

EJERCICIO 2

La página <https://petstore.swagger.io/> proporciona la documentación sobre apis de una "PetStore". Utilizando un software para pruebas de servicios REST realizar las siguientes pruebas, identificando las entradas, capturando las salidas, test, variables, etc, en cada uno de los siguientes casos:

- Configuración a nivel de cabecera

```
Feature: PetStore
  Background: Cabecera
    #Definimos url base
    Given url 'https://petstore.swagger.io/v2'
```

- Añadir una mascota a la tienda

```
@test
Scenario: Añadir una mascota a la tienda.
  #Identificamos el path
  Given path '/pet'
  And request
  """
  {
    "id": '500',
    "category": {
      "id": 2,
      "name": "Animal"
    },
    "name": "Drako",
    "photoUrls": [
      "string"
    ],
    "tags": [
      {
        "id": 10,
        "name": "Canino"
      }
    ],
    "status": "available"
  }
  """
  When method post
  Then status 200
```

- Consultar la mascota ingresada previamente (Búsqueda por ID)

```
#Siguiendo la página del "https://petstore.swagger.io/v2" identificamos método a consumir
#(GET) para la búsqueda de la mascota con ID, referimos el id del método anterior para
# realizar la consulta correspondiente y visualizar que la mascota fue agregada correctamente.
@test
Scenario: Consultar la mascota ingresada previamente Búsqueda por ID.
  Given path '/pet/500'
  When method get
  Then status 200
```

- Actualizar el nombre de la mascota y el estatus de la mascota a "sold"

```
@test
Scenario: Actualizar el nombre de la mascota y el estatus de la mascota a sold.
  #Identificamos el path
  Given path '/pet'
  And request
  """
  {
    "id": 500,
    "category": {
      "id": 2,
      "name": "Animal"
    },
    "name": "Lalo",
    "photoUrls": [
      "string"
    ],
    "tags": [
      {
        "id": 10,
        "name": "Canino"
      }
    ],
    "status": "sold"
  }
  """
  When method put
  Then status 200
```

- Consultar la mascota modificada por estatus (Búsqueda por estatus)

```
#Siguiendo la página del "https://petstore.swagger.io/v2" identificamos método a consumir
#(GET) para la búsqueda de la mascota por status=sold.
# realizar la consulta correspondiente y visualizar que la mascota fue actualizada correctamente.
@test
Scenario: Consultar la mascota modificada por estatus Búsqueda por estatus.
  #Url completa por carácter especial "?"
  Given url 'https://petstore.swagger.io/v2/pet/findByStatus?status=sold'
  When method get
  Then status 200
```

- Se creo una clase KarateRunner para ejecutar el feature y todos los escenarios implementados.

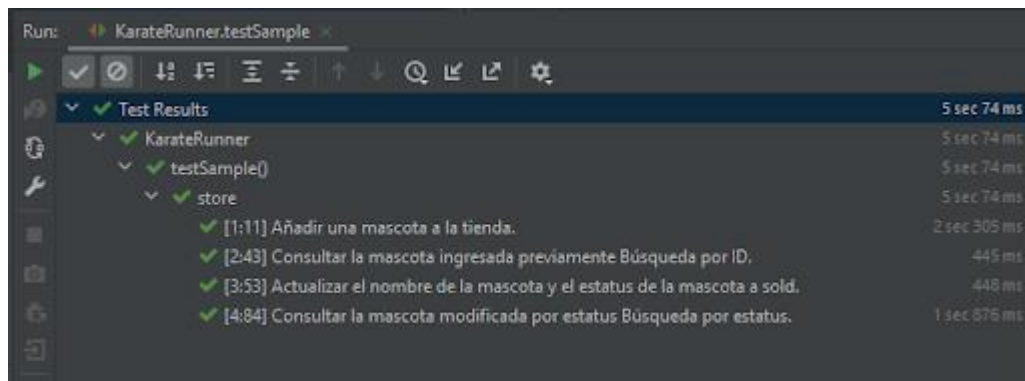


```

1  import com.intuit.karate.junit5.Karate;
2
3
4  public class KarateRunner {
5      @Karate.Test
6      Karate testSample() {
7          return Karate.run("classpath:store.feature");
8      }
9  }
10

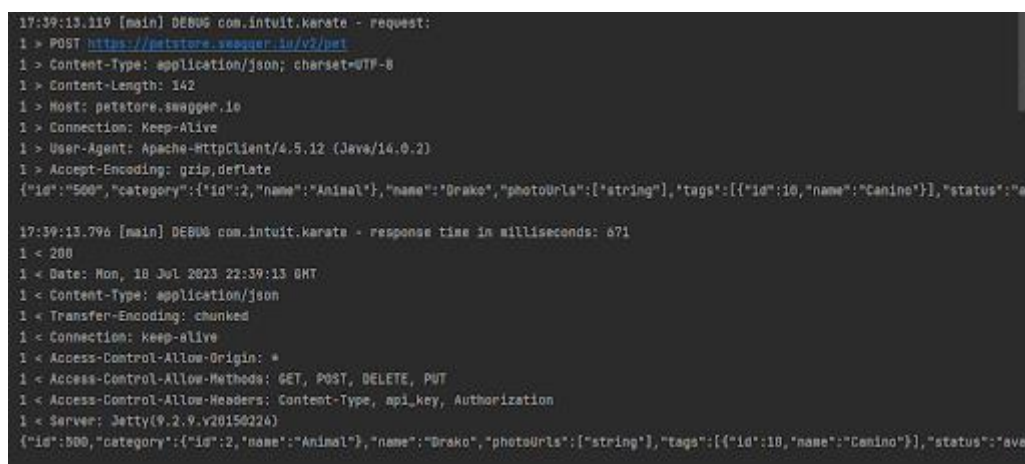
```

- Se muestra la ejecución de la clase java KarateRunner, con los casos mencionados con respuesta exitosa como se muestra en la imagen.



Test Results	Time
KarateRunner	5 sec 74 ms
testSample()	5 sec 74 ms
store	5 sec 74 ms
✓ [1:11] Añadir una mascota a la tienda.	2 sec 305 ms
✓ [2:43] Consultar la mascota ingresada previamente Búsqueda por ID.	445 ms
✓ [3:53] Actualizar el nombre de la mascota y el estatus de la mascota a sold.	448 ms
✓ [4:84] Consultar la mascota modificada por estatus Búsqueda por estatus.	1 sec 876 ms

- Del caso 1 se envia el request con el método POST configurando la cabecera los elementos: Content-Type, Content-Length y Accept-Encoding.
- Se envía como request los campos ID: 500 y name: "Drako".
- Se obtiene como respuesta el código 200 de OK y los datos del animal registrado.



```

17:39:13.119 [main] DEBUG com.intuit.karate - request:
1 > POST https://petstore.swagger.io/v2/pet
1 > Content-Type: application/json; charset=UTF-8
1 > Content-Length: 142
1 > Host: petstore.swagger.io
1 > Connection: Keep-Alive
1 > User-Agent: Apache-HttpClient/4.5.12 (Java/14.0.2)
1 > Accept-Encoding: gzip,deflate
{"id":500,"category":{"id":2,"name":"Animal"},"name":"Drako","photoUrls":["string"],"tags":[{"id":10,"name":"Canine"}],"status":"available"}

17:39:13.796 [main] DEBUG com.intuit.karate - response time in milliseconds: 671
1 < 200
1 < Date: Mon, 10 Jul 2023 22:39:13 GMT
1 < Content-Type: application/json
1 < Transfer-Encoding: chunked
1 < Connection: keep-alive
1 < Access-Control-Allow-Origin: *
1 < Access-Control-Allow-Methods: GET, POST, DELETE, PUT
1 < Access-Control-Allow-Headers: Content-Type, api_key, Authorization
1 < Server: Jetty(9.2.9.v20150226)
{"id":500,"category":{"id":2,"name":"Animal"},"name":"Drako","photoUrls":["string"],"tags":[{"id":10,"name":"Canine"}],"status":"available"}

```

- Del caso 2 se realiza la consulta con el método GET enviando la URL de la mascota registrada con el ID: 500.
- Se obtiene como respuesta el código 200 de OK y los datos del animal consultado con el ID 500, que están asociados con el Name: “Drako”.

```
17:39:13.818 [main] DEBUG com.intuit.karate - request:
1 > GET https://petstore.swagger.io/v2/pet/500
1 > Host: petstore.swagger.io
1 > Connection: Keep-Alive
1 > User-Agent: Apache-HttpClient/4.5.12 (Java/14.0.2)
1 > Accept-Encoding: gzip,deflate

17:39:14.248 [main] DEBUG com.intuit.karate - response time in milliseconds: 429
1 < 200
1 < Date: Mon, 18 Jul 2023 22:39:14 GMT
1 < Content-Type: application/json
1 < Transfer-Encoding: chunked
1 < Connection: keep-alive
1 < Access-Control-Allow-Origin: *
1 < Access-Control-Allow-Methods: GET, POST, DELETE, PUT
1 < Access-Control-Allow-Headers: Content-Type, api_key, Authorization
1 < Server: Jetty(9.2.9.v20150224)
{"id":500,"category":{"id":2,"name":"Animal"},"name":"Drako","photoUrls":["string"],"tags":[{"id":18,"name":"Canino"}],"status":"available"}
```

- Del caso 3 se realiza la actualización con el método PUT enviando la URL de la consulta, generando el request con el ID: 500 y en el campo Name: “Lalo” que anteriormente se registró como Name: “Drako”.
- Se obtiene como respuesta el código 200 de OK y los datos del animal actualizado con el ID: 500 y el Name: “Lalo”.

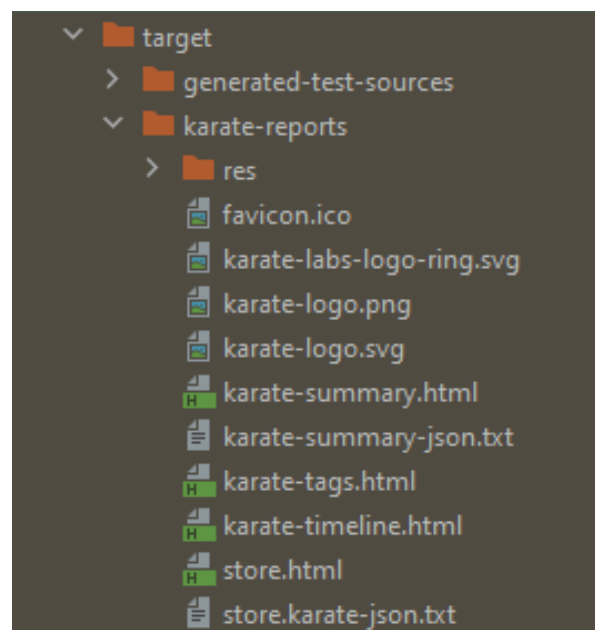
```
17:39:14.262 [main] DEBUG com.intuit.karate - request:
1 > PUT https://petstore.swagger.io/v2/pet
1 > Content-Type: application/json; charset=UTF-8
1 > Content-Length: 134
1 > Host: petstore.swagger.io
1 > Connection: Keep-Alive
1 > User-Agent: Apache-HttpClient/4.5.12 (Java/14.0.2)
1 > Accept-Encoding: gzip,deflate
{"id":500,"category":{"id":2,"name":"Animal"},"name":"Lalo","photoUrls":["string"],"tags":[{"id":18,"name":"Canino"}],"status":"sold"}

17:39:14.701 [main] DEBUG com.intuit.karate - response time in milliseconds: 438
1 < 200
1 < Date: Mon, 18 Jul 2023 22:39:14 GMT
1 < Content-Type: application/json
1 < Transfer-Encoding: chunked
1 < Connection: keep-alive
1 < Access-Control-Allow-Origin: *
1 < Access-Control-Allow-Methods: GET, POST, DELETE, PUT
1 < Access-Control-Allow-Headers: Content-Type, api_key, Authorization
1 < Server: Jetty(9.2.9.v20150224)
{"id":500,"category":{"id":2,"name":"Animal"},"name":"Lalo","photoUrls":["string"],"tags":[{"id":18,"name":"Canino"}],"status":"sold"}
```

- Del caso 4 se realiza la consulta con el método GET enviando la URL, generando el request con el status sold.
- Se obtiene como respuesta el código 200 de OK y los datos de los animales actualizados.

```
17:50:11.611 [main] DEBUG com.intuit.karate - request:
1 > GET https://petstore.swagger.io/v2/pet/findByStatus?status=sold
1 > Host: petstore.swagger.io
1 > Connection: Keep-Alive
1 > User-Agent: Apache-HttpClient/4.5.12 (Java/14.0.2)
1 > Accept-Encoding: gzip,deflate

17:50:12.241 [main] DEBUG com.intuit.karate - response time in milliseconds: 629
1 < 200
1 < Date: Mon, 10 Jul 2023 22:50:12 GMT
1 < Content-Type: application/json
1 < Transfer-Encoding: chunked
1 < Connection: keep-alive
1 < Access-Control-Allow-Origin: *
1 < Access-Control-Allow-Methods: GET, POST, DELETE, PUT
1 < Access-Control-Allow-Headers: Content-Type, api_key, Authorization
1 < Server: Jetty(9.2.9.v20150224)
[{"id":300,"category":{"id":0,"name":"Animal"},"name":"Lulu","photoUrls":["string"],"tags":[{"id":0,"name":"perro"}],
"status":"sold"}, {"id":77194,"category":{"id":444,"name":"Squirrel"},"name":"Duke","photoUrls":["http://test
.com/photo4.jpg"],"tags":[{"id":77194,"name":"Tag Duke"}],"status":"sold"}, {"id":24093935,"category":{"id":-6714225,
"name":"do magna"},"name":"doggie","photoUrls":["aute","velit"],"tags":[{"id":71561130,"name":"deserunt ut
occaecat"}, {"id":-1195842,"name":"nulla nostrud occaecat"}],"status":"sold"}, {"id":20522,"category":{"id":444,
"name":"Squirrel"},"name":"Duke","photoUrls":["http://test.com/photo4.jpg"],"tags":[{"id":20522,"name":"Tag
Duke"}],"status":"sold"}]
```



- Reporte de los métodos ejecutados con Karate a nivel web (Chrome)

1.-

Summary Top Feature: show/feature PetStore			
<div> </div> <div> Scenarios 2023-07-10 22:19:13 p.m. [111] Añadir una mascota a la tienda. [141] Consultar la mascota ingresada previamente buscando por ID. [151] Actualizar el nombre de la mascota y el estatus de la mascota a 200. [164] Consultar la mascota modificada por estatus buscando por estatus. </div>	Scenario: [111] Añadir una mascota a la tienda.		ms: 1298
	Background:		
	#Identificamos el path		
	13	Given path '/pet'	1
Scenario: [241] Consultar la mascota ingresada previamente buscando por ID.	Background:		
	#Identificamos el path		
	44	Given path '/pet/500'	1
	45	When method get	1059
Scenario: [241] Consultar la mascota ingresada previamente buscando por ID.	Then status 200		0
	Background:		
	#Identificamos el path		
	55	Given path '/pet'	1
Scenario: [241] Actualizar el nombre de la mascota y el estatus de la mascota a 200.	Background:		
	#Identificamos el path		
	66	Given path '/pet/500'	1
	67	When method put	1059
Scenario: [241] Actualizar el nombre de la mascota y el estatus de la mascota a 200.	Then status 200		0
	Background:		
	#Identificamos el path		
	77	Given path '/pet/500'	1
Scenario: [241] Consultar la mascota modificada por estatus buscando por estatus.	Background:		
	#Identificamos el path		
	88	Given path '/pet/status/available'	1
	89	When method get	1059
Scenario: [241] Consultar la mascota modificada por estatus buscando por estatus.	Then status 200		0

2.-

Scenario: [111] Añadir una mascota a la tienda.			
>> Background:			
#Identificamos el path			
13	Given path '/pet'		1
14	And request		105
35	When method post		1059
17:39:13.122 request: 1 > POST https://petstore.swagger.io/v2/pet 1 > Content-Type: application/json; charset=utf-8 1 > Content-Length: 142 1 > Host: petstore.swagger.io 1 > Connection: Keep-Alive 1 > User-Agent: Apache-HttpClient/4.5.12 (Java/14.0.2) 1 > Accept-Encoding: gzip, deflate {"id":500,"category":{"id":2,"name":"Animal"},"name":"Drako","photoUrls":["string"],"tags":[{"id":10,"name":"Canino"}],"status":"available"}			
17:39:13.796 response time in milliseconds: 671 1 < 200 1 < Date: Mon, 10 Jul 2023 22:19:13 GMT 1 < Content-Type: application/json 1 < Transfer-Encoding: chunked 1 < Connection: keep-alive 1 < Access-Control-Allow-Origin: * 1 < Access-Control-Allow-Methods: GET, POST, DELETE, PUT 1 < Access-Control-Allow-Headers: Content-Type, api_key, Authorization 1 < Server: Jetty(9.2.9.v20150224) {"id":500,"category":{"id":2,"name":"Animal"},"name":"Drako","photoUrls":["string"],"tags":[{"id":10,"name":"Canino"}],"status":"available"}			
36	Then status 200		0

- Imagen con todos los métodos ejecutados correctamente.

