# Java Daily Folder Structure - 30 Day Plan

## Initial Setup

First, create this base structure:

Copy

JavaMastery/

├── src/

│ ├── main/

│ │ ├── java/

│ │ │ └── com/

│ │ │ └── yourname/

│ │ └── resources/

│ └── test/

│ └── java/

│ └── com/

│ └── yourname/

└── pom.xml

## Week 1: Java Fundamentals

### Day 1: Java Setup & First Steps

Copy

com/yourname/

├── fundamentals/

│ └── basics/

│ ├── HelloWorld.java

│ ├── Variables.java

│ ├── DataTypes.java

│ └── Operators.java

└── hackerrank/

└── day1/

├── WelcomeToJava.java

├── StdinAndStdout1.java

├── JavaIfElse.java

└── StdinAndStdout2.java

### Day 2: Java Control Flow

Copy

com/yourname/

├── fundamentals/

│ └── basics/

│ ├── IfStatements.java

│ ├── SwitchStatements.java

│ ├── ForLoops.java

│ ├── WhileLoops.java

│ └── SimpleCalculator.java

└── hackerrank/

└── day2/

├── JavaLoops1.java

├── JavaLoops2.java

├── JavaDatatypes.java

└── JavaEndOfFile.java

### Day 3: Java Classes & Objects I

Copy

com/yourname/

├── fundamentals/

│ └── oop/

│ ├── Student.java

│ ├── Course.java

│ ├── ClassesDemo.java

│ └── ConstructorsDemo.java

└── hackerrank/

└── day3/

├── JavaOOP.java

├── JavaInheritance1.java

└── JavaMethodOverriding.java

### Day 4: Java Classes & Objects II

Copy

com/yourname/

├── fundamentals/

│ └── oop/

│ ├── EncapsulationDemo.java

│ ├── StaticDemo.java

│ ├── FinalDemo.java

│ └── OverloadingDemo.java

└── hackerrank/

└── day4/

├── JavaInheritance2.java

├── JavaAbstractClass.java

└── JavaInterface.java

### Day 5: Java Inheritance & Polymorphism

Copy

com/yourname/

├── fundamentals/

│ └── inheritance/

│ ├── Person.java

│ ├── Student.java

│ ├── Teacher.java

│ └── SchoolDemo.java

└── hackerrank/

└── day5/

├── SuperKeyword.java

├── InstanceofKeyword.java

└── JavaIterator.java

### Day 6: Java Abstract Classes & Interfaces

Copy

com/yourname/

├── fundamentals/

│ └── inheritance/

│ ├── Shape.java (abstract class)

│ ├── Circle.java

│ ├── Rectangle.java

│ ├── Drawable.java (interface)

│ └── Printable.java (interface)

└── hackerrank/

└── day6/

├── JavaInterface.java

├── ExceptionHandling.java

└── TryCatch.java

### Day 7: Java Packages & Exception Handling

Copy

com/yourname/

├── fundamentals/

│ ├── exceptions/

│ │ ├── BasicExceptions.java

│ │ ├── CustomExceptions.java

│ │ └── ExceptionHandlingDemo.java

│ └── packages/

│ ├── util/

│ │ └── Helper.java

│ └── model/

│ └── User.java

├── projects/

│ └── taskmanager/

│ ├── model/

│ │ ├── Task.java

│ │ └── Priority.java

│ ├── service/

│ │ └── TaskService.java

│ └── TaskManagerApp.java

└── hackerrank/

└── day7/

├── JavaFactoryPattern.java

├── JavaHashset.java

└── JavaGenerics.java

## Week 2: Java Intermediate

### Day 8: Java Collections Framework I

Copy

com/yourname/

├── intermediate/

│ └── collections/

│ ├── ListDemo.java

│ ├── ArrayListDemo.java

│ ├── LinkedListDemo.java

│ └── SetDemo.java

├── projects/

│ └── taskmanager/

│ ├── repository/

│ │ └── TaskRepository.java (using collections)

│ └── ...

└── hackerrank/

└── day8/

├── JavaList.java

├── JavaMap.java

└── JavaSort.java

### Day 9: Java Collections Framework II

Copy

com/yourname/

├── intermediate/

│ └── collections/

│ ├── MapDemo.java

│ ├── HashMapDemo.java

│ ├── TreeMapDemo.java

│ └── QueueDemo.java

├── projects/

│ └── taskmanager/

│ └── ... (enhanced with Maps)

└── hackerrank/

└── day9/

├── JavaPriorityQueue.java

├── JavaBitSet.java

└── JavaComparator.java

### Day 10: Java Generics

Copy

com/yourname/

├── intermediate/

│ └── generics/

│ ├── Box.java

│ ├── Pair.java

│ ├── GenericMethodsDemo.java

│ └── WildcardDemo.java

├── projects/

│ └── taskmanager/

│ └── ... (enhanced with generics)

└── hackerrank/

└── day10/

└── [algorithm challenges]

### Day 11: Java I/O Fundamentals

Copy

com/yourname/

├── intermediate/

│ └── io/

│ ├── FileDemo.java

│ ├── FileInputStreamDemo.java

│ ├── FileReaderDemo.java

│ └── ScannerDemo.java

├── projects/

│ └── taskmanager/

│ ├── persistence/

│ │ └── FileTaskRepository.java

│ └── ...

└── hackerrank/

└── day11/

├── JavaRegex.java

├── JavaRegex2.java

└── ValidUsernameRegex.java

### Day 12: Java Serialization & Date/Time API

Copy

com/yourname/

├── intermediate/

│ ├── io/

│ │ ├── SerializationDemo.java

│ │ └── DeserializationDemo.java

│ └── datetime/

│ ├── LocalDateDemo.java

│ ├── LocalTimeDemo.java

│ └── LocalDateTimeDemo.java

├── projects/

│ └── taskmanager/

│ └── ... (add date handling)

└── hackerrank/

└── day12/

├── JavaBigDecimal.java

├── JavaBigInteger.java

└── JavaPrimalityTest.java

### Day 13: Java Regular Expressions

Copy

com/yourname/

├── intermediate/

│ └── regex/

│ ├── PatternMatcherDemo.java

│ ├── RegexGroupsDemo.java

│ ├── RegexReplaceDemo.java

│ └── EmailValidator.java

├── projects/

│ └── taskmanager/

│ └── utils/

│ └── Validators.java

└── hackerrank/

└── day13/

├── PatternSyntaxChecker.java

├── UsernameChecker.java

└── TagContentExtractor.java

### Day 14: Java Multi-threading Basics

Copy

com/yourname/

├── intermediate/

│ └── multithreading/

│ ├── ThreadDemo.java

│ ├── RunnableDemo.java

│ ├── ThreadLifecycleDemo.java

│ └── ThreadPriorityDemo.java

├── projects/

│ └── taskmanager/

│ └── ... (add background task)

└── hackerrank/

└── day14/

└── [contest solutions]

## Week 3: Java Advanced

### Day 15: Java Thread Synchronization

Copy

com/yourname/

├── advanced/

│ └── concurrency/

│ ├── SynchronizedMethodDemo.java

│ ├── SynchronizedBlockDemo.java

│ ├── VolatileDemo.java

│ └── ThreadSafeCollections.java

├── projects/

│ └── taskmanager/

│ └── ... (make thread-safe)

└── hackerrank/

└── day15/

└── [algorithm challenges]

### Day 16: Java Concurrency Utilities I

Copy

com/yourname/

├── advanced/

│ └── concurrency/

│ ├── ExecutorServiceDemo.java

│ ├── CallableFutureDemo.java

│ ├── CountDownLatchDemo.java

│ └── CyclicBarrierDemo.java

├── projects/

│ └── librarymanagement/

│ ├── model/

│ │ ├── Book.java

│ │ └── Member.java

│ └── LibraryApp.java

└── hackerrank/

└── day16/

└── [medium challenges]

### Day 17: Java Concurrency Utilities II

Copy

com/yourname/

├── advanced/

│ └── concurrency/

│ ├── AtomicVariablesDemo.java

│ ├── LockDemo.java

│ ├── BlockingQueueDemo.java

│ └── CompletableFutureDemo.java

├── projects/

│ └── librarymanagement/

│ └── ... (add concurrent operations)

└── hackerrank/

└── day17/

└── [concurrency challenges]

### Day 18: Java Lambda Expressions

Copy

com/yourname/

├── advanced/

│ └── lambdas/

│ ├── BasicLambdaDemo.java

│ ├── FunctionalInterfacesDemo.java

│ ├── MethodReferencesDemo.java

│ └── LambdaCollectionsDemo.java

├── projects/

│ └── librarymanagement/

│ └── ... (refactor with lambdas)

└── hackerrank/

└── day18/

└── [functional programming]

### Day 19: Java Stream API I

Copy

com/yourname/

├── advanced/

│ └── streams/

│ ├── StreamCreationDemo.java

│ ├── StreamFilterMapDemo.java

│ ├── StreamSortedDemo.java

│ └── OptionalDemo.java

├── projects/

│ └── librarymanagement/

│ └── ... (add stream processing)

└── hackerrank/

└── day19/

└── [stream challenges]

### Day 20: Java Stream API II

Copy

com/yourname/

├── advanced/

│ └── streams/

│ ├── ParallelStreamsDemo.java

│ ├── CollectorsDemo.java

│ ├── GroupingPartitioningDemo.java

│ └── CustomCollectorDemo.java

├── projects/

│ └── librarymanagement/

│ └── ... (enhance with advanced streams)

└── hackerrank/

└── day20/

└── [data analysis challenges]

### Day 21: Java Modules & New Features

Copy

com/yourname/

├── advanced/

│ └── features/

│ ├── VarDemo.java

│ ├── TextBlocksDemo.java

│ ├── RecordsDemo.java

│ └── SealedClassesDemo.java

├── projects/

│ └── librarymanagement/

│ └── module-info.java

└── hackerrank/

└── day21/

└── [practice challenges]

## Week 4: Java Professional Skills

### Day 22: Java Design Patterns I

Copy

com/yourname/

├── professional/

│ └── patterns/

│ ├── singleton/

│ │ └── SingletonDemo.java

│ ├── factory/

│ │ └── FactoryDemo.java

│ ├── builder/

│ │ └── BuilderDemo.java

│ └── prototype/

│ └── PrototypeDemo.java

├── projects/

│ └── ecommerce/

│ ├── model/

│ │ ├── Product.java

│ │ └── User.java

│ └── EcommerceApp.java

└── hackerrank/

└── day22/

└── [design challenges]

### Day 23: Java Design Patterns II

Copy

com/yourname/

├── professional/

│ └── patterns/

│ ├── observer/

│ │ └── ObserverDemo.java

│ ├── strategy/

│ │ └── StrategyDemo.java

│ ├── decorator/

│ │ └── DecoratorDemo.java

│ └── adapter/

│ └── AdapterDemo.java

├── projects/

│ └── ecommerce/

│ └── ... (apply patterns)

└── hackerrank/

└── day23/

└── [pattern implementation]

### Day 24: SOLID Principles

Copy

com/yourname/

├── professional/

│ └── solid/

│ ├── SingleResponsibilityDemo.java

│ ├── OpenClosedDemo.java

│ ├── LiskovSubstitutionDemo.java

│ ├── InterfaceSegregationDemo.java

│ └── DependencyInversionDemo.java

├── projects/

│ └── ecommerce/

│ └── ... (refactor with SOLID)

└── hackerrank/

└── day24/

└── [design challenges]

### Day 25: Java Testing with JUnit

Copy

com/yourname/

├── professional/

│ └── testing/

│ ├── Calculator.java

│ └── StringUtils.java

├── projects/

│ └── ecommerce/

│ └── ... (add features)

├── hackerrank/

│ └── day25/

│ └── [timed contests]

└── test/

└── java/

└── com/

└── yourname/

├── professional/

│ └── testing/

│ ├── CalculatorTest.java

│ └── StringUtilsTest.java

└── projects/

└── ecommerce/

└── ... (tests)

### Day 26: Java Mocking with Mockito

Copy

com/yourname/

├── professional/

│ └── testing/

│ ├── service/

│ │ └── UserService.java

│ └── repository/

│ └── UserRepository.java

├── projects/

│ └── ecommerce/

│ └── ... (add features)

├── hackerrank/

│ └── day26/

│ └── [timed contests]

└── test/

└── java/

└── com/

└── yourname/

├── professional/

│ └── testing/

│ └── service/

│ └── UserServiceTest.java

└── projects/

└── ecommerce/

└── ... (mockito tests)

### Day 27: Java Build Tools & CI/CD

Copy

com/yourname/

├── professional/

│ └── build/

│ └── MavenDemo.java

├── projects/

│ └── ecommerce/

│ └── ... (finalize features)

└── hackerrank/

└── day27/

└── [timed contests]

### Day 28: Java Data Structures Implementation

Copy

com/yourname/

├── professional/

│ └── algorithms/

│ ├── LinkedListImpl.java

│ ├── StackImpl.java

│ ├── QueueImpl.java

│ └── BinaryTreeImpl.java

├── projects/

│ └── ecommerce/

│ └── ... (optimize)

└── hackerrank/

└── day28/

└── [mock interview]

### Day 29: Algorithm Practice

Copy

com/yourname/

├── professional/

│ └── algorithms/

│ ├── StringAlgorithms.java

│ ├── DynamicProgramming.java

│ ├── GreedyAlgorithms.java

│ └── GraphAlgorithms.java

├── projects/

│ └── ecommerce/

│ └── ... (optimize)

└── hackerrank/

└── day29/

└── [mock interview]

### Day 30: Live Coding Simulation & Interview Preparation

Copy

com/yourname/

├── interviews/

│ ├── SystemDesignDemo.java

│ ├── ObjectModelingDemo.java

│ ├── PerformanceTipsDemo.java

│ └── BestPracticesDemo.java

├── projects/

│ └── ecommerce/

│ └── ... (final optimizations)

└── hackerrank/

└── day30/

└── [final mock interview]

## HackerRank Solutions Structure

For each day, create a folder in the hackerrank directory:

Copy

com/yourname/hackerrank/day{X}/

Where X is the day number (1-30), and place your solutions to the daily HackerRank challenges in this folder.

## Project Progression

### Week 1: Task Manager Project

Set up a basic console Task Manager application in:

Copy

com/yourname/projects/taskmanager/

### Week 3: Library Management System

Start a more advanced Library Management System:

Copy

com/yourname/projects/librarymanagement/

### Week 4: E-commerce Platform

Build your comprehensive final project:

Copy

com/yourname/projects/ecommerce/