**모노리식 서비스에서 일부 서비스를 마이크로서비스로 전환**

**모노리스 기반 쇼핑몰 서비스에서 배송 서비스를 분리하고, Feign Client 를 사용해 모노리식 쇼핑몰과 분리된 배송 마이크로서비스 분리하는 Lab 이다.  
Feign Client 는 동기(Request/Response) 방식으로 서비스간의 통신을 가능하면 레가시 코드의 변경을 최소화 하여 트랜스폼하는 방법이다.**

* **monolith 서비스 기동 확인 (8081 port)**
  + **http localhost:8081**
* **Order.java 에서 deliveryService 로컬 객체를 통해 배송처리 중임을 확인:**

**@PostPersist**

**private void callDeliveryStart(){**

**Delivery delivery = new Delivery();**

**delivery.setQuantity(this.getQuantity());**

**delivery.setProductId(this.getProductId());**

**delivery.setProductName(this.getProductName());**

**delivery.setDeliveryAddress(this.getCustomerAddr());**

**delivery.setCustomerId(this.getCustomerId());**

**delivery.setCustomerName(this.getCustomerName());**

**delivery.setDeliveryState(DeliveryStatus.DeliveryStarted.name());**

**delivery.setOrder(this);**

**// 배송 시작**

**DeliveryService deliveryService = Application.applicationContext.getBean(DeliveryService.class);**

**deliveryService.startDelivery(delivery);**

**}**

* **Order.java의 startDelivery 메서드에 디버그 포인트 설치**
* **라인번호(84) 앞을 클릭하면, 빨간색의 원(breakpoint)이 나타남**
* **주문 생성**

**http localhost:8081/orders productId=1 quantity=3 customerId="1@uengine.org" customerName="hong" customerAddr="seoul"**

* **DeliveryServiceImpl.java 를 통해서 배송처리가 되는 Monolith 임을 확인.**

**기존 Monolith 구현체 제거, FeignClient 의 활성화**

* **DeliveryServiceImpl.java 제거**
* **DeliveryService.java 의 주석 제거**

**package com.example.template.delivery;**

**import org.springframework.cloud.openfeign.FeignClient;**

**import org.springframework.web.bind.annotation.RequestMapping;**

**import org.springframework.web.bind.annotation.RequestMethod;**

**@FeignClient(name ="delivery", url="${api.url.delivery}")**

**public interface DeliveryService {**

**@RequestMapping(method = RequestMethod.POST, value = "/deliveries", consumes = "application/json")**

**void startDelivery(Delivery delivery);**

**}**

* **Monolith 서비스를 재기동 한다.**
* **신규 배송 서비스 기동 확인 (8082 port)**
  + **http localhost:8082**
* **주문 생성**

**http localhost:8081/orders productId=1 quantity=3 customerId="1@uengine.org" customerName="hong" customerAddr="seoul"**

* **주문 요청시 배송서비스가 호출되어 배송처리가 원격 마이크로서비스에 의해 처리된 것을 확인**

**http localhost:8082/deliveries**

* **디버거를 통하여 Order.java 의 deliveryService.startDelivery(…) 호출의 deliveryService 객체가 Proxy 객체로 변경된 것을 확인**

**Service Clear**

* **다음 Lab을 위해 기동된 모든 서비스 종료**

**fuser -k 8081/tcp**

**fuser -k 8082/tcp**

**상세설명**

**CheckPoints**

1. 주문이 이루어지고 배송서비스로 데이터가 전달된다

2. 모노리스에는 더이상 배송데이터가 존재하지 않는다.