

*Suggested Teaching Guidelines for*  
**Java Programming**  
**PG-DBDA August 2025**

**Duration:** 34 class room hours + 36 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**

**Text Book:**

Java for Dummies, Barry Burd, Wiley India, Seventh Edition.

**Reference:**

1. Java Server Programming (J2EE 1.7 Edition) Black Book by Dreamtech Software Team
2. Core Java : Fundamentals - Volume 1 Gary Cornell, Cay S. Horstmann/ Pearson
3. Advanced Java programming by Uttam K Roy / Oxford University press

**Note:** Each session having 2 Hours of Theory & 2 Hours of Lab unless indicated otherwise.

**Session 1 & 2:**

- Java 8 Basics :Overview of Java, Features of Java, Scope of variables
- Object Oriented Concepts
- Java Virtual Machine –Overview
- 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 3:**

- 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)

*Suggested Teaching Guidelines for*  
**Java Programming**  
**PG-DBDA August 2025**

**Assignment – Lab:**

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

**Session 4:**

- 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

**Session 5:**

- 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 6:**

- 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

*Suggested Teaching Guidelines for*  
**Java Programming**  
**PG-DBDA August 2025**

**Session 7:**

- Enumerations
- Auto boxing
- Annotations

**Assignment – Lab:**

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

**Session 8:**

- java.util, java.lang

**Assignment – Lab:**

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

**Session 9 & 10:**

- Generics and Collections

**Assignment – Lab:**

1. Implement String class and util package
2. Using the collection framework define an appropriate interface to your above application

**Session 11:**

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

**Session 12:**

- Introduction to Streams
- Streams vs. Collections
- java.util.stream.Stream API
- Types of Primitive Streams : IntStream, LongStream, DoubleStream & its API

*Suggested Teaching Guidelines for*

**Java Programming**

**PG-DBDA August 2025**

- Different operations on streams : filter, map, reduce, sort, flatMap, anyMatch, count, boxing.
- Overview of Java 8/17 Date Time API

**Assignment – Lab:**

1. Process bank accounts collection using stream functions.

**Session 13 & 14:**

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

**Session: 15:**

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

**Assignment – Lab:**

- Make your above Employee, manager classes object persistent.

**Session: 16:**

- JVM Architecture
- Java Reflection
- Basic Reflection API for finding out details of the class name, super classes & interfaces.

**Assignment – Lab:**

- Use the Java Reflection API to print the name of a given class, its superclass, and all implemented interfaces.
- Create a Java program that lists all the methods declared in an interface implemented by a class using reflection.

*Suggested Teaching Guidelines for*

**Java Programming**

**PG-DBDA August 2025**

**Session 17: (2T + 4L)**

- Introduction of JDBC API
- JDBC Architecture
- JDBC Drivers
- Drivers
- Connection
- Statement, Prepared Statement and Result Set interfaces and their relationship to provider implementations

**Assignment – Lab:**

- Build an application to get student's details using database.