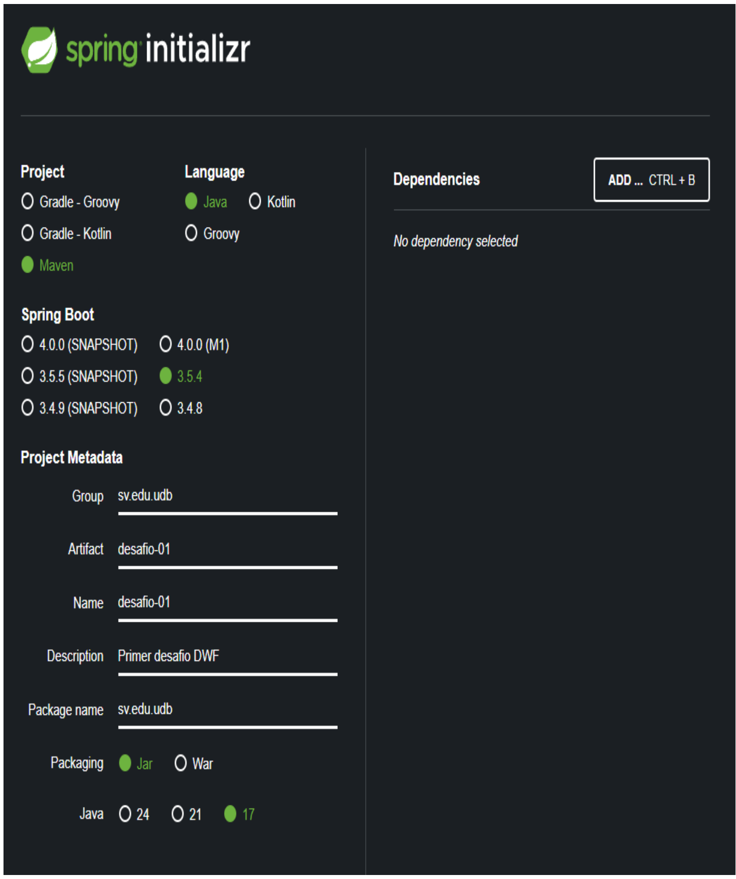
Bryan Steven Hernandez Polio ---- HP240512 ------ Desafio01_DWF

Creación de proyecto / Estructura del mismo:



spring initializr

Project

○ Gradle - Groovy
○ Gradle - Kotlin
● Maven

Language

● Java ○ Kotlin
○ Groovy

Spring Boot

○ 4.0.0 (SNAPSHOT) ○ 4.0.0 (M1)
○ 3.5.5 (SNAPSHOT) ● 3.5.4
○ 3.4.9 (SNAPSHOT) ○ 3.4.8

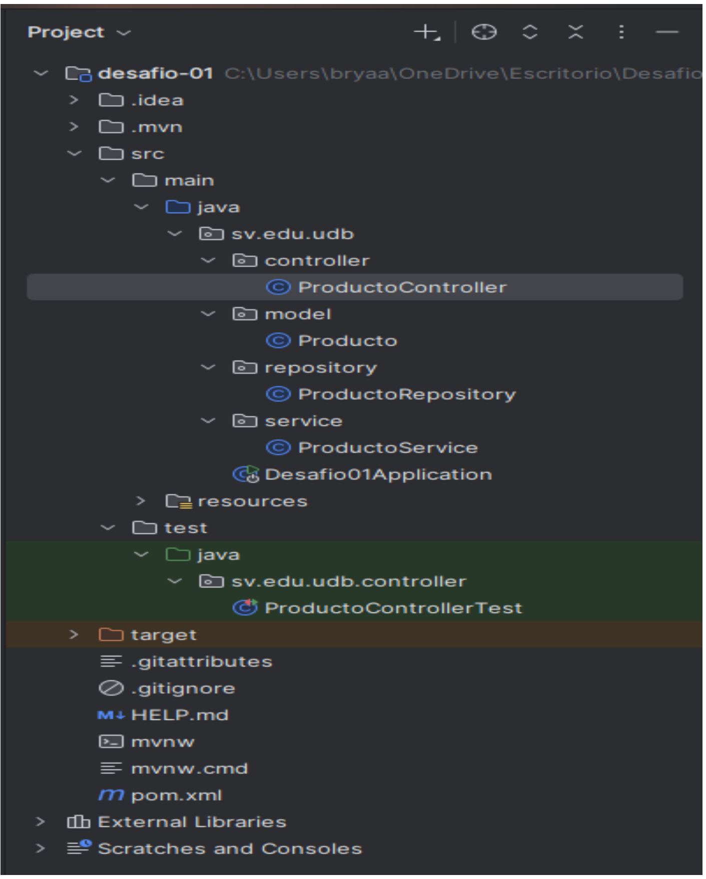
Project Metadata

Group: sv.edu.udb
Artifact: desafio-01
Name: desafio-01
Description: Primer desafio DWF
Package name: sv.edu.udb
Packaging: ● Jar ○ War
Java: ○ 24 ○ 21 ● 17

Dependencies

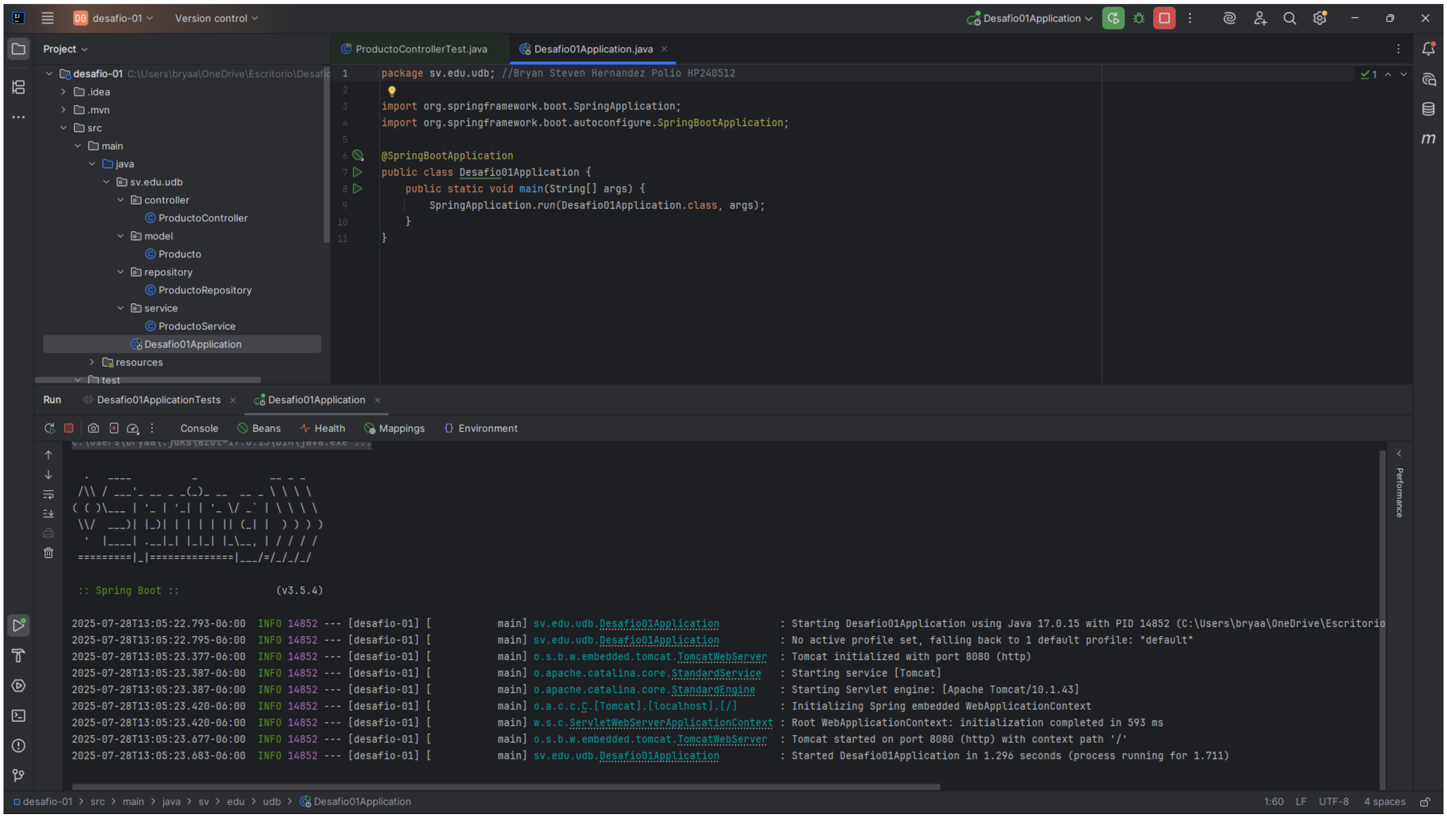
ADD ... CTRL + B

No dependency selected



```
Project
desafio-01  C:\Users\bryaa\OneDrive\Escritorio\Desafio
  .idea
  .mvn
  src
    main
      java
        sv.edu.udb
          controller
            ProductoController
          model
            Producto
          repository
            ProductoRepository
          service
            ProductoService
          Desafio01Application
      resources
    test
      java
        sv.edu.udb.controller
          ProductoControllerTest
  target
  .gitattributes
  .gitignore
  HELP.md
  mvnw
  mvnw.cmd
  pom.xml
External Libraries
Scratches and Consoles
```

Ejecución de la Aplicación ya terminada, por medio de Desafio01Application:

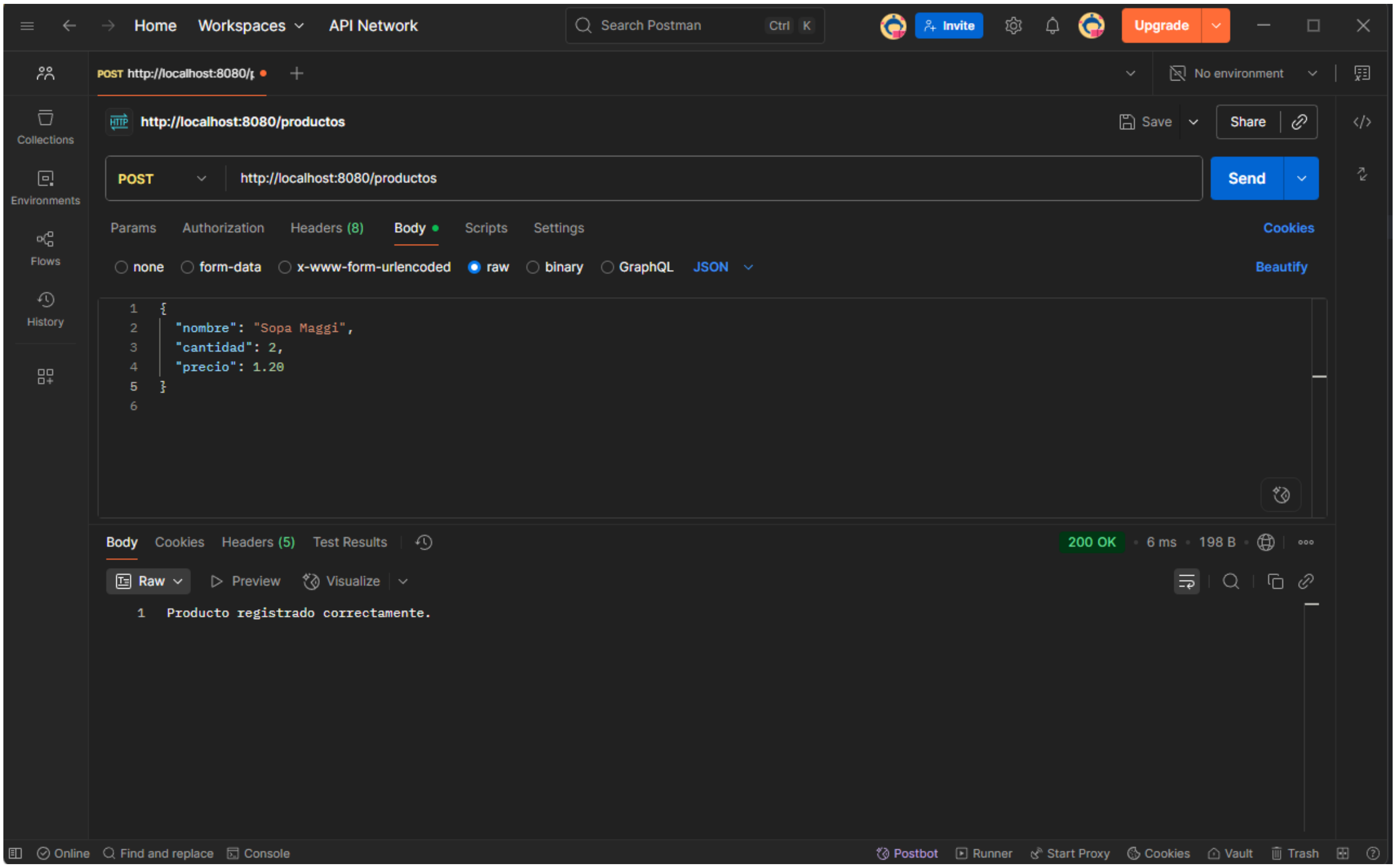


desafio-01
Version control
Desafio01Application
Project
ProductoControllerTest.java
Desafio01Application.java
Run
Desafio01ApplicationTests
Desafio01Application
Console
Beans
Health
Mappings
Environment
Performance
desafio-01 > src > main > java > sv > edu > udb > Desafio01Application
1:60 LF UTF-8 4 spaces

```java
package sv.edu.udb; //Bryan Steven Hernandez Polio HP240512

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

@SpringBootApplication
public class Desafio01Application {
    public static void main(String[] args) {
        SpringApplication.run(Desafio01Application.class, args);
    }
}
```

```
C:\Users\bryaa\.jdks\azul-17.0.15\bin\java.exe ...

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.5.4)

2025-07-28T13:05:22.793-06:00  INFO 14852 --- [desafio-01] [           main] sv.edu.udb.Desafio01Application          : Starting Desafio01Application using Java 17.0.15 with PID 14852 (C:\Users\bryaa\OneDrive\Escritorio
2025-07-28T13:05:22.795-06:00  INFO 14852 --- [desafio-01] [           main] sv.edu.udb.Desafio01Application          : No active profile set, falling back to 1 default profile: "default"
2025-07-28T13:05:23.377-06:00  INFO 14852 --- [desafio-01] [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (http)
2025-07-28T13:05:23.387-06:00  INFO 14852 --- [desafio-01] [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-07-28T13:05:23.387-06:00  INFO 14852 --- [desafio-01] [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.43]
2025-07-28T13:05:23.420-06:00  INFO 14852 --- [desafio-01] [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2025-07-28T13:05:23.420-06:00  INFO 14852 --- [desafio-01] [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 593 ms
2025-07-28T13:05:23.677-06:00  INFO 14852 --- [desafio-01] [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8080 (http) with context path '/'
2025-07-28T13:05:23.683-06:00  INFO 14852 --- [desafio-01] [           main] sv.edu.udb.Desafio01Application          : Started Desafio01Application in 1.296 seconds (process running for 1.711)
```

Prueba de POST:



POST http://localhost:8080/productos

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

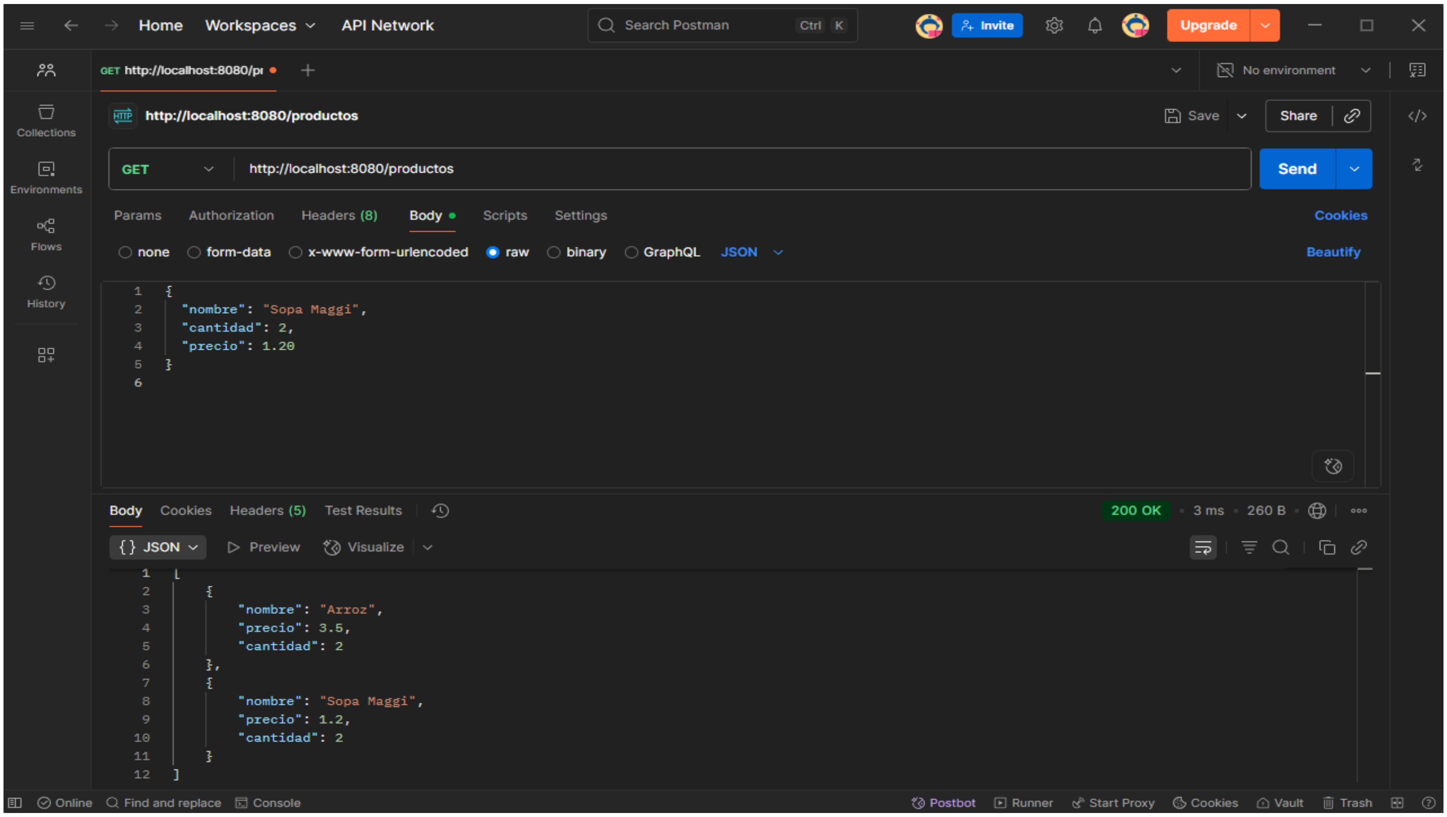
```
{
  "nombre": "Sopa Maggi",
  "cantidad": 2,
  "precio": 1.20
}

```

200 OK · 6 ms · 198 B

Raw

```
Producto registrado correctamente.
```

Prueba de GET:



GET http://localhost:8080/productos

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
  "nombre": "Sopa Maggi",
  "cantidad": 2,
  "precio": 1.20
}

```

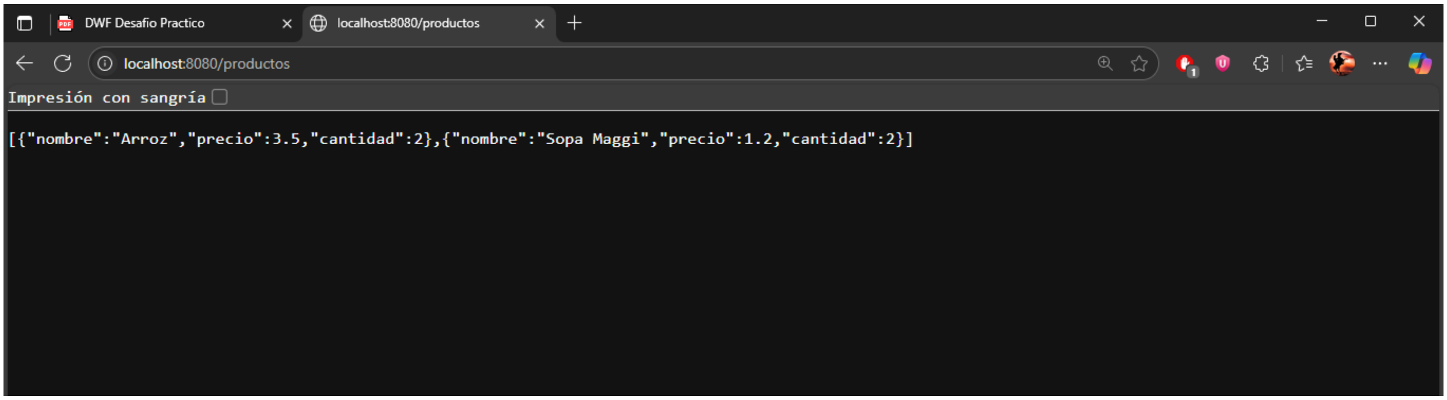
200 OK · 3 ms · 260 B

{ } JSON

```
[
    {
        "nombre": "Arroz",
        "precio": 3.5,
        "cantidad": 2
    },
    {
        "nombre": "Sopa Maggi",
        "precio": 1.2,
        "cantidad": 2
    }
]
```

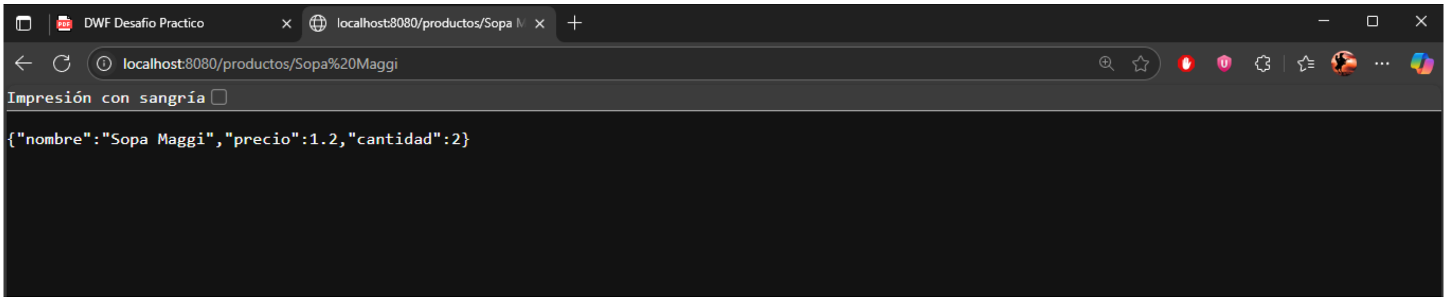
Pruebas en navegador para ver los registros anteriores de Postman:

Productos en general, registrados anteriormente:



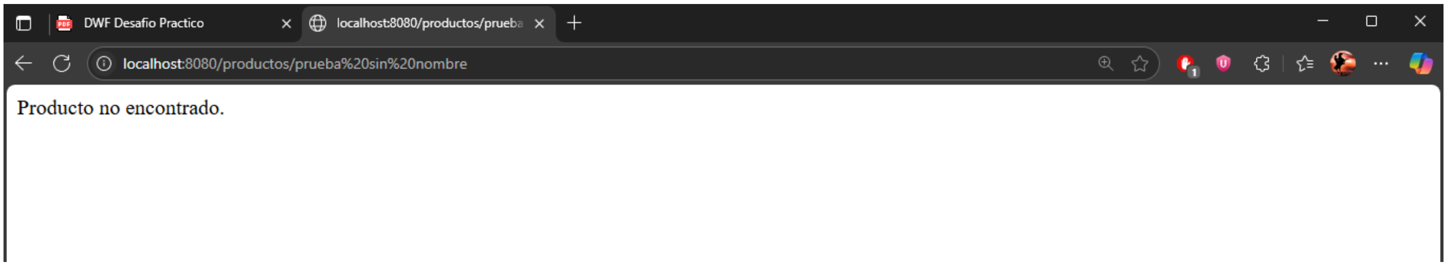
```
[{"nombre":"Arroz","precio":3.5,"cantidad":2},{"nombre":"Sopa Maggi","precio":1.2,"cantidad":2}]
```

Buscando el producto por nombre en específico registrado:

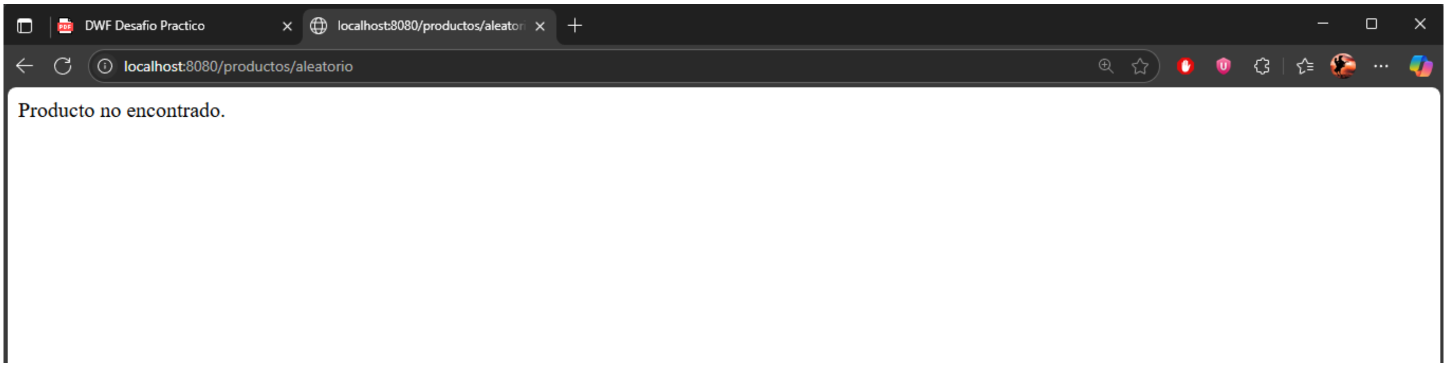

DWF Desafio Practico
localhost:8080/productos/Sopa M
localhost:8080/productos/Sopa%20Maggi
Impresión con sangría □

```
{"nombre":"Sopa Maggi","precio":1.2,"cantidad":2}
```

Pruebas por medio de nombres aleatorios o que no fueron registrados nos darán el siguiente resultado:

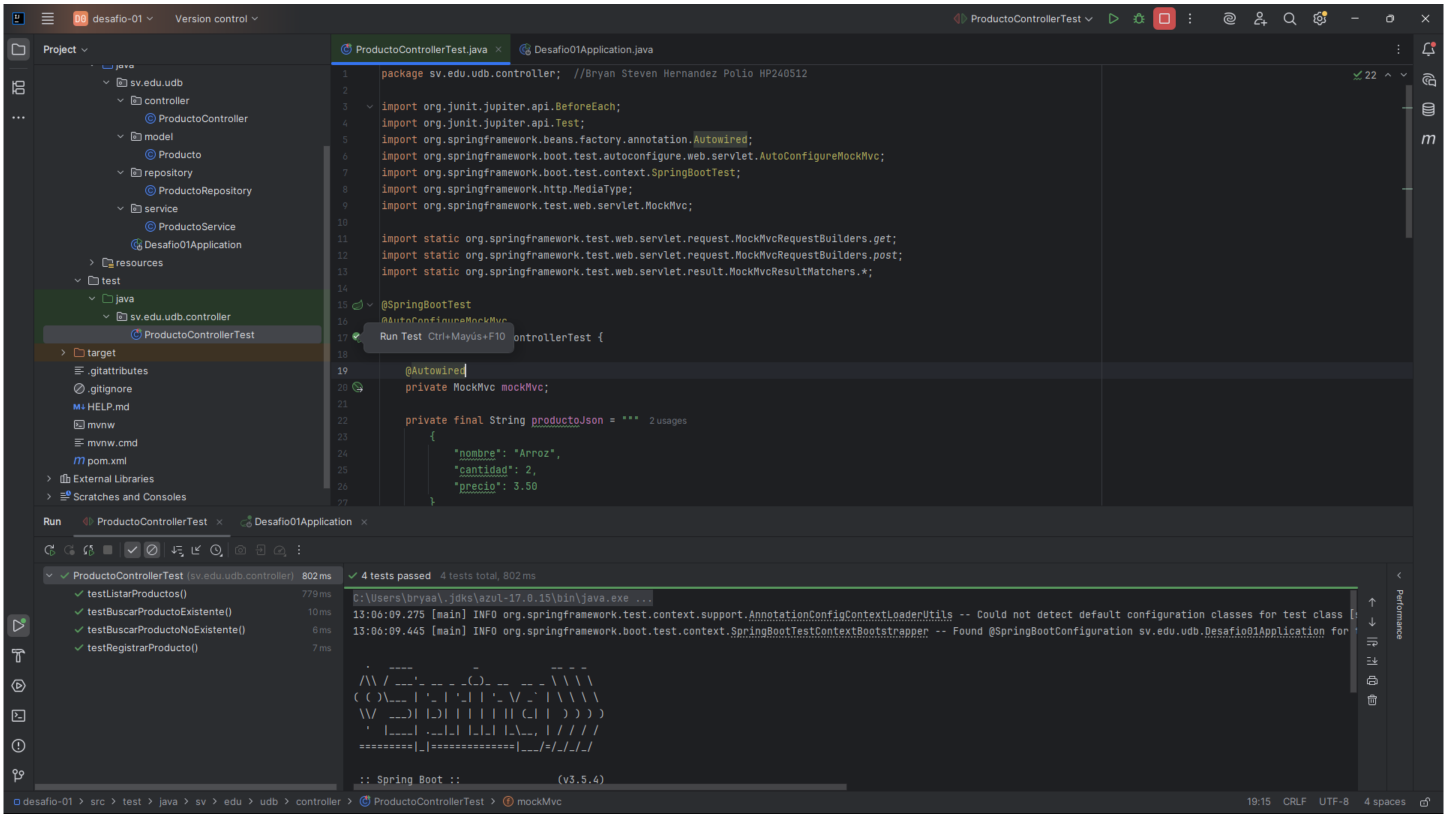

DWF Desafio Practico
localhost:8080/productos/prueba
localhost:8080/productos/prueba%20sin%20nombre

```
Producto no encontrado.
```


DWF Desafio Practico
localhost:8080/productos/aleator
localhost:8080/productos/aleatorio

```
Producto no encontrado.
```

Pruebas unitarias ejecutadas, para ver el funcionamiento correcto de Post / Get en ProductoControllerTest:



```
package sv.edu.udb.controller;  //Bryan Steven Hernandez Polio HP240512

import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.http.MediaType;
import org.springframework.test.web.servlet.MockMvc;

import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.get;
import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.post;
import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*;

@SpringBootTest
@AutoConfigureMockMvc
                      ontrollerTest {

    @Autowired
    private MockMvc mockMvc;

    private final String productoJson = """
        {
            "nombre": "Arroz",
            "cantidad": 2,
            "precio": 3.50
        }
```

```
✓ ProductoControllerTest (sv.edu.udb.controller) 802 ms
    ✓ testListarProductos()              779 ms
    ✓ testBuscarProductoExistente()       10 ms
    ✓ testBuscarProductoNoExistente()      6 ms
    ✓ testRegistrarProducto()              7 ms
```

```
✓ 4 tests passed  4 tests total, 802 ms
C:\Users\bryaa\.jdks\azul-17.0.15\bin\java.exe ...
13:06:09.275 [main] INFO org.springframework.test.context.support.AnnotationConfigContextLoaderUtils -- Could not detect default configuration classes for test class [
13:06:09.445 [main] INFO org.springframework.boot.test.context.SpringBootTestContextBootstrapper -- Found @SpringBootConfiguration sv.edu.udb.Desafio01Application for

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.5.4)
```