

- 1) Docker Desktop'u başlatın.
- 2) Proje dizinine giderek docker-compose up komutunu çalıştırın.
- 3) Tüm container'ların başarılı şekilde ayağa kalkmasını bekleyin.
- 4) Ardından Java uygulamasını başlatın.

**Not:** Uygulama Debezium Postgres Connector kullandığı için yalnızca PostgreSQL üzerinde çalışacak şekilde tasarlanmıştır. PostgreSQL, Docker Compose içerisinde zaten hazır olarak kurulup başlatıldığı için ek bir veritabanı kurulumu veya manuel yapılandırma yapmanız gereklidir.

Veritabanına bağlanmak için:

```
docker exec -it <container_id> psql -U postgres kafka-case
```

**Kafka Topicler:**

**kafka\_case.public.package** =Debezium'un PostgreSQL'den yakaladığı tüm ham INSERT/UPDATE/DELETE değişikliklerini yayınladığı CDC topic'i.

**mapped\_package\_cdc** =Java uygulamasının, kafka\_case.public.package topic'indeki kayıtları dönüştürdükten sonra yazdığı çıktı topic'i.

**bootstrap\_mapped\_package** =Tüm package kayıtlarını dönüştürüp toplu şekilde Kafka'ya gönderen REST endpoint:

```
GET http://localhost:8090/kafka/bootstrap
```

**single\_mapped\_packages** =Belirli bir package'ı dönüştürüp Kafka'ya gönderen REST endpoint:

```
GET http://localhost:8090/kafka/send/{id}
```

Uygulamanın consumer tarafı, architecture sorusu için özel olarak hazırlanmıştır. Başlangıçta tüm bağlantılar otomatik olarak kurulmakta olup ek bir işlem yapmanız gereklidir.

Uygulamada kullanılmayan column'lar performans ve sadelik amacıyla filtrelenmiştir.