Maxwell Bruckhaus

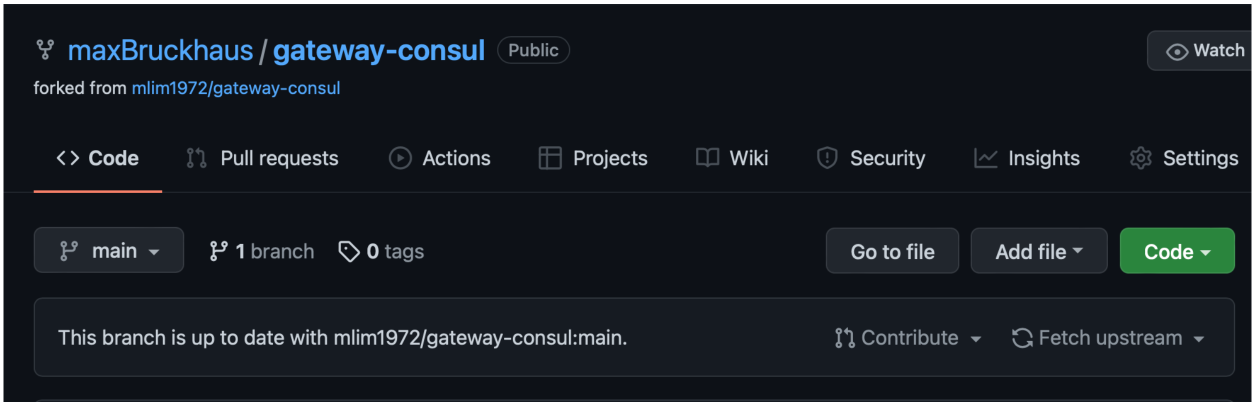
Professor Lim

CS 686-05

April 5th, 2022

## Assignment 13 - Circuit Breaker

1) Fork repo

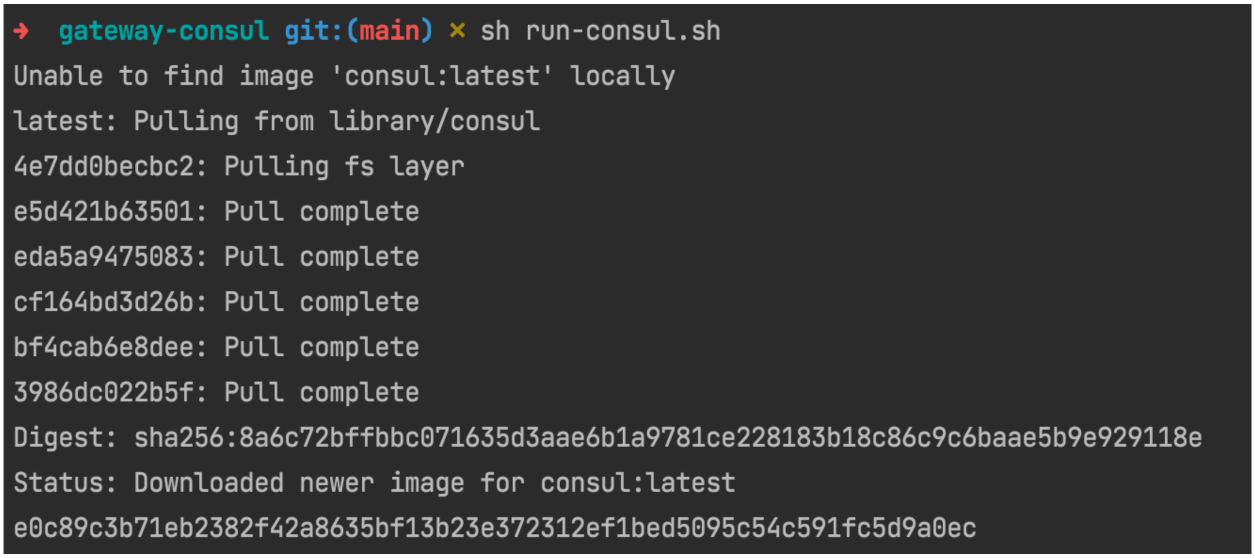


maxBruckhaus / gateway-consul Public

forked from mlim1972/gateway-consul

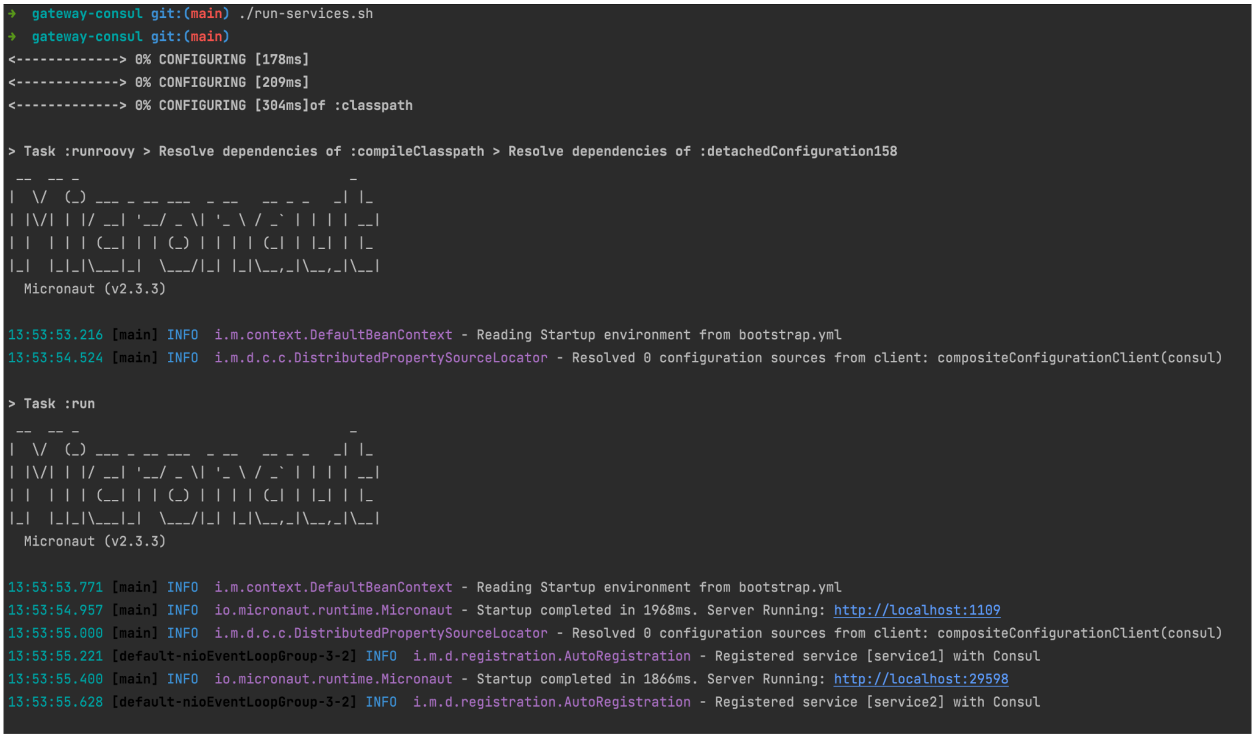
This branch is up to date with mlim1972/gateway-consul:main.

2) Start Consul



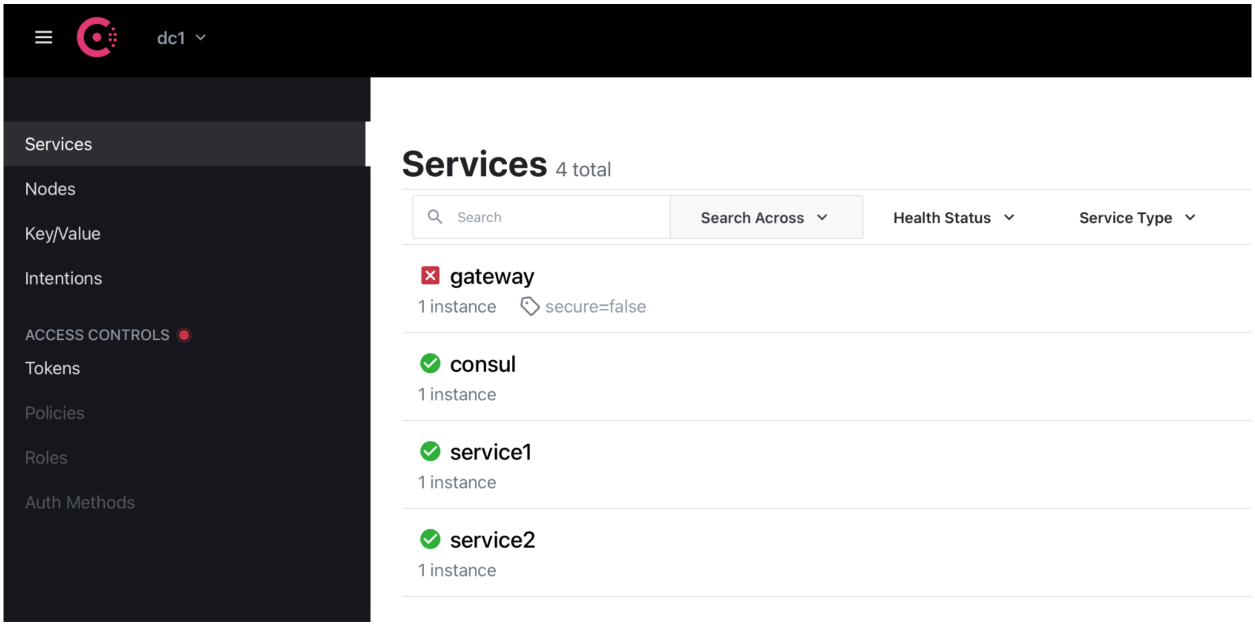
```
➜  gateway-consul git:(main) ✗ sh run-consul.sh
Unable to find image 'consul:latest' locally
latest: Pulling from library/consul
4e7dd0becbc2: Pulling fs layer
e5d421b63501: Pull complete
eda5a9475083: Pull complete
cf164bd3d26b: Pull complete
bf4cab6e8dee: Pull complete
3986dc022b5f: Pull complete
Digest: sha256:8a6c72bffbbc071635d3aae6b1a9781ce228183b18c86c9c6baae5b9e929118e
Status: Downloaded newer image for consul:latest
e0c89c3b71eb2382f42a8635bf13b23e372312ef1bed5095c54c591fc5d9a0ec
```

## 3) Run services



```
➜  gateway-consul git:(main) ./run-services.sh
➜  gateway-consul git:(main)
<-------------> 0% CONFIGURING [178ms]
<-------------> 0% CONFIGURING [209ms]
<-------------> 0% CONFIGURING [304ms]of :classpath

> Task :runroovy > Resolve dependencies of :compileClasspath > Resolve dependencies of :detachedConfiguration158
 __  __ _                                  _
|  \/  (_) ___ _ __ ___  _ __   __ _ _   _| |_
| |\/| | |/ __| '__/ _ \| '_ \ / _` | | | | __|
| |  | | | (__| | | (_) | | | | (_| | |_| | |_
|_|  |_|_|\___|_|  \___/|_| |_|\__,_|\__,_|\__|
  Micronaut (v2.3.3)

13:53:53.216 [main] INFO  i.m.context.DefaultBeanContext - Reading Startup environment from bootstrap.yml
13:53:54.524 [main] INFO  i.m.d.c.c.DistributedPropertySourceLocator - Resolved 0 configuration sources from client: compositeConfigurationClient(consul)

> Task :run
 __  __ _                                  _
|  \/  (_) ___ _ __ ___  _ __   __ _ _   _| |_
| |\/| | |/ __| '__/ _ \| '_ \ / _` | | | | __|
| |  | | | (__| | | (_) | | | | (_| | |_| | |_
|_|  |_|_|\___|_|  \___/|_| |_|\__,_|\__,_|\__|
  Micronaut (v2.3.3)

13:53:53.771 [main] INFO  i.m.context.DefaultBeanContext - Reading Startup environment from bootstrap.yml
13:53:54.957 [main] INFO  io.micronaut.runtime.Micronaut - Startup completed in 1968ms. Server Running: http://localhost:1109
13:53:55.000 [main] INFO  i.m.d.c.c.DistributedPropertySourceLocator - Resolved 0 configuration sources from client: compositeConfigurationClient(consul)
13:53:55.221 [default-nioEventLoopGroup-3-2] INFO  i.m.d.registration.AutoRegistration - Registered service [service1] with Consul
13:53:55.400 [main] INFO  io.micronaut.runtime.Micronaut - Startup completed in 1866ms. Server Running: http://localhost:29598
13:53:55.628 [default-nioEventLoopGroup-3-2] INFO  i.m.d.registration.AutoRegistration - Registered service [service2] with Consul
```

## 4) Consul UI



Services 4 total

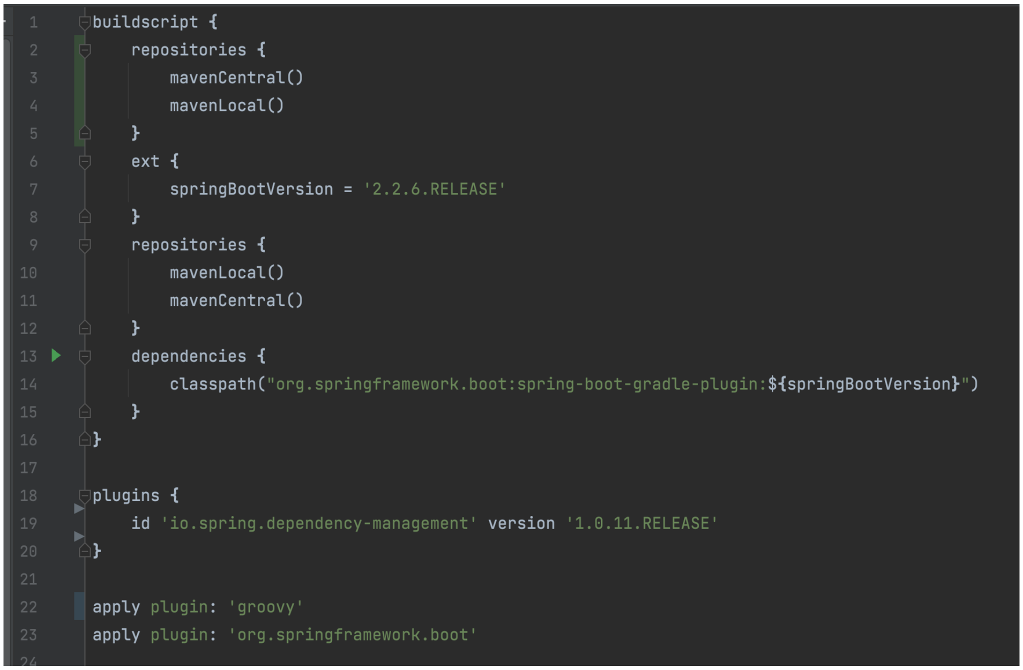
☒ gateway
1 instance secure=false

✓ consul
1 instance

✓ service1
1 instance

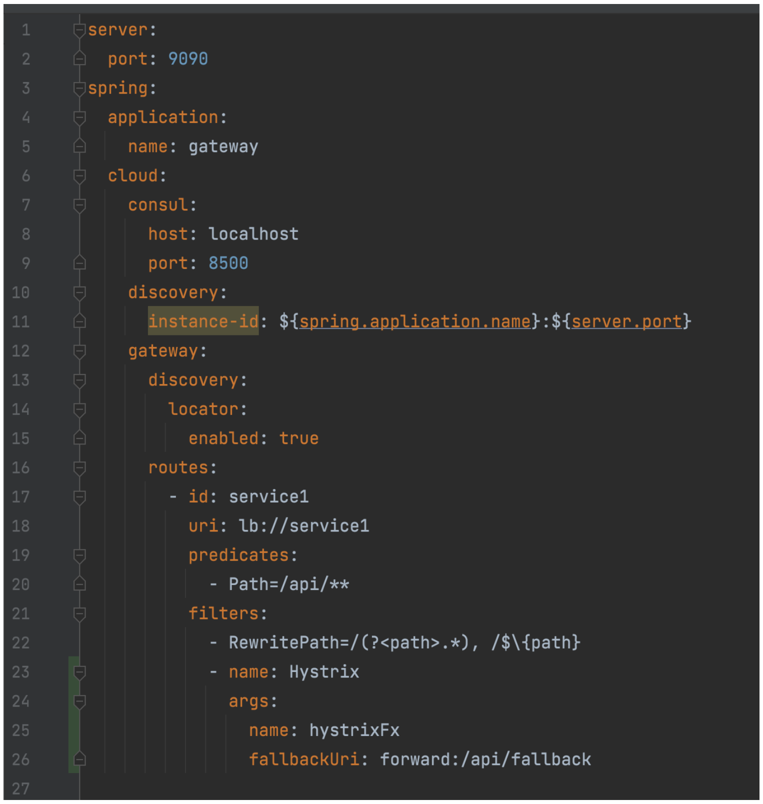
✓ service2
1 instance

5) Update build.gradle script



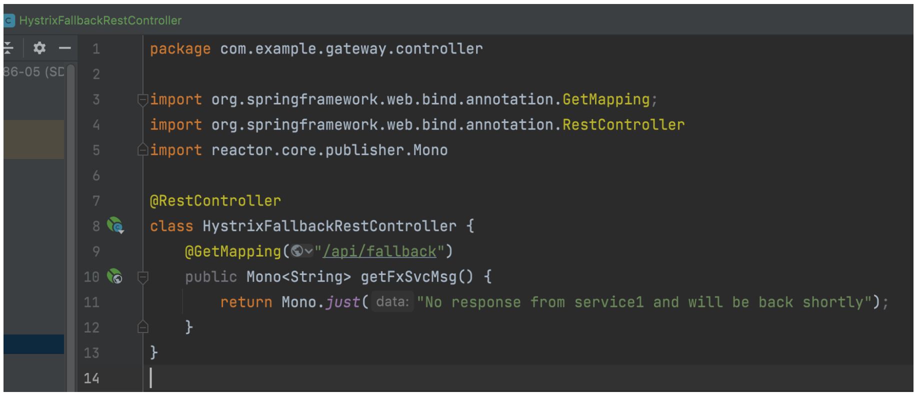
```
buildscript {
    repositories {
        mavenCentral()
        mavenLocal()
    }
    ext {
        springBootVersion = '2.2.6.RELEASE'
    }
    repositories {
        mavenLocal()
        mavenCentral()
    }
    dependencies {
        classpath("org.springframework.boot:spring-boot-gradle-plugin:${springBootVersion}")
    }
}

plugins {
    id 'io.spring.dependency-management' version '1.0.11.RELEASE'
}

apply plugin: 'groovy'
apply plugin: 'org.springframework.boot'
```

6) Update application.yaml



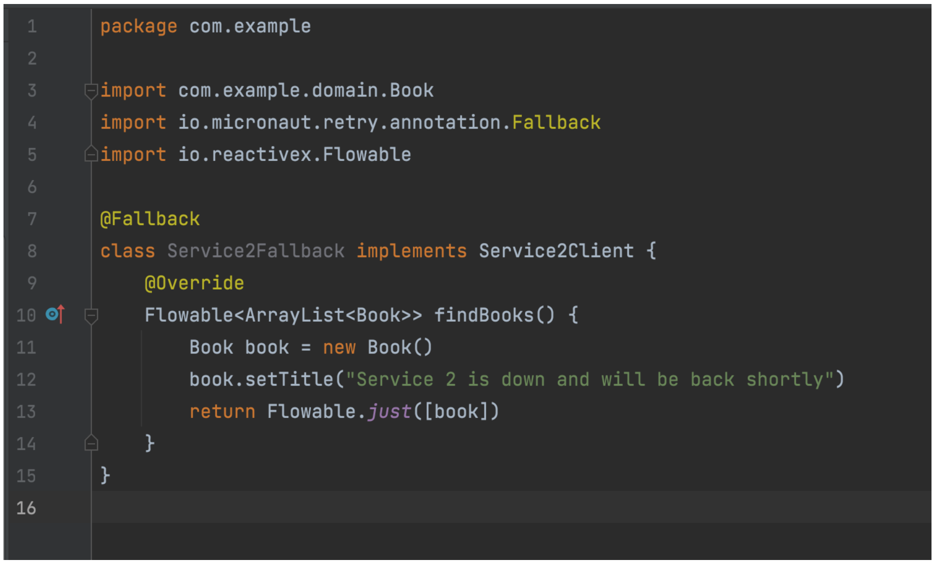
```
server:
  port: 9090
spring:
  application:
    name: gateway
  cloud:
    consul:
      host: localhost
      port: 8500
    discovery:
      instance-id: ${spring.application.name}:${server.port}
    gateway:
      discovery:
        locator:
          enabled: true
      routes:
        - id: service1
          uri: lb://service1
          predicates:
            - Path=/api/**
          filters:
            - RewritePath=/(?<path>.*), /$\{path}
            - name: Hystrix
              args:
                name: hystrixFx
                fallbackUri: forward:/api/fallback
```

7) Create fallback controller



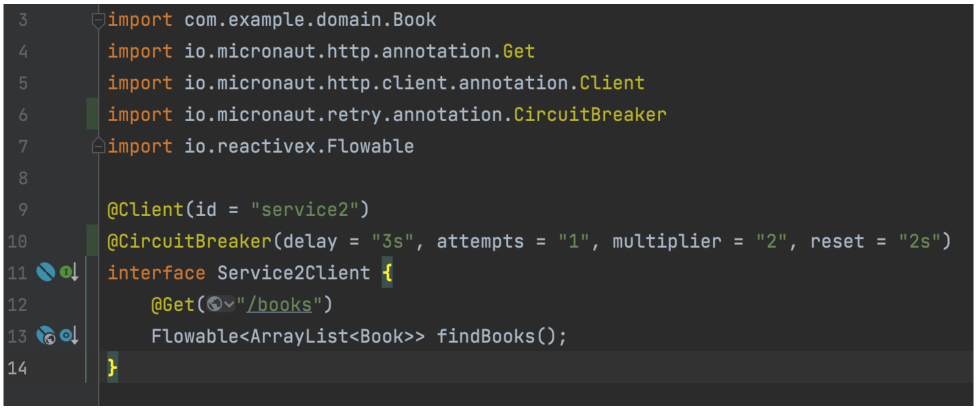
```
package com.example.gateway.controller

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController
import reactor.core.publisher.Mono

@RestController
class HystrixFallbackRestController {
    @GetMapping("/api/fallback")
    public Mono<String> getFxSvcMsg() {
        return Mono.just("No response from service1 and will be back shortly");
    }
}
```

8) Create service2 fallback



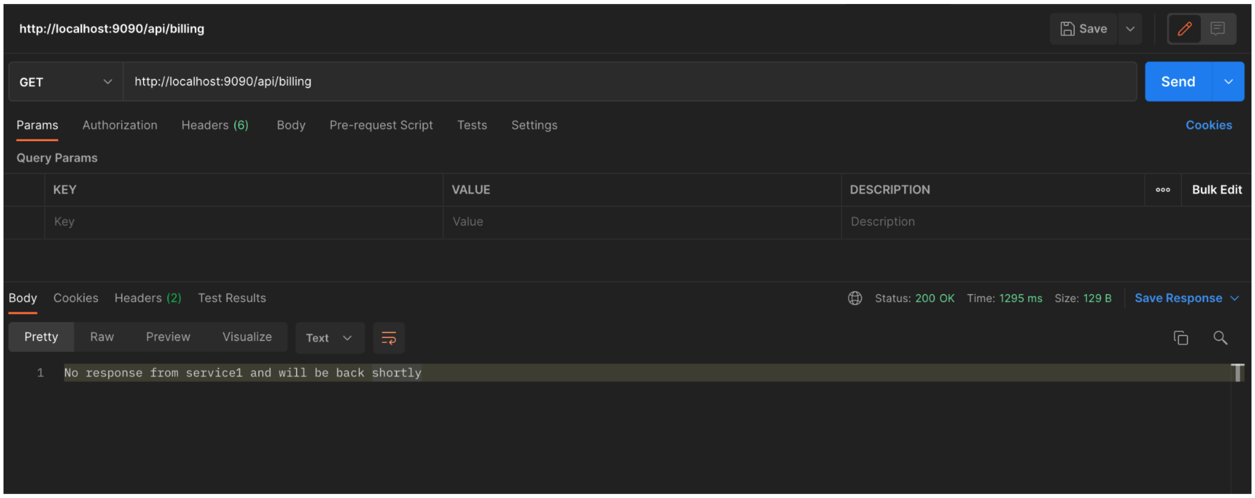
```
package com.example

import com.example.domain.Book
import io.micronaut.retry.annotation.Fallback
import io.reactivex.Flowable

@Fallback
class Service2Fallback implements Service2Client {
    @Override
    Flowable<ArrayList<Book>> findBooks() {
        Book book = new Book()
        book.setTitle("Service 2 is down and will be back shortly")
        return Flowable.just([book])
    }
}
```

9) Create circuit breaker



```
import com.example.domain.Book
import io.micronaut.http.annotation.Get
import io.micronaut.http.client.annotation.Client
import io.micronaut.retry.annotation.CircuitBreaker
import io.reactivex.Flowable

@Client(id = "service2")
@CircuitBreaker(delay = "3s", attempts = "1", multiplier = "2", reset = "2s")
interface Service2Client {
    @Get("/books")
    Flowable<ArrayList<Book>> findBooks();
}
```

10) No response from service1



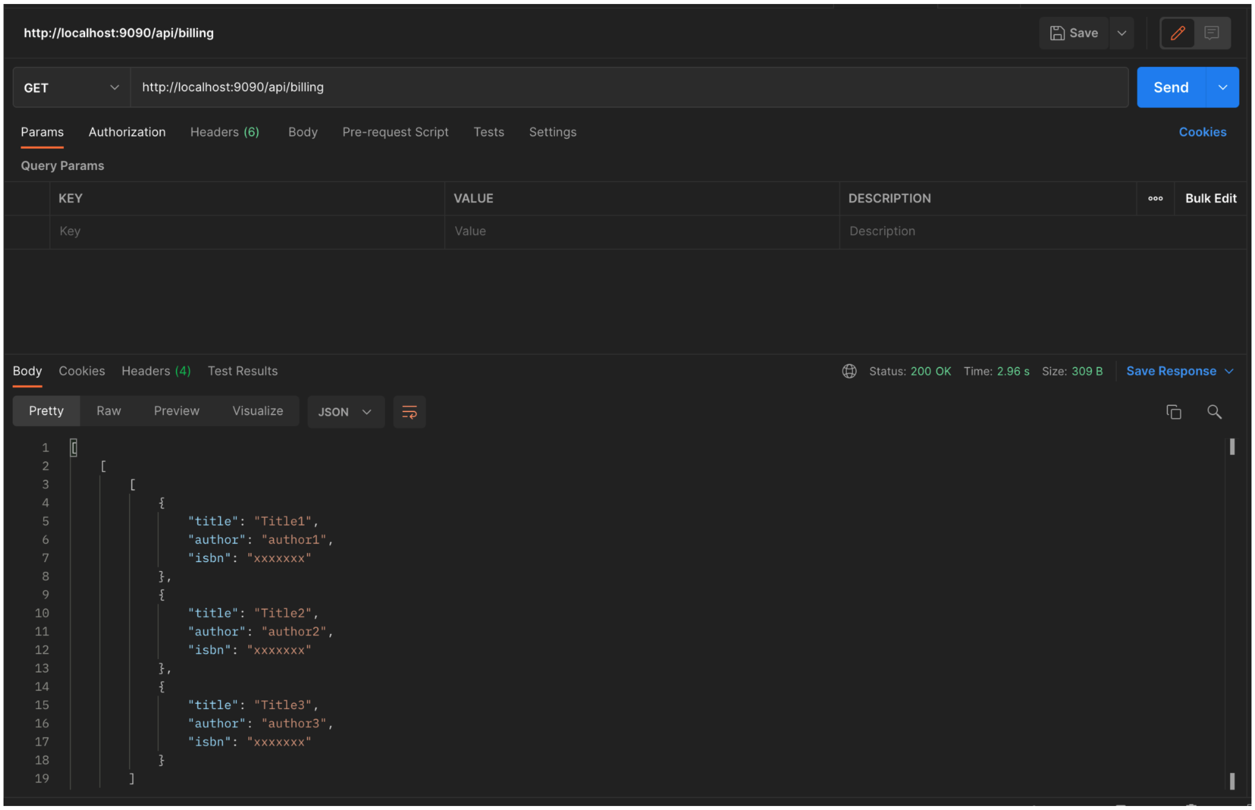
http://localhost:9090/api/billing

GET http://localhost:9090/api/billing

Status: 200 OK Time: 1295 ms Size: 129 B

```
No response from service1 and will be back shortly
```

11) Successful GET with books



http://localhost:9090/api/billing

GET http://localhost:9090/api/billing

Status: 200 OK Time: 2.96 s Size: 309 B

```
[
    [
        [
            {
                "title": "Title1",
                "author": "author1",
                "isbn": "xxxxxxx"
            },
            {
                "title": "Title2",
                "author": "author2",
                "isbn": "xxxxxxx"
            },
            {
                "title": "Title3",
                "author": "author3",
                "isbn": "xxxxxxx"
            }
        ]
```