# java – Lesson Plan

Generated on: 2025-07-20 16:01:13

## Java - Advanced Level (12 Weeks, 1 Hour/Week)  
  
\*\*Target Learning Outcomes:\*\* Students will demonstrate advanced proficiency in Java programming, including design patterns, concurrency, and advanced JVM features.  
  
  
\*\*Week 1: Design Patterns I - Creational Patterns\*\*  
  
\* \*\*Topic:\*\* Creational Design Patterns  
\* \*\*Subtopics:\*\* Singleton, Factory, Abstract Factory, Builder, Prototype  
\* \*\*Activities:\*\* Code examples demonstrating each pattern, problem-solving exercises involving pattern selection, mini-project: implementing a simple game using appropriate creational patterns.  
  
\*\*Week 2: Design Patterns II - Structural & Behavioral Patterns\*\*  
  
\* \*\*Topic:\*\* Structural & Behavioral Design Patterns  
\* \*\*Subtopics:\*\* Adapter, Facade, Decorator (Structural), Observer, Strategy, Template Method, Command (Behavioral)  
\* \*\*Activities:\*\* Code examples, discussion on choosing appropriate patterns for specific scenarios, refactoring existing code to incorporate design patterns.  
  
\*\*Week 3: Concurrency Basics - Threads & Synchronization\*\*  
  
\* \*\*Topic:\*\* Introduction to Concurrency in Java  
\* \*\*Subtopics:\*\* Thread creation and lifecycle, synchronization using `synchronized` keyword, `volatile` keyword, deadlock scenarios.  
\* \*\*Activities:\*\* Hands-on exercises creating and managing threads, debugging deadlock situations, code review focusing on thread safety.  
  
\*\*Week 4: Concurrency - Advanced Techniques\*\*  
  
\* \*\*Topic:\*\* Advanced Concurrency Techniques  
\* \*\*Subtopics:\*\* `ReentrantLock`, `Semaphore`, `CountDownLatch`, `CyclicBarrier`, Executor framework (`ExecutorService`, `ThreadPoolExecutor`), Future and Callable.  
\* \*\*Activities:\*\* Implementation of concurrent data structures, performance comparisons of different concurrency approaches, solving multi-threaded programming problems.  
  
  
\*\*Week 5: Java Memory Model & Garbage Collection\*\*  
  
\* \*\*Topic:\*\* Understanding the Java Memory Model (JVM)  
\* \*\*Subtopics:\*\* Heap, Stack, PermGen/Metaspace, Garbage Collection algorithms (Mark and Sweep, G1GC, ZGC), Tuning Garbage Collection.  
\* \*\*Activities:\*\* Analyzing heap dumps, understanding GC logs, experimenting with different GC options using JVM flags, profiling memory usage of application.  
  
  
\*\*Week 6: Networking in Java\*\*  
  
\* \*\*Topic:\*\* Building Network Applications  
\* \*\*Subtopics:\*\* Sockets (TCP/UDP), Server and Client implementation, NIO (Non-blocking I/O).  
\* \*\*Activities:\*\* Building a simple chat application using sockets, exploring NIO for improved performance.  
  
\*\*Week 7: Java Reflection & Introspection\*\*  
  
\* \*\*Topic:\*\* Runtime code manipulation  
\* \*\*Subtopics:\*\* `Class` object, accessing fields and methods dynamically, using reflection for testing and dynamic code generation.  
\* \*\*Activities:\*\* Building a simple framework using reflection, exploring security implications of reflection.  
  
\*\*Week 8: Java Streams API\*\*  
  
\* \*\*Topic:\*\* Efficient Data Processing  
\* \*\*Subtopics:\*\* Functional interfaces, lambda expressions, stream operations (map, filter, reduce, collect), parallel streams.  
\* \*\*Activities:\*\* Processing large datasets using streams, comparing performance of sequential and parallel streams.  
  
  
\*\*Week 9: Testing in Java\*\*  
  
\* \*\*Topic:\*\* Unit Testing & Mocking  
\* \*\*Subtopics:\*\* JUnit, Mockito, Test Driven Development (TDD) principles.  
\* \*\*Activities:\*\* Writing unit tests for existing code, creating mock objects, practicing TDD workflow.  
  
  
\*\*Week 10: Advanced Java Generics & Collections\*\*  
  
\* \*\*Topic:\*\* Extending Generics and Collections  
\* \*\*Subtopics:\*\* Wildcards, bounded wildcards, custom generic classes, advanced collection usage (e.g., concurrent collections).  
\* \*\*Activities:\*\* Designing generic data structures, implementing custom collections with specific functionalities.  
  
  
\*\*Week 11: Working with Databases (JDBC)\*\*  
  
\* \*\*Topic:\*\* Connecting Java applications to databases  
\* \*\*Subtopics:\*\* JDBC API, database connection, prepared statements, transactions.  
\* \*\*Activities:\*\* Connecting to a database (e.g., MySQL, PostgreSQL), performing CRUD operations, handling transactions.  
  
  
\*\*Week 12: Project Work & Review\*\*  
  
\* \*\*Topic:\*\* Final Project Presentation & Course Review  
\* \*\*Subtopics:\*\* Students present their final projects, addressing challenges encountered and lessons learned. Course review and Q&A session.  
\* \*\*Activities:\*\* Project presentations, open discussion on advanced Java topics, feedback session.  
  
  
\*\*Note:\*\* This is a suggested plan. The specific topics and activities can be adjusted based on student progress and interests. The instructor should also provide relevant reading materials and online resources. Each week should include some coding exercises and problem-solving activities.

---  
Watermark: TeachMate AI | Version 1.0