CODING SHUTTLE

Caching and Transactions

Homework

Recap

1. Introduction to Caching
2. Spring Boot default caching
3. Redis Cache with Redis cloud
4. Database Transaction
5. Transaction Isolation Levels
6. Transaction in Spring Boot
7. Transaction Locks in Spring Data JPA

Homework

1. Implement cache expiration and eviction policies in a Spring Boot application. Use Redis or in-memory caching, and set different TTL (Time-To-Live) values for different types of cached data.

2. Configure Redis Cache in Spring Boot using Redis Cloud. Implement caching for a weather API service where frequently requested weather data for the same city is cached in Redis.

Homework

3. Create a Spring Boot Banking application that demonstrates different transaction isolation levels (e.g., READ_COMMITTED, REPEATABLE_READ, SERIALIZABLE). Simulate race conditions by performing concurrent read/write operations on the same data.

4. Implement both optimistic and pessimistic locking in a Spring Boot application. Test the difference between the two by simulating concurrent updates on the same record.