Configuration for Data Transfer Postgres to Cassandra Spring Application

Step1: Create a Spring Application

Step2: Maven Dependencies Required

* For Cassandra : Dependency for Cassandra Driver and Spring Data Cassandra

<dependency>

     <groupId>com.datastax.cassandra</groupId>

     <artifactId>cassandra-driver-core</artifactId>

    </dependency>

<dependency>

     <groupId>org.springframework.data</groupId>

     <artifactId>spring-data-cassandra</artifactId>

    </dependency>

<dependency>

     <groupId>org.springframework.boot</groupId>

     <artifactId>spring-boot-starter-data-cassandra</artifactId>

    </dependency>

* For Postgres : Dependency for Postgres and Spring Data Jpa

<dependency>

     <groupId>org.postgresql</groupId>

     <artifactId>postgresql</artifactId>

    </dependency>

<dependency>

     <groupId> org.springframework.boot</groupId>

     <artifactId> spring-boot-starter-data-jpa</artifactId>

    </dependency>

Step3: Required Configuration for Cassandra and Postgres

* For Cassandra

Create a CassandraConfig java class and create cluster using following properties.

spring.data.cassandra.keyspace-name=demo

spring.data.cassandra.contact-points=127.0.0.1

spring.data.cassandra.username=cassandra

spring.data.cassandra.password=cassandra

spring.data.cassandra.port=9042

spring.data.cassandra.schema-action=CREATE\_IF\_NOT\_EXISTS

* For Postgres

Create a PostgresConfig java class and Create DataSource, EntityManager and Transaction manager using the following properties.

spring.datasource.url=jdbc:postgresql://localhost:5432/demo

spring.datasource.username=postgres

spring.datasource.password=postgres

hibernate.dialect = org.hibernate.dialect.PostgreSQLDialect

hibernate.hbm2ddl.auto = update

Step4:Design your Cassandra Model in a denormalized form based on Postgres Data Model

Step5:Write API to fetch records from Postgres ,map it to corresponding Cassandra Model and save in cassandra