GET Поиск товара по несуществующему ID

SeverityPriorityBehaviorTypeO Normal↑ HighNegativeFunctionalLayerIs FlakyMilestoneAutomationNot setNo-Manual

Status Is Muted

Actual No

Pre-conditions

Postman for Web

Version11.1.17-240522-0631 Browser & OS Chrome 124, Linux x86_64

Steps to reproduce

Step	1
Action	Отправить GET-запрос
Input data	https://qa.demoshopping.ru/products/id/0000
Expected result	Запрос успешно отправлен на сервер
Step	2
Action	Проверить код состояния
Input data	
Expected result	HTTP Status: 404 Not Found
Step	3
Action	Проверить тело ответа от сервера
Input data	
Expected result	Тело ответа в формате HTML возвращается от сервера и будет иметь следующий вид: Товар с таким I D не найден

POST Добавление нового пользователя

SeverityPriorityBehaviorTypeO Normal↑ HighPositiveFunctionalLayerIs FlakyMilestoneAutomationNot setNo-Manual

Status Is Muted

Actual No

Pre-conditions

Postman for Web

Version11.1.17-240522-0631 Browser & OS Chrome 124, Linux x86_64

Пользователь ранее не был зарегистрирован

Steps to reproduce

Step	1
Action	Выполнить требования Postman: Body -> Raw -> JSON { "username": "nikita1234", "password": "nikita12345" }
Input data	
Expected result	Требования выполнены успешно
Step	2
	2
Action	Отправить POST-запрос
	Отправить POST-запрос
Input data Expected	Отправить POST-запрос https://qa.demoshopping.ru/users

Step 4

Input data

Expected

result

HTTP Status: 200 OK

Action	Проверить тело ответа от сервера
Input data	
Expected result	Тело ответа в формате HTML возвращается от сервера и будет иметь следующий вид: Пользователь усп ешно добавлен с ID: { product_id }

PUT Полное обновление товара по ID с невалидными данными поля "name"

SeverityPriorityBehaviorTypeO Normal↑ HighNegativeFunctional

Layer Is Flaky Milestone Automation

Not set No - Manual Status Is Muted

Actual No

Pre-conditions

Postman for Web

Version11.1.17-240522-0631 Browser & OS Chrome 124, Linux x86_64

Steps to reproduce

Input data

Expected

result

HTTP Status: 400 Bad Request

Bыполнить требования Postman: Body -> Raw -> JSON {	Step	1
Expected result Требования выполнены успешно Step 2 Action Отправить PUT-запрос Input data https://qa.demoshopping.ru/products/id/{{productId}}. Expected result Запрос успешно отправлен на сервер Step 3	Action	Postman: Body -> Raw -> JSON { "name": 123, "description": "test", "price": 888, "category": "test", "manufacturer": "test", "imageUrl": "test", "freeShipping": true
Step Action Отправить PUT-запрос Input data https://qa.demoshopping.ru/products/id/{{productId}}} Expected result Запрос успешно отправлен на сервер Step 3	Input data	
Action Отправить PUT-запрос Input data <a a="" href="https://qa.demoshopping.ru/products/id/{{productId}}} Expected result Step 3</td><td></td><td>Требования выполнены успешно</td></tr><tr><td>Input data <a href=" https:="" id="" products="" qa.demoshopping.ru="" {{productid}}}<=""> Expected result Step 3	Step	2
Expected result Запрос успешно отправлен на сервер Step 3	Action	Отправить PUT-запрос
Step 3	Input data	https://qa.demoshopping.ru/products/id/{{productId}}
		Запрос успешно отправлен на сервер
Action Проверить код состояния	Step	3
	Action	Проверить код состояния

Step	4
Action	Проверить тело ответа от сервера
Input data	
Expected result	Тело ответа в формате HTML возвращается от сервера и будет иметь следующий вид: Неправильный т ип значения для 'name'

РАТСН Частичное обновление товара по ID

Severity **Priority Behavior Type** O Normal ↑ High Positive **Functional** Milestone **Automation** Layer Is Flaky Not set Manual No

Status Is Muted

Actual No

Pre-conditions

Postman for Web

Version11.1.17-240522-0631 Browser & OS Chrome 124, Linux x86_64

Steps to reproduce

1 Step

> Выполнить требования Postman: Body -> Raw -> JSON

Action "price": 900

}

Input data

Expected Требования выполнены успешно result

2 Step

Action Отправить РАТСН-запрос

https://qa.demoshopping.ru/products/id/{{productId}}} Input data

Expected result

Запрос успешно отправлен на сервер

3 Step

Action Проверить код состояния

Input data

Expected HTTP Status: 200 OK result

4 Step

Action Проверить тело ответа от сервера

Input data

Expected result

Тело ответа в формате HTML возвращается от сервера и будет иметь следующий вид: Товар частичн о обновлён

DELETE Удаление товара по запрещенному ID

SeverityPriorityBehaviorTypeO Normal↑ HighNegativeFunctionalLayerIs FlakyMilestoneAutomationNot setNo-Manual

Status Is Muted

Actual No

Pre-conditions

Postman for Web

Version11.1.17-240522-0631 Browser & OS Chrome 124, Linux x86_64

Steps to reproduce

Step	1
Action	Отправить DELETE-запрос
Input data	https://qa.demoshopping.ru/products/id/2
Expected result	Запрос успешно отправлен на сервер
Step	2
Action	Проверить код состояния
Input data	
Expected result	HTTP Status: 403 Forbidden
Step	3
Action	Проверить тело ответа от сервера
Input data	
Expected result	Тело ответа в формате HTML возвращается от сервера и будет иметь следующий вид: Удаление этого товара запрещено