeSet
 - 8. TreeMap
 - L. Improved Generic Instance Creation
- XVIII. Appendix A: Navigating Eclipse.**