# 🧠 Java Mastery Syllabus for Beginners (Career-Focused Track)

## 📘 Module 1: Java Basics (1–2 Weeks)

Goal: Understand the Java language fundamentals.

### Topics:

- Introduction to Programming & Java
- Java Installation (JDK, IntelliJ / VS Code / Eclipse setup)
- Writing your first Java program
- Basic Syntax (keywords, identifiers, comments)
- Data Types and Variables
- Operators (arithmetic, logical, relational, assignment)
- Input & Output (Scanner class, System.out)
- Type Casting and Conversion
- Practice: Simple calculator, swapping numbers, average of 3 numbers

---

## 🧩 Module 2: Control Flow Statements (1 Week)

Goal: Learn to control program execution flow.

### Topics:

- Conditional statements: if, if-else, nested if, switch
- Looping: for, while, do-while
- Break, Continue, Return
- Practice:
  - Print patterns
  - Factorial, Fibonacci, prime numbers
  - Number guessing game

---

## 🧮 Module 3: Arrays and Strings (1–2 Weeks)

Goal: Work with data collections.

### Topics:

- 1D & 2D Arrays
- Traversing and updating arrays
- Common algorithms: searching, sorting (bubble, selection, insertion)
- String class and StringBuilder
- String operations (length, substring, equals, split, etc.)
- Practice:
  - Reverse string/array
  - Count vowels/consonants
  - Find duplicates

---

## 🧱 Module 4: Methods & Recursion (1 Week)

Goal: Learn modular and reusable code design.

### Topics:

- Method declaration and calling
- Parameters, return types, overloading
- Recursion basics and examples
- Practice:
  - Factorial using recursion
  - Sum of digits recursively
  - Palindrome check using recursion

---

## ⚙️ Module 5: Object-Oriented Programming (2–3 Weeks)

Goal: Understand OOP — the heart of Java.

### Topics:

- What is OOP?
- Classes and Objects
- Constructors & this keyword
- Access modifiers (public, private, protected)
- Static keyword
- Inheritance (super keyword, method overriding)
- Polymorphism (compile-time and runtime)
- Encapsulation and Abstraction
- final keyword
- Practice:
    - Student/Employee management program
    - Bank account system (deposit/withdraw)
    - Vehicle or Shape inheritance example

---

## 🧰 Module 6: Exception Handling & Debugging (1 Week)

Goal: Handle runtime errors safely.

### Topics:

- Types of errors (compile, runtime, logical)
- Try, catch, finally blocks
- Throw and throws
- Custom exceptions
- Practice:
    - Divide by zero handling
    - Custom Age validation exception

---

## 📦 Module 7: Collections Framework (2 Weeks)

Goal: Work with dynamic data structures.

Topics:

- What is the Collection Framework?
- List, Set, and Map interfaces
- ArrayList, LinkedList, HashSet, TreeSet
- HashMap, LinkedHashMap, TreeMap
- Iterators and for-each loop
- Practice:
  - Count word frequency
  - Student score management using HashMap

---

## 🗂️ Module 8: File Handling (1 Week)

Goal: Learn to read and write files.

Topics:

- File class
- Reading files (FileReader, BufferedReader)
- Writing files (FileWriter, BufferedWriter)
- Practice:
  - Copy text from one file to another
  - Count lines and words in a file

---

## 🌐 Module 9: JDBC (Database Connectivity) (2 Weeks)

Goal: Connect Java with databases.

Topics:

- Introduction to JDBC
- Steps to connect Java and MySQL

- CRUD operations
- PreparedStatement vs Statement
- Exception handling in JDBC
- Practice:
  - Student database app
  - Employee payroll management system

---

## 👩‍💻 Module 10: Advanced Java (2–3 Weeks)

Goal: Get job-ready for full-stack or backend roles.

### Topics:

- Multi-threading
- Synchronization
- Lambda Expressions (Java 8)
- Streams API
- Introduction to Servlets & JSP
- Basic Spring Boot overview (optional intro)

---

## 🎯 Final Phase: Projects & Interview Prep (Ongoing)

### Projects:

1. Console-based Library Management System
2. CRUD Web App (Java + JDBC + MySQL)
3. File-based To-Do List App
4. Mini Bank System

### Interview Prep:

- 100 Java MCQs
- DSA + OOP Questions

- Practice from LeetCode / HackerRank (Java track)