# Java Base

## Introduzione a Java

- Introduzione al linguaggio ad oggetti JAVA, e preparazione ambiente di sviluppo con jdk e IDE.

## Primi Concetti di Programmazione Object Oriented

- Concetto di classe, oggetto, variabile, proprietà, metodo, costruttore e package.
- Reference this per l’utilizzo nei metodi con i parametri.
- Implementazione dei metodi accessor e mutator.

## Tipi di Dati e Casting

- Tipi di dati primitivi e reference.
- Concetti di cast e promotion.

## Iterazioni e strutture di controllo

- Implementazione di algoritmi in java utilizzando assegnazioni, cicli, condizioni.

## Enumerazioni, Array e String

- Altre tipologie di dati: enumerazioni, Array, ArrayList, String e StringBuilder

## Programmazione avanzata Object Oriented

- Analisi e Test dei tre principi fondamentali della Programmazione ad Oggetti:

    - incapsulamento,
    - ereditarietà
    - polimorfismo.
- Reference super.

## Modificatori Java

- Spiegazione dettagliata dei modificatori public, private, protected, default, final e static.

## Classi Astratte ed Interfacce

- Implementazione ed utilizzo di classi astratte ed interfacce.

## Modellazione UML OO

- Descrizione dei CLASS DIAGRAM utili nella progettazione del software.

## Gestione delle Eccezioni

- Eccezioni, errori ed asserzioni.
- Utilizzo del blocco try – catch – finally e metodo di propagazione delle eccezioni.
- I principali componenti della libreria Swing.

# Programma Certificazione Oracle OCA Oracle Certified Associate, Java SE

---

## Java Building Blocks

- Understanding the Java Class Structure
    - Fields and Methods

    - Comments
    - Classes vs. Files
- Writing a main() Method
- Understanding Package Declarations and Imports
    - Wildcards
    - Redundant Imports
    - Naming Conflicts
    - Creating a New Package
    - Code Formatting on the Exam
- Creating Objects
    - Constructors
    - Reading and Writing Object Fields
    - Instance Initializer Blocks
    - Order of Initialization
- Distinguishing Between Object References and Primitives * Primitive Types * Reference Types * Key Differences
- Declaring and Initializing Variables
    - Declaring Multiple Variables
- Identifiers
- Understanding Default Initialization of Variables
    - Local Variables
    - Instance and Class Variables
- Understanding Variable Scope
- Ordering Elements in a Class
- Destroying Objects
    - Garbage Collection
    - finalize()
- Benefits of Java

# Operators and Statements

- Understanding Java Operators
- Working with Binary Arithmetic Operators
    - Arithmetic Operators

    - Numeric Promotion
- Working with Unary Operators
    - Logical Complement and Negation Operators
    - Increment and Decrement Operators
- Using Additional Binary Operators
    - Assignment Operators
    - Compound Assignment Operators
    - Relational Operators
    - Logical Operators
    - Equality Operators
- Understanding Java Statements
    - The if-then Statement
    - The if-then-else Statement
    - The switch Statement
    - The while Statement
    - The do-while Statement
    - The for Statement
- Understanding Advanced Flow Control
    - Nested Loops
    - Adding Optional Labels
    - The break Statement
    - The continue Statement

# Core Java APIs

- Creating and Manipulating Strings
    - Concatenation
    - Immutability
    - The String Pool
    - Important String Methods
    - Method Chaining
- Using the StringBuilder Class
    - Mutability and Chaining
    - Creating a StringBuilder

    - Important StringBuilder Methods
    - StringBuilder vs. StringBuffer
- Understanding Equality
- Understanding Java Arrays
    - Creating an Array of Primitives
    - Creating an Array with Reference Variables
    - Using an Array
    - Sorting
    - Searching
    - Varargs
    - Multidimensional Arrays
- Understanding an ArrayList
    - Creating an ArrayList
    - Using an ArrayList
    - Wrapper Classes
    - Autoboxing
    - Converting Between array and List
    - Sorting
- Working with Dates and Times
    - Creating Dates and Times
    - Manipulating Dates and Times
    - Working with Periods
    - Formatting Dates and Times
    - Parsing Dates and Times

# Methods and Encapsulation

- Designing Methods
    - Optional Specifiers
    - Return Type
    - Method Name
    - Parameter List
    - Optional Exception List
    - Method Body

- Working with Varargs
- Applying Access Modifiers
    - Private Access
    - Default (Package Private) Access
    - Protected Access
    - Public Access
    - Designing Static Methods and Fields
    - Calling a Static Variable or Method
    - Static vs. Instance
    - Static Variables
    - Static Initialization
    - Static Imports
- Passing Data Among Methods
- Overloading Methods
- Creating Constructors
    - Default Constructor
    - Overloading Constructors
    - Final Fields
    - Order of Initialization
- Encapsulating Data
    - Creating Immutable Classes
- Writing Simple Lambdas
    - Lambda Example
    - Lambda Syntax
    - Predicates

# Class Design

- Introducing Class Inheritance
    - Extending a Class
    - Applying Class Access Modifiers
    - Creating Java Objects
    - Defining Constructors
    - Calling Inherited Class Members

    - Inheriting Methods
    - Inheriting Variables
- Creating Abstract Classes
    - Defining an Abstract Class
    - Creating a Concrete Class
    - Extending an Abstract Class
- Implementing Interfaces
    - Defining an Interface
    - Inheriting an Interface
    - Interface Variables
    - Default Interface Methods
    - Static Interface Methods
- Understanding Polymorphism
    - Object vs. Reference
    - Casting Objects
    - Virtual Methods
    - Polymorphic Parameters
    - Polymorphism and Method Overriding

# Exceptions

- Understanding Exceptions
    - The Role of Exceptions
    - Understanding Exception Types
    - Throwing an Exception
- Using a try Statement
    - Adding a finally Block
    - Catching Various Types of Exceptions
    - Throwing a Second Exception
- Recognizing Common Exception Types
    - Runtime Exceptions
    - Checked Exceptions
    - Errors
- Calling Methods That Throw Exceptions

    - Subclasses
    - Printing an Exception

# Oracle Certified Professional Java SE 8 Programmer

## Advanced Class Design

- Reviewing OCA Concepts
    - Access Modifiers
    - Overloading and Overriding
    - Abstract Classes
    - Static and Final
    - Imports
- Using instanceof
- Understanding Virtual Method Invocation
- Annotating Overridden Methods
- Coding equals, hashCode, and toString
    - toString
    - equals
    - hashCode
- Working with Enums
    - Using Enums in Switch Statements
    - Adding Constructors, Fields, and Methods
- Creating Nested Classes
    - Member Inner Classes
    - Local Inner Classes
    - Anonymous Inner Classes
    - Static Nested Classes

## Design Patterns and Principles

- Designing an Interface

    - Purpose of an Interface
- Introducing Functional Programming
    - Defining a Functional Interface
    - Implementing Functional Interfaces with Lambdas
    - Applying the Predicate Interface
- Implementing Polymorphism
    - Distinguishing between an Object and a Reference
    - Casting Object References
- Understanding Design Principles
    - Encapsulating Data
    - Creating JavaBeans
    - Applying the Is-a Relationship
    - Applying the Has-a Relationship
    - Composing Objects
- Working with Design Patterns
    - Applying the Singleton Pattern
    - Creating Immutable Objects
    - Using the Builder Pattern
    - Creating Objects with the Factory Pattern

# Generics and Collections

- Reviewing OCA Collections
    - Array and ArrayList
    - Searching and Sorting
    - Wrapper Classes and Autoboxing
    - The Diamond Operator
- Working with Generics
    - Generic Classes
    - Generic Interfaces
    - Generic Methods
    - Interacting with Legacy Code
    - Bounds
    - Putting It All Together

- Using Lists, Sets, Maps, and Queues
    - Common Collections Methods
    - Using the List Interface
    - Using the Set Interface
    - Using the Queue Interface
    - Map
    - Comparing Collection Types
- Comparator vs. Comparable
    - Comparable
    - Comparator
- Searching and Sorting
- Additions in Java 8
    - Using Method References
    - Removing Conditionally
    - Updating All Elements
    - Looping through a Collection
    - Using New Java 8 Map APIs

# Functional Programming

- Using Variables in Lambdas
- Working with Built-In Functional Interfaces
    - Implementing Supplier
    - Implementing Consumer and BiConsumer
    - Implementing Predicate and BiPredicate
    - Implementing Function and BiFunction
    - Implementing UnaryOperator and BinaryOperator
    - Checking Functional Interfaces
- Returning an Optional
- Using Streams
    - Creating Stream Sources
    - Using Common Terminal Operations
    - Using Common Intermediate Operations
    - Putting Together the Pipeline

    - Printing a Stream
- Working with Primitives
    - Creating Primitive Streams
    - Using Optional with Primitive Streams
    - Summarizing Statistics
    - Learning the Functional Interfaces for Primitives
- Working with Advanced Stream Pipeline Concepts
    - Linking Streams to the Underlying Data
    - Chaining Optionals
    - Collecting Results

## Dates, Strings, and Localization

- Working with Dates and Times
    - Creating Dates and Times
    - Manipulating Dates and Times
    - Working with Periods
    - Working with Durations
    - Accounting for Daylight Savings Time
- Reviewing the String class
- Adding Internationalization and Localization
    - Picking a Locale
    - Using a Resource Bundle
    - Formatting Numbers
    - Formatting Dates and Times

## Exceptions and Assertions

- Reviewing Exceptions
    - Exceptions Terminology
    - Categories of Exceptions
    - Exceptions on the OCP
    - Try Statement
    - Throw vs. Throws

- Creating Custom Exceptions
- Using Multi-catch
- Using Try-With-Resources
    - Try-With-Resources Basics
    - AutoCloseable
    - Suppressed Exceptions
    - Putting It Together
- Rethrowing Exceptions
- Working with Assertions
    - The assert Statement
    - Enabling Assertions
    - Using Assertions

# Concurrency

- Introducing Threads
    - Distinguishing Thread Types
    - Understanding Thread Concurrency
    - Introducing Runnable
    - Creating a Thread
    - Polling with Sleep
- Creating Threads with the ExecutorService
    - Introducing the Single-Thread Executor
    - Shutting Down a Thread Executor
    - Submitting Tasks
    - Waiting for Results
    - Scheduling Tasks
    - Increasing Concurrency with Pools
- Synchronizing Data Access
    - Protecting Data with Atomic Classes
    - Improving Access with Synchronized Blocks
    - Synchronizing Methods
    - Understanding the Cost of Synchronization
- Using Concurrent Collections

    - Introducing Concurrent Collections
    - Understanding Memory Consistency Errors
    - Working with Concurrent Classes
    - Obtaining Synchronized Collections
- Working with Parallel Streams
    - Creating Parallel Streams
    - Processing Tasks in Parallel
    - Processing Parallel Reductions
- Managing Concurrent Processes
    - Creating a CyclicBarrier
    - Applying the Fork/Join Framework
- Identifying Threading Problems
    - Understanding Liveness
    - Managing Race Conditions

# IO

- Understanding Files and Directories
    - Conceptualizing the File System
    - Introducing the File Class
- Introducing Streams
    - Stream Fundamentals
    - Stream Nomenclature
    - Common Stream Operations
- Working with Streams
    - The FileInputStream and FileOutputStream Classes
    - The FileReader and FileWriter classes
    - The ObjectInputStream and ObjectOutputStream Classes
    - The PrintStream and PrintWriter Classes
    - Review of Stream Classes
- Interacting with Users
    - The Old Way
    - The New Way

# NIO.2

- Introducing NIO.2
    - Introducing Path
    - Creating Paths
- Interacting with Paths and Files
    - Providing Optional Arguments
    - Using Path Objects
    - Interacting with Files
- Understanding File Attributes
    - Discovering Basic File Attributes
    - Improving Access with Views
- Presenting the New Stream Methods
    - Conceptualizing Directory Walking
    - Walking a Directory
    - Searching a Directory
    - Listing Directory Contents
    - Printing File Contents
- Comparing Legacy File and NIO.2 Methods

# JDBC

- Introducing Relational Databases and SQL
    - Identifying the Structure of a Relational Database
    - Writing Basic SQL Statements
- Introducing the Interfaces of JDBC
- Connecting to a Database
    - Building a JDBC URL
    - Getting a Database Connection
- Obtaining a Statement
    - Choosing a ResultSet Type
    - Choosing a ResultSet Concurrency Mode
- Executing a Statement
- Getting Data from a ResultSet

    - Reading a ResultSet
    - Getting Data for a Column
    - Scrolling ResultSet
- Closing Database Resources
- Dealing with Exceptions