

Core Java 8

Lesson 00: Java SE 8



Course Goals and Non Goals

➤ Course Goals

- Implementing OOPs features in Java
- Developing Java Desktop Applications
- Use of Core JDK 1.8 API
- Testing using Junit 4
- Implementing Multithreading

➤ Course Non Goals

- Developing GUI applications





Pre-requisites

Basic Programming Concepts

OOPs

XML

Intended Audience



Developers new to Java technology





Day Wise Schedule

Day 1

- Lesson 1: Introduction to Java
- Lesson 2: Eclipse 4.4 (Luna) as an IDE
- Lesson 3: Language Fundamentals
- Lesson 4: Classes and Objects

Day 2

- Lesson 5: Exploring Basic Java Class Libraries
- Lesson 6: Inheritance and Polymorphism

Day 3

- Lesson 7: Abstract Classes and Interfaces
- Lesson 8: Regular Expressions

Day 4

- Lesson 9 : Exception Handling
- Lesson 10: Array

Day 5

- Lesson 11: Collection
- Lesson 12: Generics



Day Wise Schedule

Day 6

- Lesson 13: File IO
- Lesson 14: Introduction to Junit 4

Day 7

- Lesson 15: Property Files
- Lesson 16: Java Database Connectivity (JDBC 4.0)

Day 8

- Lesson 17: Introduction to Layered Architecture
- Lesson 18: Advanced Testing

Day 9

- Lesson 19: Logging with Log4J
- Lesson 20: Multithreading

Day 10

- Lesson 21: Lambda Expressions
- Lesson 22: Stream API



Table of Contents

Lesson 1: Introduction to Java

- 1.1: Introduction to Java
- 1.2: Features of Java
- 1.3: Simple Program in Java
- 1.4: Developing software in Java

Lesson 2: Eclipse 4.4 (Luna) as an IDE

- 2.1: Installation and Setting up Eclipse
- 2.2: Introduction to Eclipse IDE
- 2.3: Creating and Managing Java Projects
- 2.4: Miscellaneous Options



Table of Contents

Lesson 3: Language Fundamentals

- 3.1: Keywords
- 3.2: Primitive Data Types
- 3.3: Operators and Assignments
- 3.4: Variables and Literals
- 3.5: Flow Control: Java's Control Statements
- 3.6: Best Practices

Lesson 4: Classes and Objects

- 4.1: Classes and Objects
- 4.2: Packages
- 4.3: Access Specifiers
- 4.4: Constructors - Default and Parameterized
- 4.5: this reference
- 4.6: Memory management in java
- 4.7: using static keyword
- 4.8: Enum
- 4.9: Best Practices



Table of Contents

Lesson 5: Exploring Basic Java Class Libraries

- 5.1: The Object Class
- 5.2: Wrapper Classes
- 5.3: Type casting
- 5.4: Using Scanner Class
- 5.5: System Class
- 5.6: String Handling
- 5.7: Date and Time API
- 5.8: Best Practices

Lesson 6: Inheritance and Polymorphism

- 6.1: Inheritance
- 6.2: Using super keyword
- 6.3: InstanceOf Operator
- 6.4: Method & Constructor overloading
- 6.5: Method overriding
- 6.6: @override annotation
- 6.7: Using final keyword



Table of Contents

Lesson 7: Abstract Classes and Interfaces

- 7.1: Abstract class
- 7.2: Interfaces
- 7.3: default methods
- 7.4: static methods on Interface
- 7.5 : Interface rules
- 7.6: Abstract class Vs Interface
- 7.7: Runtime Polymorphism

Lesson 8: Regular Expressions

- 8.1: Regular Expressions
- 8.2: Validating data
- 8.3: Best Practices



Table of Contents

Lesson 9: Exception Handling

- 9.1: Introduction
- 9.2: Exception Types and Exception Hierarchy
- 9.3: Try-catch-finally
- 9.4: Try-with-resources
- 9.5: Multi catch blocks
- 9.6: Throwing exceptions using throw
- 9.7: Declaring exceptions using throws
- 9.8: User defined Exceptions
- 9.9: Best Practices



Table of Contents

Lesson 10: Array

- 10.1: One/two dimensional array
- 10.2: Enhance For Loop
- 10.3: Method with Variable Argument Lists
- 10.4: Arrays class

Lesson 11: Collection

- 11.1: Collections Framework
- 11.2: Collection Interfaces
- 11.3: Iterating Collections
- 11.4: Implementing Classes
- 11.5: Comparable and Comparator
- 11.6: Map implementation
- 11.7: Legacy classes
- 11.8: Common Best Practices on Collections

Lesson 12: Generics

- 12.1: Generics
- 12.2: Writing Generic Classes
- 12.3: Using Generics with Collections



Table of Contents

Lesson 13: File IO

- 13.1: Overview of I/O Streams
- 13.2: Types of Streams
- 13.3: The Byte-stream I/O hierarchy
- 13.4: Character Stream Hierarchy
- 13.5: Buffered Stream
- 13.6: The File class
- 13.7: The Path class
- 13.8: Object Stream
- 13.9: Best Practices

Lesson 14 : Introduction to JUnit 4

- 14.1: Introduction
- 14.2: JUnit
- 14.3: Installing and Running JUnit
- 14.4: Testing with JUnit
- 14.5: Testing Exceptions
- 14.6: Test Fixtures
- 14.7: Best Practices



Table of Contents

Lesson 15: Property Files

- 15.1: What are Property Files?
- 15.2: Types of Property files
- 15.3: User defined Properties

Lesson 16: Java Database Connectivity (JDBC 4.0)

- 16.1: Java Database Connectivity - Introduction
- 16.2: Database Connectivity Architecture
- 16.3: JDBC APIs
- 16.4: Database Access Steps
- 16.5: Calling database procedures
- 16.6: Using Transaction
- 16.7: Best Practices



Table of Contents

Lesson 17: Introduction to Layered Architecture

- 17.1: Introduction
- 17.2: Testing DAO Classes

Lesson 18: Advanced Testing Concepts

- 18.1: Advanced Testing concepts
- 18.2: Test Suites
- 18.3: Parameterized Tests
- 18.4: Mocking Concepts



Table of Contents

Lesson 19: Logging with Log4J

- 19.1 Log4J Introduction
- 19.2 Log4J Concepts
- 19.3 Installation of Log4J
- 19.4 Configuring Log4J
- 19.5: Log4J Pros and Cons

Lesson 20: Multithreading

- 20.1 Understanding Threads
- 20.2 Thread life cycle
- 20.3 Scheduling threads- Priorities
- 20.4 Controlling threads using sleep(),join()



Table of Contents

Lesson 21: Lambda Expressions

- 21.1: Introduction to Functional Interface
- 21.2: Writing Lambda Expressions
- 21.3: Built in Functional Interfaces
- 21.4: Built in Functional Interfaces and Lambda Expressions
- 21.5: Method reference

Lesson 22: Stream API

- 22.1: Introduction to Stream API
- 22.2: Working with Stream API
- 22.3: Stream Operations



References

Books:

- Java, The Complete Reference; by Herbert Schildt
- Thinking in Java; by Bruce Eckel
- Beginning Java 8 Fundamentals by Kishori Sharan

Websites:

- Java home page: <http://java.sun.com/>
- JDK 1.8 documentation: <http://docs.oracle.com/javase/8/docs/>
- Multithreading :
<https://docs.oracle.com/javase/tutorial/essential/concurrency/index.html>



Next Step Courses



Servlets
JSP





Other Parallel Technology Areas

C ++

C#.Net

Visual Basic.Net