1. Start with Java Fundamentals

- **Learn basic syntax and core concepts**:
    - Basic syntax
    - Data types and variables
    - Conditionals
    - Loops
    - Functions
    - Working with files and APIs
    - Packages, classes, and interfaces

At this stage, the goal is to understand how to write basic programs and get familiar with Object-Oriented Programming (OOP).

2. Dive into the JVM and internal workings

- Understand **how the JVM (Java Virtual Machine) works**:
    - Memory management
    - Garbage collection
    - Exception handling
    - Threads and basic multithreading

This knowledge is crucial for understanding performance and resource management.

3. Collections and data handling

- **Learn the Collection Framework**:
    - Collection Framework
    - Generics
    - Streams API
- **Serialization and networking**:
    - Serialization
    - Networking & Sockets

These skills will help you efficiently manage data and interact with other systems through networking.

4. Master build tools and libraries:

- **Build tools**:
    - Maven or Gradle
- **ORM frameworks**:
    - Hibernate, Spring Data JPA (for database interaction)
- **Web frameworks**:

    - Spring, Spring Boot (key frameworks for Java development)

5. Testing your applications:

- Learn **testing tools**:
    - Unit testing with JUnit or TestNG
    - Mocking with Mockito
    - REST Assured for API testing

6. Additional skills:

- Logging with **Log4j2**, Logback, or TinyLog.
- **Database management** with JDBC or JPA.

7. Certification Progression

- Once you have mastered the essential tools and approaches for Java SE 8, you can pursue the **OCA (Java SE 8)** certification.
- After OCA, dive deeper into professional topics and prepare for the **OCP certification for Java SE 11 or SE 17**.

Final Goal: Build full-fledged applications using Spring and advanced testing techniques.

This path will guide you from the fundamentals to more advanced skills, allowing you to confidently pursue certification at different stages.