

## Projects for Junior Developer Learning Java, Python, and Node.js

### Project 1: Basic CRUD Application

#### Objective:

Create a simple CRUD application to manage a list of books.

#### Requirements:

- Implement basic CRUD operations (Create, Read, Update, Delete).
- Use a relational database (MySQL or PostgreSQL).
- Create basic HTML pages for displaying and managing the book list.

#### Description for Developer:

Build a simple Book Management application where users can add, view, update, and delete books. Each book should have a title, author, and publication year. Implement this in three versions using Spring Boot, Django, and Express. Ensure you use a relational database to store the data.

## Project 2: User Authentication and Authorization

### Objective:

Implement user registration, login, and role-based access control.

### Requirements:

- User registration and login functionality.
- Role-based access control (e.g., admin, user).
- Secure password storage (use hashing).
- Basic user management interface.

### Description for Developer:

Create a User Management System where users can register, log in, and have different roles (admin, user). Admin users should be able to manage (create, update, delete) other users. Implement this in three versions using Spring Boot, Django, and Express, focusing on secure authentication and authorization mechanisms.

### Project 3: RESTful API for Task Management

#### Objective:

Develop a RESTful API for a task management system.

#### Requirements:

- RESTful endpoints for managing tasks (CRUD operations).
- Each task should have a title, description, status, and due date.
- Implement pagination and sorting for task retrieval.

#### Description for Developer:

Build a RESTful API for a Task Management System where users can create, update, delete, and retrieve tasks. Each task should include a title, description, status, and due date. Implement pagination and sorting for task retrieval. Develop this API using Spring Boot, Django Rest Framework, and Express.

## Project 4: E-commerce Platform Backend

### Objective:

Create a backend for a simple e-commerce platform.

### Requirements:

- Product management (CRUD operations for products).
- User authentication and authorization.
- Shopping cart functionality.
- Order processing (place, view, and manage orders).

### Description for Developer:

Develop the backend for an E-commerce Platform where users can browse products, add them to a shopping cart, and place orders. Admin users should be able to manage products. Implement user authentication and authorization. Build this backend using Spring Boot, Django, and Express.

## Project 5: Real-time Chat Application

### Objective:

Build a real-time chat application.

### Requirements:

- User authentication.
- Real-time messaging between users.
- Use WebSockets for real-time communication.
- Basic chat interface.

### Description for Developer:

Create a Real-time Chat Application where users can log in and send messages to each other in real-time. Implement user authentication and use WebSockets for real-time communication. Develop this application using Spring Boot with WebSocket support, Django Channels, and Express with Socket.io.

## Project 6: Social Media Platform Backend

### Objective:

Develop the backend for a social media platform.

### Requirements:

- User profile management.
- Post creation and management (CRUD operations for posts).
- Implement likes and comments on posts.
- Follow/unfollow functionality.
- News feed showing posts from followed users.

### Description for Developer:

Build the backend for a Social Media Platform where users can create profiles, follow other users, create posts, like, and comment on posts. Implement a news feed to show posts from followed users. Develop this backend using Spring Boot, Django, and Express.

### Summary of Challenges:

1. **Basic CRUD Application:** Focus on understanding frameworks and basic database operations.
2. **User Authentication and Authorization:** Learn about security, authentication, and role-based access control.
3. **RESTful API for Task Management:** Dive into API development, pagination, and sorting.
4. **E-commerce Platform Backend:** Explore e-commerce functionalities, including product management and order processing.
5. **Real-time Chat Application:** Understand real-time communication using WebSockets.
6. **Social Media Platform Backend:** Implement complex functionalities like user interactions, following, and a dynamic news feed.