Suggested Teaching Guidelines for

Object Oriented Programming with Java 8

PG-DBDA September 2023

Duration: 46 class room hours + 44 Lab hours

Objective: To reinforce knowledge of Java Programming

Prerequisites: Knowledge of Linux command, Oops concepts and any programming language

Evaluation method:
Theory exam – 40% weightage
Lab exam – 40% weightage
Internal exam – 20% weightage

List of Books / Other training material

Reference:

1. Java Server Programming (J2EE 1.7 Edition) Black Book by Dreamtech Software Team
2. Java 8 Programming Black Book by Dreamtech Press
3. Core Java : Fundamentals - Volume 1 Gary Cornell, Cay S. Horstmann/ Pearson
4. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
5. Core Java : Advanced Features - Volume 2 Gary Cornell, Cay S. Horstmann/ Pearson
6. Beginning Java 2 by Ivor Horton; Wrox Publication
7. The Complete Reference Java Eight Edition, Herbert Schidt/ TMH
8. Object-Oriented Analysis and Design with applications by Booch
9. Core Java 8 for Beginners by Sharanam Shah, Vaishali Shah / Shroff Publishers & Distributors
10. Murach's Java Programming 4th edition by Joel Murach / Shroff Publishers & Distributors
11. Advanced Java programming by Uttam K Roy / Oxford University press
12. Sun Certified Enterprise Architect For Java EE Study Guide by Cade, 2nd Edition (Paperback)
13. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
14. Professional Java EE Design Patterns by Murat Yener, Alex Theedom, Reza Rahman

Note: Each session having 2 Hours

Session 1, 2 and 3:
Lecture

- Java 8 Basics :Overview of Java, Features of Java, Scope of variables
- Object Oriented Concepts
- JDK and its usage (Java Compiler, Java Runtime, Java Debugger, Java doc)
- Working with Data Types: Structure of a Java Class, Importing Packages, Difference between object reference variables and primitive variables, how to read or write to object fields)

Session 4:
Lecture

- Object's lifecycle(creation, reassignment, garbage collection: new, finalize)
- Wrapper classes (Boolean, Double and Integer)

- Operators (Unary, Binary, Arithmetic, Assignment, Compound, Relational, Logical, Equality) and Control Statements (if, if-else, for, while, switch, do-while, break and continue, ternary constructs)

Assignment – Lab:

1. Create Java Program for simple calculator, compile & test it.

Session 5:

Lecture

- Packages and classpath
- Arrays
- Understanding of String Class, StringBuilder Class, StringBuffer class
- Methods and Encapsulation: Methods, Access Modifiers, Method Overloading, Passing Data, Creating Constructors, Immutable Classes

Assignment – Lab:

Get yourself acquainted with java environment. Build a class Emp, which contains details about the employee and compile and run its instance

Assignment – Reading:

Study the book Java FAQ

Session 6:

Lecture

- Class Inheritance, Abstract Classes, Inner Classes, Interface and Implementation classes.
- Understanding Polymorphism: Object vs Reference, Object Casting, Virtual Methods, Method Overriding

Assignment – Lab:

Create an inner class for a manager, which contains information about the manager. Use the appropriate interfaces. Create an anonymous inner class for Tech. Members using the Session one assignment

Session 7 & 8:

Lecture

- Exception-Handling: Basics, Role of Exceptions, Types
- Using try and catch, Multiple Catch, Nested try (throw, throws, finally)
- Built-in Exceptions, Runtime Exceptions Checked Exceptions, Errors
- Creating own Exception Subclasses

Assignment – Lab:

Create a user defined exception to check whether your employee exist in your data structure and using the catch and finally block. Redeem an appropriate solution

Session 9:

Lecture

- Enumerations, Auto boxing, and Annotations

Assignment – Lab:

Create sample classes to understand boxing & unboxing. Use different methods of java defined wrapper classes.

Session 10 & 11:

Lecture

- Java API: java.util, java.lang, java.math

Assignment – Lab:

Create an appropriate data structures to store your employee object and use the java.util.package properties.

Session 12 & 13:

Lecture

- Generics and Collections
- TCP and IP
- Communication with TCP/IP Protocol

Assignment – Lab:

1. Implement String class and util package
2. Using the collection framework define an appropriate interface to your above application
3. Implement to Send File Contents (two way communication Java)
4. A Simple Java TCP Server and TCP Client
5. Create a user defined exception to check whether your employee exist in your data structure and using the catch and finally block. Redeem an appropriate solution

Session 14:

Lecture

- Functional Programming Overview
- Functional Interfaces
- Explore java.util.function package : Predicate, Map, Consumer, Supplier
- Lambda Expressions
- Impact of Functional programming upon Collection Framework

Session 15 & 16:

Lecture

- Introduction to Streams
- Streams vs. Collections
- java.util.stream.Stream API
- Types of Primitive Streams : IntStream,LongStream,DoubleStream & its API
- Different operations on streams : filter, map, reduce, sort, flatMap, anyMatch, count, boxing.
- Overview of Java 8 Date Time API

Assignment – Lab:

1. Process bank accounts collection using stream functions.

Session 17 & 18:

Lecture

- Java Concurrency: Using threads in Java, Life cycle of thread
- Advantages and issues
- Thread class, thread groups
- The Runnable interface

Session 19:

Lecture

- Synchronization, Inter-Thread communication
- Executor Framework overview

Assignment – Lab:

- Using Multi-Threading create concurrent java application, to write data to file in a thread safe manner.Apply Thread safety to Collection Framework API classes
- Apply multi-threading techniques to file handling and ensure thread safety.

Session: 20:

Lecture

- The java.io Package
- Files
- Byte Streams and Unicode Character Streams
- Persistence of objects
- Object Serialization Methods

Assignment – Lab:

Make your above Employee, manger classes objects persistent.

Session: 21:

Lecture: Reflection in Java

- Java Reflection Classes, Methods, Getter Setters, Constructors, Annotations, generics, Arrays, Dynamic method invocation

Assignment – Lab:

Create a new array, whose size and component types are not known until runtime, and then modify the array's components

Session: 22:

Lecture: Reflection in Java & JVM Architecture

- Why Java Reflection
- What is it ?
- Basic Reflection API for finding out details of the class name, super classes & interfaces.
- What is a Java Virtual Machine?
- The Architecture of the Java Virtual Machine

Assignment – Lab:

Implement Java reflection API for Modifying and finding out details of the class name, super classes & interfaces.

Session 23:

Lecture

- Introduction of JDBC API
- JDBC Architecture
- JDBC Drivers
- Drivers, Connection, Statement, Prepared Statement and Result Set interfaces and their relationship to provider implementations
- Writing JDBC Application along with DAO & POJO Layers

- Stored Procedures and functions invocation

Assignment – Lab:

- Build an application to get student's details using database.
- Invoke stored procedure & a function.