**EJERCICIO API**

Get a list of all available books.

GET /books

Get the details of a specific book by providing its unique ID.

GET /books/123

Add a new book.

POST /books

Body:

{

"title": "El Quijote",

"author": "Miguel de Cervantes",

"genre": "Novela",

"year": 1605

}

Update information for an existing book by providing its unique ID.

PUT /books/123

Body:

{

"title": "Don Quijote de la Mancha",

"author": "Miguel de Cervantes",

"genre": "Novela",

"year": 1605

}

Delete a book using its unique ID.

DELETE /books/123

**Ejercicio GHERKIN**

Feature Sign up

@test01

Scenario: User registration

Given: I am a new customer at home page

When: I navigate to sign up page

Then: I should be able to see the Sign up form displayed

Feature Add to cart

@test02

Scenario: Add to cart

Given: I am a customer at item page

When: I add the item to my cart

Then: I should be able to see the confirmation pop up

Feature Visualize my cart

@test02

Scenario: Visualize my cart

Given: I am a customer at anypage

When: I click the cart button

Then: I should be able to visualize the items I added

Notas

Uso de @DataProvider para parametrizar los tests y asignarles valores

Estos devuelven matrices o iteradores (evitar usar los iteradores porque gasta mas recursos)

Handling sincro

Implicit: cuando se usa el get, click, y findElement hay unos waits implícitos

Explicit: usarlos en la base, no en la pagina, crearlo en la base y pasárselo como argumento a mis métodos

Browser único por prueba

No muchas validaciones por test? O sea una assertion por test o que?

Evitar dependencias entre los tests, para eso se usan los before y los grupos

Mencionar siempre la función de los métodos (comentar)