

## Zipkin server daywise assignment

(pom.xml)

```
<dependencies>
```

```
  <!-- Spring Boot Web -->
```

```
  <dependency>
```

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

```
    <artifactId>spring-boot-starter-web</artifactId>
```

```
  </dependency>
```

```
  <!-- Spring Cloud Sleuth + Zipkin -->
```

```
  <dependency>
```

```
    <groupId>org.springframework.cloud</groupId>
```

```
    <artifactId>spring-cloud-starter-sleuth</artifactId>
```

```
  </dependency>
```

```
  <dependency>
```

```
    <groupId>org.springframework.cloud</groupId>
```

```
    <artifactId>spring-cloud-sleuth-zipkin</artifactId>
```

```
  </dependency>
```

```
  <!-- Feign Client for service calls -->
```

```
  <dependency>
```

```
    <groupId>org.springframework.cloud</groupId>
```

```
    <artifactId>spring-cloud-starter-openfeign</artifactId>
```

```
  </dependency>
```

```
</dependencies>
```

Order Entity (Order.java)

```
public class Order {  
    private String orderId;  
    private String customerId;  
    private double amount;  
    private String status;  
  
}
```

Order Controller (OrderController.java)

```
import org.springframework.web.bind.annotation.*;  
import org.springframework.beans.factory.annotation.Autowired;
```

```
@RestController
```

```
@RequestMapping("/orders")
```

```
public class OrderController {
```

```
    @Autowired
```

```
    private PaymentServiceClient paymentServiceClient;
```

```
    @PostMapping
```

```
    public Order createOrder(@RequestBody Order order) {
```

```
        // Process order logic
```

```
        order.setStatus("CREATED");
```

```
        // Call Payment Service
```

```
        order = paymentServiceClient.processPayment(order);

        return order;
    }
}
```

Feign Client (PaymentServiceClient.java)

```
import org.springframework.cloud.openfeign.FeignClient;
import org.springframework.web.bind.annotation.PostMapping;

@FeignClient(name = "payment-service", url = "${payment.service.url}")
public interface PaymentServiceClient {

    @PostMapping("/payments")
    Order processPayment(Order order);
}
```

Payment Controller (PaymentController.java)

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

@RestController
@RequestMapping("/payments")
public class PaymentController {

    @PostMapping
    public Order processPayment(@RequestBody Order order) {
        // Simulate payment processing
    }
}
```

```

    try {
        Thread.sleep(120); // Simulate processing time
        order.setStatus("PAID");
    } catch (InterruptedException e) {
        order.setStatus("PAYMENT_FAILED");
    }

    return order;
}
}

```

#### Order Service Application Class

```

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.openfeign.EnableFeignClients;

```

```
@SpringBootApplication
```

```
@EnableFeignClients
```

```

public class OrderServiceApplication {
    public static void main(String[] args) {
        SpringApplication.run(OrderServiceApplication.class, args);
    }
}

```

#### Payment Service Application Class

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class PaymentServiceApplication {
```

```
    public static void main(String[] args) {
```

```
        SpringApplication.run(PaymentServiceApplication.class, args);
```

```
    }
```

```
}
```