Java Advanced (Programming SE 8 – OCP)

Prerequisites / Further Training

- Java Beginner
- Beginning SQL

Also have a look at our Java Bootcamp

Alignment

Oracle OCP Certification aligned to Oracle OCP Java Exam

After this course you should be able to

- Have a good understanding of programming and the building blocks of an OO programming language, with an emphasis on JAVA.
- Build small apps in Java, making use of I/O, Networking, GUI
- Prepare for Oracle OCA and OCP exams

Course Material

Course Material Provided

Course Contents

DAY 1

Abstract and Nested Classes

- Modeling Business Problems with Classes
- Enabling Generalization
- Identifying the Need for Abstract Classes
- Defining Abstract Classes
- Defining Abstract Methods
- Validating Abstract Classes

- Final Methods
- Final Classes
- Final Variables
- Declaring Final Variables
- Nested Classes
- Example: Member Class
- Enumerations
- Enum Usage
- Complex Enums

Interfaces and Lambda Expressions

- Java Interfaces
- A Problem Solved by Interfaces
- CrushedRock Class
- The SalesCalcs Interface
- Adding an Interface
- Interface References
- Interface Reference Usefulness
- Interface Code Flexibility
- default Methods in Interfaces
- default Method: Example
- static Methods in Interfaces
- Constant Fields
- Extending Interfaces
- Implementing and Extending
- Anonymous Inner Classes
- Anonymous Inner Class: Example
- String Analysis Regular Class
- String Analysis Regular Test Class
- String Analysis Interface: Example
- String Analyzer Interface Test Class
- Encapsulate the for Loop
- String Analysis Test Class with Helper Method
- String Analysis Anonymous Inner Class
- String Analysis Lambda Expression
- Lambda Expression Defined

- What Is a Lambda Expression?
- Lambda Expression Shorthand
- Lambda Expressions as Variables

7 Generics and Collections

- Generics 7-4
- Simple Cache Class Without Generics 7-5
- Generic Cache Class 7-6
- Generics in Action 7-7
- Generics with Type Inference Diamond 7-8
- Collections 7-9
- Collection Types 7-10
- Collection Interfaces and Implementation 7-11
- List Interface 7-12
- ArrayList 7-13
- Autoboxing and Unboxing 7-14
- ArrayList Without Generics 7-15
- Generic ArrayList 7-16
- Generic ArrayList: Iteration and Boxing 7-17
- Set Interface 7-18
- TreeSet: Implementation of Set 7-19
- Map Interface 7-20
- Map Types 7-21
- TreeMap: Implementation of Map 7-22
- Deque Interface 7-23
- Stack with Deque: Example 7-24
- Ordering Collections 7-25

DAY 2

Collections, Streams, and Filters

- Collections, Streams, and Filters 8-3
- The Person Class 8-4
- Person Properties 8-5
- Builder Pattern 8-6
- Collection Iteration and Lambdas 8-7

- RoboCallTest07: Stream and Filter 8-8
- RobocallTest08: Stream and Filter Again 8-9
- SalesTxn Class 8-10
- Java Streams 8-11
- The Filter Method 8-12
- Method References 8-13
- Method Chaining 8-14
- Pipeline Defined 8-16

Lambda Built-in Functional Interfaces

- Built-in Functional Interfaces 9-3
- The java.util.function Package 9-4
- Example Assumptions 9-5
- Predicate 9-6
- Predicate: Example 9-7
- Consumer 9-8
- Consumer: Example 9-9
- Function 9-10
- Function: Example 9-11
- Supplier 9-12
- Supplier: Example 9-13
- Primitive Interface 9-14
- Return a Primitive Type 9-15
- Return a Primitive Type: Example 9-16
- Process a Primitive Type 9-17
- Process Primitive Type: Example 9-18
- Binary Types 9-19
- Binary Type: Example 9-20
- Unary Operator 9-21
- UnaryOperator: Example 9-22
- Wildcard Generics Review 9-23

Lambda Operations

- Objectives 10-2
- Streams API 10-3
- Types of Operations 10-4

- Extracting Data with Map 10-5
- Taking a Peek 10-6
- Search Methods: Overview 10-7
- Search Methods 10-8
- Optional Class 10-9
- Lazy Operations 10-10
- Stream Data Methods 10-11
- Performing Calculations 10-12
- Sorting 10-13
- Comparator Updates 10-14
- Saving Data from a Stream 10-15
- Collectors Class 10-16
- Quick Streams with Stream.of 10-17
- Flatten Data with flatMap 10-18

DAY 3

Exceptions and Assertions

- Error Handling 11-3
- Exception Handling in Java 11-4
- try-catch Statement 11-5
- Exception Objects 11-6
- Exception Categories 11-7
- Handling Exceptions 11-8
- finally Clause 11-9
- try-with-resources Statement 11-10
- Catching Multiple Exceptions 11-11
- Declaring Exceptions 11-12
- Handling Declared Exceptions 11-13
- Throwing Exceptions 11-14
- Custom Exceptions 11-15
- Assertions 11-16
- Assertion Syntax 11-17
- Internal Invariants 11-18
- Control Flow Invariants 11-19
- Class Invariants 11-20

- Controlling Runtime Evaluation of Assertions 11-21

Java Date/Time API

- Why Is Date and Time Important? 12-3
- Previous Java Date and Time 12-4
- Java Date and Time API: Goals 12-5
- Working with Local Date and Time 12-6
- Working with LocalDate 12-7
- LocalDate: Example 12-8
- Working with LocalTime 12-9
- LocalTime: Example 12-10
- Working with LocalDateTime 12-11
- LocalTimeDate: Example 12-12
- Working with Time Zones 12-13
- Daylight Savings Time Rules 12-14
- Modeling Time Zones 12-15
- Creating ZonedDateTime Objects 12-16
- Working with ZonedDateTime Gaps/Overlaps 12-17
- ZoneRules 12-18
- Working Across Time Zones 12-19
- Date and Time Methods 12-20
- Date and Time Amounts 12-21
- Period 12-22
- Duration 12-23
- Calculating Between Days 12-24
- Making Dates Pretty 12-25
- Using Fluent Notation 12-26

Java I/O Fundamentals

- Objectives 13-2
- Java I/O Basics 13-3
- I/O Streams 13-4
- I/O Application 13-5
- Data Within Streams 13-6
- Byte Stream InputStream Methods 13-7
- Byte Stream OutputStream Methods 13-8

- Byte Stream: Example 13-9
- Character Stream Reader Methods 13-10
- Character Stream Writer Methods 13-11
- Character Stream: Example 13-12
- I/O Stream Chaining 13-13
- Chained Streams: Example 13-14
- Console I/O 13-15
- Writing to Standard Output 13-16
- Reading from Standard Input 13-17
- Channel I/O 13-18
- Persistence 13-19
- Serialization and Object Graphs 13-20
- Transient Fields and Objects 13-21
- Transient: Example 13-22
- Serial Version UID 13-23
- Serialization: Example 13-24
- Writing and Reading an Object Stream 13-25
- Serialization Methods 13-26

DAY 4

Java File I/O (NIO.2)

- Objectives 14-2
- New File I/O API (NIO.2) 14-3
- Limitations of java.io.File 14-4
- File Systems, Paths, Files 14-5
- Relative Path Versus Absolute Path 14-6
- Java NIO.2 Concepts 14-7
- Path Interface 14-8
- Path Interface Features 14-9
- Path: Example 14-10
- Removing Redundancies from a Path 14-11
- Creating a Subpath 14-12
- Joining Two Paths 14-13
- Symbolic Links 14-14
- Working with Links 14-15

- File Operations 14-16
- Checking a File or Directory 14-17
- Creating Files and Directories 14-19
- Deleting a File or Directory 14-20
- Copying a File or Directory 14-21
- Moving a File or Directory 14-22
- List the Contents of a Directory 14-23
- Walk the Directory Structure 14-24
- BufferedReader File Stream 14-25
- NIO File Stream 14-26
- Read File into ArrayList 14-27
- Managing Metadata 14-28
- Symbolic Links 14-29

Concurrency

- Objectives 15-2
- Task Scheduling 15-3
- Legacy Thread and Runnable 15-4
- Extending Thread 15-5
- Implementing Runnable 15-6
- The java.util.concurrent Package 15-7
- Recommended Threading Classes 15-8
- java.util.concurrent.ExecutorService 15-9
- Example ExecutorService 15-10
- Shutting Down an ExecutorService 15-11
- java.util.concurrent.Callable 15-12
- Example Callable Task 15-13
- java.util.concurrent.Future 15-14
- Example 15-15
- Threading Concerns 15-16
- Shared Data 15-17
- Problems with Shared Data 15-18
- Nonshared Data 15-19
- Atomic Operations 15-20
- Out-of-Order Execution 15-21
- The synchronized Keyword 15-22

- synchronized Methods 15-23
- synchronized Blocks 15-24
- Object Monitor Locking 15-25
- Threading Performance 15-26
- Performance Issue: Examples 15-27
- java.util.concurrent Classes and Packages 15-28
- The java.util.concurrent.atomic Package 15-29
- java.util.concurrent.CyclicBarrier 15-30
- Thread-Safe Collections 15-32

The Fork-Join Framework

- Objectives 16-2
- Parallelism 16-3
- Without Parallelism 16-4
- Naive Parallelism 16-5
- The Need for the Fork-Join Framework 16-6
- Work-Stealing 16-7
- A Single-Threaded Example 16-8
- java.util.concurrent.ForkJoinTask<V> 16-9
- xiv
- RecursiveTask Example 16-10
- compute Structure 16-11
- compute Example (Below Threshold) 16-12
- compute Example (Above Threshold) 16-13
- ForkJoinPool Example 16-14
- Fork-Join Framework Recommendations 16-15
- Summary 16-16
- Practice 16-1 Overview: Using the Fork-Join Framework 16-17
- Quiz 16-18
- 17 Parallel Streams
- Objectives 17-2
- Streams Review 17-3
- Old Style Collection Processing 17-4
- New Style Collection Processing 17-5
- Stream Pipeline: Another Look 17-6

- Styles Compared 17-7
- Parallel Stream 17-8
- Using Parallel Streams: Collection 17-9
- Using Parallel Streams: From a Stream 17-10
- Pipelines Fine Print 17-11
- Embrace Statelessness 17-12
- Avoid Statefulness 17-13
- Streams Are Deterministic for Most Part 17-14
- Some Are Not Deterministic 17-15
- Reduction 17-16
- Reduction Fine Print 17-17
- Reduction: Example 17-18
- A Look Under the Hood 17-24
- Illustrating Parallel Execution 17-25
- Performance 17-36
- A Simple Performance Model 17-37

DAY 5

Building Database Applications with JDBC

- Objectives 18-2
- Using the JDBC API 18-3
- Using a Vendor’s Driver Class 18-4
- Key JDBC API Components 18-5
- Writing Queries and Getting Results 18-6
- Using a ResultSet Object 18-7
- CRUD Operations Using JDBC API: Retrieve 18-8
- CRUD Operations Using JDBC: Retrieve 18-9
- CRUD Operations Using JDBC API: Create 18-10
- CRUD Operations Using JDBC API: Update 18-11
- CRUD Operations Using JDBC API: Delete 18-12
- SQLException Class 18-13
- Closing JDBC Objects 18-14
- try-with-resources Construct 18-15
- Using PreparedStatement 18-16
- Using PreparedStatement: Setting Parameters 18-17

- Executing PreparedStatement 18-18
- PreparedStatement:Using a Loop to Set Values 18-19
- Using CallableStatement 18-20

Localization

- Objectives 19-2
- Why Localize? 19-3
- A Sample Application 19-4
- Locale 19-5
- Properties 19-6
- Loading and Using a Properties File 19-7
- Loading Properties from the Command Line 19-8
- Resource Bundle 19-9
- Resource Bundle File 19-10
- Sample Resource Bundle Files 19-11
- Initializing the Sample Application 19-12
- Sample Application: Main Loop 19-13
- The printMenu Method 19-14
- Changing the Locale 19-15
- Sample Interface with French 19-16
- Format Date and Currency 19-17
- Displaying Currency 19-18
- Formatting Currency with NumberFormat 19-19
- Displaying Dates 19-20
- Displaying Dates with DateTimeFormatter 19-21
- Format Styles 19-22

Duration and pricing

Price Group A

Certificate

Read About Our Certificates

Bookings

You can download the course registration form on our home page or by clicking here

Brochure

You may download a pdf copy of this page by clicking on the pdf icon at the top of the page.

Questions

Please email us

Schedule

On the calendar below. If your browser doesn’t display the calendar below, please click on this link or try using Google Chrome, alternatively please enquire via our Contact Us page.