 # Java Full Stack - Syllabus

Java Core

1. Introduction to Java

- **. History and Evolution of Java**
- **. Java's Place in Modern Development**

2. Java Installation

- **. Installing JDK and JRE**
- **. Setting up Environment Variables**

3. Basic Concepts

- **. Syntax and Semantics**
- **. Variables, Data Types, and Operators**
- **. Control Structures (if-else, switch, loops)**

4. Object-Oriented Programming (OOP)

- **. Classes and Objects**
- **. Inheritance, Polymorphism, Encapsulation, and Abstraction**
- **. Interfaces and Abstract Classes**

5. Core Libraries

- **. Collections Framework (List, Set, Map)**
- **. Exception Handling**
- **. Input/Output Streams**
- **. Multithreading and Concurrency**

MySQL

1. Introduction to MySQL

- **. What is MySQL?**
- **. MySQL vs. Other Databases**

2. Installation and Setup

- **. Installing MySQL Server**
- **. Using MySQL Workbench**
- **. Command-Line Basics**

3. Database Concepts

- **. Databases, Tables, and Relationships**

- **. Data Types in MySQL**

4. CRUD Operations

- **. Creating, Reading, Updating, and Deleting Records**
- **. SQL Syntax and Queries**

5. Advanced SQL

- **. Joins (Inner, Outer, Left, Right)**
- **. Indexes**
- **. Stored Procedures and Functions**
- **. Triggers**

Frontend (HTML and CSS)

1. HTML Basics

- **. HTML Structure**
- **. Elements and Attributes**
- **. Forms and Input Elements**
- **. HTML5 New Elements**

2. CSS Basics

- **. CSS Syntax**
- **. Selectors, Properties, and Values**
- **. Box Model (Margin, Border, Padding, Content)**
- **. Positioning and Layout (Float, Flexbox, Grid)**

3. Advanced CSS

- **. CSS3 New Features**
- **. Transitions and Animations**
- **. Responsive Design (Media Queries)**

Spring Boot

1. Introduction to Spring Boot

- **. Overview of Spring Framework**
- **. Advantages of Spring Boot**

2. Setting Up Spring Boot

- **. Installing and Configuring Spring Boot**
- **. Using Spring Initializr**

- **. Project Structure**

3. Core Concepts

- **. Dependency Injection**
- **. Spring Boot Annotations**
- **. Application Properties and Configuration**

4. Building RESTful Services

- **. Creating Controllers**
- **. Handling Requests and Responses**
- **. Using Spring Data JPA for Database Integration**

5. Security

- **. Introduction to Spring Security**
- **. Implementing Basic Authentication**

6. Advanced Spring Boot

- **. Custom Error Handling**
- **. Integrating with Frontend (Thymeleaf, React, Angular)**
- **. Deploying Spring Boot Applications**

Projects

Project 1: CRUD Application

- **. Description: Create a simple CRUD application using Spring Boot, Hibernate, MySQL, and a frontend with HTML/CSS/JavaScript.**
- **. Features:**
    - **User Management (Create, Read, Update, Delete)**
    - **Product Management (Create, Read, Update, Delete)**
- **. Technologies Used:**
    - **Backend: Spring Boot, Hibernate, MySQL**
    - **Frontend: HTML, CSS, JavaScript (optional: React or Angular)**
- **. Steps:**
    - **Set up Spring Boot project with dependencies for Hibernate and MySQL.**
    - **Create entity classes and repositories for users and products.**
    - **Implement CRUD operations in the service layer.**
    - **Create RESTful controllers for handling requests.**

- **Develop frontend pages for listing, adding, editing, and deleting users and products.**
- **Connect frontend with backend using AJAX or fetch API.**

Project 2: User Login and Authentication

- **. Description: Develop a user login and registration system with Spring Boot, Spring Security, Hibernate, MySQL, and a frontend using HTML/CSS/JavaScript.**
- **. Features:**
  - **User Registration**
  - **User Login**
  - **Authentication and Authorization**
  - **Role-Based Access Control**
- **. Technologies Used:**
  - **Backend: Spring Boot, Spring Security, Hibernate, MySQL**
  - **Frontend: HTML, CSS, JavaScript (optional: React or Angular)**
- **. Steps:**
  - **Set up Spring Boot project with dependencies for Spring Security, Hibernate, and MySQL.**
  - **Create entity classes for users and roles.**
  - **Configure Spring Security for authentication and authorization.**
  - **Implement user registration and login functionalities.**
  - **Create RESTful controllers for handling user-related requests.**
  - **Develop frontend pages for user registration and login.**
  - **Secure application endpoints based on user roles.**