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: Java Language Fundamentals

Day 2

- Lesson 4: Declaration And Access Control
- Lesson 4: Classes and Objects

Day 3

- Lesson 5: Exploring Basic Java Class Libraries
- Lesson 6: Array
- Lesson 7: Assignments
- Lesson 7: Regular Expressions

Day 4

- Lesson 8: Inheritance and Polymorphism

Day 5

- Lesson 9: Abstract Classes and Interfaces

Day 6 [Online] 4 Hrs

- Lesson 10: Exception Handling

Day Wise Schedule

Day 7

- MCQ Assessment 1 (20 questions)
- Lesson 11 CoreJava8 - Collection and Generics[Online]

Day 8

- Lesson 11 :Collection Recap[With Comparable, Comparator, equals ,Hash code]
- I/O and NIO [Self Reading]

Day 9

- Lesson 12: Multithreading
- Lesson 13: Concurrent Patterns In Java

Day 10

- Lesson 14: Concurrent Collections In Java.[Self Reading]
- Lesson 14: Stream API
- Lesson 14: Lambda
- Maven [Self Reading]

Day 11

- Lesson 15: TDD with Junit 5 [Online]

Day 12

- Oracle SQL[Online]
- Jdbc [Self Learning]

Day Wise Schedule

Day 13

- Jdbc   [Self Learning]
- Core Java 8 practical Test on doselect
- MCQ Assessment 2 (20 questions)

Table of Contents

Lesson 1: Introduction to Java

- Introduction to Java
- Features of Java
- Simple Program in Java
- Developing software in Java

Lesson 2: Eclipse 4.4 (Luna) as an IDE

- Installation and Setting up Eclipse
- Introduction to Eclipse IDE
- Creating and Managing Java Projects
- Miscellaneous Options

Table of Contents

Lesson 3: Language Fundamentals

- Keywords
- Primitive Data Types
- Operators and Assignments
- Variables and Literals
- Flow Control: Java’s Control Statements
- Best Practices

Lesson 4: Classes and Objects

- Classes and Objects
- Packages
- Access Specifiers
- Constructors - Default and Parameterized
- this reference
- Memory management in java
- using static keyword
- Enum
- Best Practices

Table of Contents

Lesson 5: Exploring Basic Java Class Libraries

- The Object Class
- Wrapper Classes
- Type casting
- Using Scanner Class
- System Class
- String Handling
- Date and Time API
- Best Practices

Lesson 6: Arrays

- Understand the different types of Arrays
- Implement one and multi dimensional arrays
- Iterate arrays using loops
- Use varargs
- Work with java.util.Arrays

Table of Contents

Lesson 7: Regular Expressions

- Regular Expressions
- Validating data
- Best Practices

Lesson 7: Assignments

- Stack and Heap—Quick Review
- Literals, Assignments, and Variables
- Assignment Operators
- Casting Primitives
- Using a Variable or Array Element That Is Uninitialized and Unassigned
- Local (Stack, Automatic) Primitives and Objects
- Passing Variables into Methods
- Passing Object Reference Variables

Table of Contents

Lesson 7: Assignments

- Does Java Use Pass-By-Value Semantics?
- Passing Primitive Variables
- Array Declaration, Construction, and Initialization
- Initializing Blocks
- Using Wrapper Classes and Boxing
- An Overview of the Wrapper Classes
- Creating Wrapper Objects
- Using Wrapper Conversion Utilities
- Auto Boxing
- Overloading
- Garbage Collection and memory Management
- Writing Code That Explicitly Makes Objects Eligible for Garbage Collection

Table of Contents

Lesson 8: Inheritance and Polymorphism

- Inheritance
- Using super keyword
- Instance Of Operator
- Method & Constructor overloading
- Method overriding
- @override annotation
- Using final keyword

Lesson 9: Abstract Classes and Interfaces

- Abstract class
- Interfaces
- default methods
- static methods on Interface
- Interface rules
- Abstract class Vs Interface
- Runtime Polymorphism

Table of Contents

Table of Contents

Lesson 12: Multithreading

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

Lesson 13: Concurrent Patterns in Java

- Introducing Executors, What Is Wrong with the Runnable Pattern?
- Defining the Executor Pattern: A New Pattern to Launch Threads
- Defining the Executor Service Pattern, a First Simple Example
- Comparing the Runnable and the Executor Service Patterns
- Understanding the Waiting Queue of the Executor Service
- Wrapping-up the Executor Service Pattern
- From Runnable to Callable: What Is Wrong with Runnable?
- Defining a New Model for Tasks That Return Objects
- Introducing the Callable Interface to Model Task
- Introducing the Future Object to Transmit Objects Between Thread
- Wrapping-up Callable and Futures, Handling Exceptions

Table of Contents

Table of Contents

Table of Contents

Lesson 15 : Introduction to Junit 4

- Types of Tests
- Why Unit Tests Are Important
- What's JUnit?
- JUnit 5 Architecture
- IDEs and Build Tool Support
- Setting up JUnit with Maven
- Lifecycle Methods
- Test Hierarchies
- Assertions
- Disabling Tests
- AssumptionsTest Interfaces and Default Methods
- Repeating Tests
- Dynamic Tests
- Parameterized Tests
- Argument Sources
- TDD Introduction
- Types of Testing
- Testing Frameworks and Tools
- Testing Concepts,Mockito

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