GitHub Link : [Tallati-Nikhil-Bhushan/Java-Spring-Boot-Cloud-Native-Approach-Advanced-Concepts (github.com)](github.com)

During my training, I worked extensively on a wide range of projects that focused on key technologies, including Java, Spring Boot, cloud-native architecture, microservices, and AWS. Below is a summary of the projects and concepts I handled:

1. **RESTful Services:**

   o Developed basic RESTful APIs using Spring Boot (FirstRest, FirstRestJpaH2, FirstRestJpaMySQL) with proper exception handling and database integration (H2 and MySQL).

2. **Security and Authorization:**

   o Implemented JWT authentication and authorization for a Spring Boot application (angular-spring-jwt).

   o Worked on Spring Boot security with JDBC (spring-boot-security-jdbc) and JWT (spring-boot-security-jwt).

   o Integrated OAuth with Facebook, Google, and GitHub for authentication (spring-boot-security-oauth-fb, spring-boot-security-oauth-github, spring-boot-security-oauth-google).

3. **Microservices and Resilience:**

   o Explored microservice architecture using Spring Cloud (circuit-breaker-demo-micro-service, feign-demo-service, eureka-registry-service, gateway-lb-demo).

   o Integrated Resilience4J to implement circuit breakers and fallback mechanisms for microservices (hello-client-resilience4j-demo2, hello-service-resilience4j-demo).

4. **Cloud and DevOps:**

   o Gained experience in AWS services, including deploying applications to AWS Lambda (aws-lamda-demo) and managing RDS with Beanstalk (aws-rds-beanstalk-mysql-demo).

   o Implemented CI/CD pipelines using AWS ECS (ci-cd-aws-ecs-pipeline-steps.txt) and explored Docker and Kubernetes for container orchestration (docker-demo, kubernetes-demo).

   o Created Docker images and deployed them on Kubernetes clusters (kubernetes-own-image-demo).

5. **Messaging and Streaming:**

   o Worked with Kafka for real-time streaming and messaging, developing producer-consumer applications (kafka_producer, kafka_consumer).

   o Explored RabbitMQ for message brokering (rabbitMQ-demo).

6. **Service Discovery and Load Balancing:**

   o Set up Eureka for service discovery and load balancing across multiple services (eureka-registry-lb-demo, service1-lb-demo).

7. **Cloud Configuration Management:**

- o Managed dynamic configurations in microservices using Spring Cloud Config (cloud-config-demo, cloud-config-server, config-client).

8. **Logging and Monitoring:**

- o Integrated distributed tracing using Spring Cloud Sleuth (sleuth-service-demo).

In addition to these projects, I documented key processes such as Docker commands, Kubernetes commands, and OAuth2 implementation. I also worked on several AWS CLI-related tasks and created detailed instructional files, such as AwsCliEcrSteps.txt, to streamline deployment and CI/CD pipelines.

Through these diverse projects, I gained comprehensive experience in microservices, cloud-native development, security, and cloud-based DevOps practices.