IIAIT

COURSE CONTENT

Oracle Java

(Core Java)Training

Oracle Java SE 6 (Core Java) Training

1. Getting Started

- Examine Java technology
- Analyze a simple Java technology application
- Execute a Java technology application

2. Object-Oriented Programming

- Define modeling concepts: abstraction, encapsulation, and packages
- Discuss Java technology application code reuse
- Define class, member, attribute, method, constructor, and package
- Use the access modifiers private and public as appropriate for the guidelines of encapsulation
- Invoke a method on a particular object
- Use the Java technology API online documentation

3. Identifiers, Keywords, and Types

- Use comments in a source program
- Distinguish between valid and invalid identifiers
- Use the eight primitive types
- Define literal values for numeric and textual types
- Define the terms primitive variable and reference variable
- Declare variables of class type
- Construct an object using new and describe default initialization
- Describe the significance of a reference variable

4. Expressions and Flow Control

- Distinguish between instance and local variables
- Describe how to initialize instance variables
- Recognize, describe, and use Java software operators
- Distinguish between legal and illegal assignments of primitive types
- Identify boolean expressions and their requirements in control constructs
- Recognize assignment compatibility and required casts in fundamental types
- Use if, switch, for, while, and do constructions and the labeled forms of break and continue as flow control structures in a program

5. Arrays

- Declare and create arrays of primitive, class, or array types
- Explain why elements of an array are initialized
- Explain how to initialize the elements of an array
- Determine the number of elements in an array
- Create a multidimensional array
- Write code to copy array values from one array to another

6. Class Design

- Define inheritance, polymorphism, overloading, overriding, and virtual method invocation
- Use the access modifiers protected and the default (package-friendly)
- Describe the concepts of constructor and method overloading
- Describe the complete object construction and initialization operation

7. Advanced Class Features

- Create static variables, methods, and initializers
- Create final classes, methods, and variables
- Create and use enumerated types
- Use the static import statement
- Create abstract classes and methods
- Create and use an interface

8. Exceptions and Assertions

- Define exceptions
- Use try, catch, and finally statements
- Describe exception categories
- Identify common exceptions
- Develop programs to handle your own exceptions
- Use assertions
- Distinguish appropriate and inappropriate uses of assertions
- Enable assertions at runtime

9. Collections and Generics Framework

- Describe the general purpose implementations of the core interfaces in the Collections framework

- Examine the Map interface
- Examine the legacy collection classes
- Create natural and custom ordering by implementing the Comparable and Comparator interfaces
- Use generic collections and type parameters in generic classes
- Refactor existing non-generic code
- Write a program to iterate over a collection
- Examine the enhanced for loop

10. I/O Fundamentals

- Write a program that uses command-line arguments and system properties
- Examine the Properties class
- Construct node and processing streams, and use them appropriately
- Serialize and deserialize objects
- Distinguish readers and writers from streams, and select appropriately between them

11. Console I/ O and File I/O

- Read data from the console
- Write data to the console
- Describe files and file I/O

12. Building Java GUIs Using the Swing API

- Describe the JFC Swing technology
- Identify the Swing packages
- Describe the GUI building blocks: containers, components, and layout managers
- Examine top-level, general-purpose, and special-purpose properties of container
- Examine components
- Examine layout managers
- Describe the Swing single-threaded model
- Build a GUI using Swing components

13. Handling GUI-Generated Events

- Define events and event handling
- Examine the Java SE event model
- Describe GUI behavior
- Determine the user action that originated an event

- Develop event listeners
- Describe concurrency in Swing-based GUIs and describe the features of the SwingWorker class

14. GUI-Based Applications

- Describe how to construct a menu bar, menu, and menu items in a Java GUI
- Understand how to change the color and font of a component

15. Threads

- Define a thread
- Create separate threads in a Java technology program, controlling the code and data that are used by that thread
- Control the execution of a thread and write platform-independent code with threads
- Describe the difficulties that might arise when multiple threads share data
- Use wait and notify to communicate between threads
- Use synchronized to protect data from corruption

16. Networking

- Develop code to set up the network connection
- Understand TCP/IP
- Use ServerSocket and Socket classes to implement TCP/IP clients and servers

Oracle Java SE 7 (Core java) Training

1. Java Platform Overview

- Introductions
- Course Schedule
- Java Overview
- Java Platforms
- OpenJDK
- Licensing
- Java in Server Environments

- The Java Community Process

2. Java Syntax and Class Review

- Simple Java classes
- Java fields, constructors and methods
- Model objects using Java classes
- Package and import statements

3. Encapsulation and Polymorphism

- Encapsulation in Java class design
- Model business problems with Java classes
- Immutability
- Subclassing
- Overloading methods
- Variable argument methods

4. Java Class Design

- Access modifiers: private, protected and public
- Method overriding
- Constructor overloading
- The instanceof operator
- Virtual method invocation
- Polymorphism
- Casting object references
- Overriding Object methods

5. Advanced Class Design

- Abstract classes and type generalization
- The static and final modifiers
- Field modifier best practices
- The Singleton design pattern
- Designing abstract classes
- Nested classes
- Enumerated types

6. Inheritance with Java Interfaces

- Java Interfaces
- Types of Inheritance
- Object composition and method delegation
- Implementing multiple interfaces
- The DAO design pattern

7. Generics and Collections

- Generic classes and type parameters
- Type inference (diamond)
- Collections and generics
- List, set and Map
- Stack and Deque

8. String processing

- String manipulation with StringBuilder and StringBuffer
- Essential String methods
- Text parsing in Java
- Input processing with Scanner
- Text output and formatting
- Regular expressions with the Pattern and Matcher classes

9. Exceptions and Assertions

- Exceptions categories
- Standard Java Exception classes
- Creating your own Exception classes
- Using try-catch and the finally clause
- Using try-with-resources and the AutoCloseable interface
- The multi-catch feature
- Best practices using exceptions
- Assertions

10. I/O Fundamentals

- I/O using Java

- Reading the console input stream
- Writing to the console
- Using I/O Streams
- Chaining I/O Streams
- Channel I/O
- Reading and writing objects using Serialization

11. File I/O with NIO 2

- The Path interface
- The Files class
- Directory and File operations
- Managing file system attributes
- Reading, writing, and creating files
- Watching for file system changes

12. Threading

- Operating system task scheduling
- Recognizing multithreaded environments
- Creating multi-threaded solutions
- Sharing data across threads
- Synchronization and Deadlock
- Immutable objects

13. Concurrency

- Creating Atomic variables
- Using Read-Write Locks
- Thread-safe collections
- Concurrenct synchronizers (Semaphore, Phaser, and others)
- Executors and ThreadPools to concurrently schedule tasks
- Parallelism and the Fork-Join framework

14. Database Application with JDBC

- Layout of the JDBC API
- JDBC divers
- Queries and results

- PreparedStatement and CallableStatement
- Transactions
- RowSet 1.1 RowSetProvider and RowSetFactory
- The DAO Pattern and JDBC

15. Localization

- Advantages of localization
- Defining locale
- Read and set locale using the Locale object
- Resource bundles
- Format messages, dates and number

THANK YOU…