# GET - Список товаров, отфильтрованный по категории (200)

Список товаров, отфильтрованный по категории, возвращается, Code 200 при GET-запросе на /products/FindByCategory

Priority	Behavior	Туре	Layer
↑ High	Positive	Integration	API
ls Flaky	Milestone	Automation	Status
No	-	Manual	Actual

Is Muted

No

#### **Pre-conditions**

Окружение: Postman Ver. 11.27.3

Headers request:

- 1. Content-Type: application/json
- 2. 2. Accept: \*/\*
- 3. Environment Variables:
  - {{base\_url}}: <a href="https://qa.demoshopping.ru">https://qa.demoshopping.ru</a>;
     {{category}}: Laptops

#### **Tags**

**GET** 

Step	1
Action	Отправить GET запрос на

```
Тело ответа в формате JSON будет иметь следующий вид (массив записей):
                [
                     "product_id": 3,
"name": "Samsung Air 12",
                     "description": "Описание для Samsung Air 12",
Expected
                     "price": "1289.00",
result
                     "category": "Laptops",
                     "manufacturer": "Samsung",
                     "imageUrl": "/images/laptop.webp",
                     "freeShipping": 0
                   },
                ]
Step
                4
Action
                Проверить фильтрацию ответа от сервера
Input data
Expected
                Значение у полученных товаров - "category": "Laptops"
result
                5
Step
Action
                Проверить заголовки ответа
Input data
                Server: nginx/1.24.0 (Ubuntu)
Expected
                Content-Type: application/json; charset=utf-8
```

access-control-allow-methods: GET,POST,DELETE,PUT

result

## POST - Добавление нового продукта в каталог (200)

Добавление нового продукта, Code 200 при POST-запросе с валидным телом на /add-product Copy

Priority	Behavior	Туре	Layer
<b>↑</b> High	Positive	Integration	API
ls Flaky	Milestone	Automation	Status
No	-	Manual	Actual

Is Muted

No

#### **Pre-conditions**

Окружение: Postman Ver. 11.27.3

Headers request:

1. Content-Type: application/json

2. Accept: \*/\*

**Environment Variables:** 

1. {{base\_url}}: <a href="https://qa.demoshopping.ru">https://qa.demoshopping.ru</a>;

Pre-request: (расчет переменной для параметра price) pm.variables.set("price", \_.random(1,5000));

#### **Tags**

**POST** 

Step

```
Step
                  1
                   Отправить POST запрос на https://qa.demoshopping.ru/add-product
                   Тело запроса:
                  "description": "\{\{\$randomStreetName\}\}\ \{\{\$randomBsNoun\}\}",
Action
                   "price": {{price}},
                   "category": "{{$randomWeekday}}",
                   "manufacturer": "{{$randomCompanyName}}}",
                   "imageUrl": "{{$randomAnimalsImage}}",
                   "freeShipping": {{$randomBoolean}}
                   }
Input data
Expected result
                   Запрос успешно отправлен на сервер
Step
                  2
Action
                   Проверить код состояния
Input data
                  HTTP Status: 200 OK
Expected result
                  3
```

Action	Проверить тело ответа от сервера
Input data	
Expected result	Продукт успешно добавлен с ID: {номер id}
Step	4
Action	Проверить заголовки ответа
Input data	
Expected result	Server: nginx/1.24.0 (Ubuntu) Content-Type: application/json; charset=utf-8 access-control-allow-methods: GET,POST,DELETE,PUT

# PUT - Данные товара с несуществующим id не обновляются (404)

Данные товара с несуществующим id не обновляются, Code 404 при PUT-запросе на /products/id/:id

Priority	Behavior	Туре	Layer
↑ High	Negative	Integration	API
Is Flaky	Milestone	Automation	Status
No	_	Manual	Actual

Is Muted

No

#### **Pre-conditions**

Окружение: Postman Ver. 11.27.3

Headers request:

1. Content-Type: application/json

2. Accept: \*/\*

3. Environment Variables:

1. {{base\_url}}: <a href="https://qa.demoshopping.ru">https://qa.demoshopping.ru</a>;

id/100500 - id несуществующего в системе продукта

#### Tags

**PUT** 

```
1
Step
                          Отправить PUT запрос <a href="https://qa.demoshopping.ru/products/id/100500">https://qa.demoshopping.ru/products/id/100500</a>
                          Тело запроса:
                           "name": "Super Product-2",
                           "description": "Super product.",
Action
                           "price": 200,
                           "category": "Laptops",
                           "manufacturer": "Manufacturer Inc",
                           "imageUrl": "http://example.com/image2.jpg",
                           "freeShipping": false
Input data
Expected result
                          Запрос успешно отправлен на сервер
                          2
Step
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	Товар с таким ID не найден
Step	4
Action	Проверить заголовки ответа
Input data	
Expected result	Server: nginx/1.24.0 (Ubuntu) Content-Type: application/json; charset=utf-8 access-control-allow-methods: GET,POST,DELETE,PUT

# РАТСН - Количество несуществующего товара в корзине не обновляется - (404)

Количество несуществующего товара в корзине авторизованного пользователя не обновляется, Code 404 при PATCH-запросе на /cart/cartItemID

Priority	Behavior	Туре	Layer
↑ High	Negative	Integration	API
Is Flaky	Milestone	Automation	Status

Is Muted

Nο

#### **Pre-conditions**

Окружение: Postman Ver. 11.27.3

**Environment Variables:** 

- 1. {{base\_url}}: <a href="https://qa.demoshopping.ru">https://qa.demoshopping.ru</a>;
- 2. {{token}}: токен авторизации

В системе зарегистрирован пользователь с данными: TestBuyer/TestPassword. Пользователь авторизован в системе, получен токен авторизации для переменной окружения token. В корзине пользователя нет товара с указанным id - cart/100500

Headers request:

- 1. Content-Type: application/json
- 2. Accept: \*/\*
- 3. Authorisation: Bearer {{token}}

#### Tags

**PATCH** 

Step	1
Action	Отправить PATCH запрос <a href="https://qa.demoshopping.ru/cart/100500">https://qa.demoshopping.ru/cart/100500</a> Тело запроса: {     "quantity": 1 }
Input data	
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	{ "message": "Товар в корзине не найден" }
Step	4
Action	Проверить заголовки ответа
Input data	
Expected result	Server: nginx/1.24.0 (Ubuntu) Content-Type: application/json; charset=utf-8 access-control-allow-methods: GET,POST,DELETE,PUT,PATCH

## **DELETE** - Товар из основного каталога не удаляется (403)

Товар из основного каталога не доступен для удаления, Code 403 при DELETE-запросе на /products/id/:id

Priority	Behavior	Туре	Layer
↑ High	Positive	Integration	API
Is Flaky	Milestone	Automation	Status
No	-	Manual	Actual

Is Muted

No

#### **Pre-conditions**

Окружение: Postman Ver. 11.27.3

Headers request:

1. Content-Type: application/json

**Environment Variables:** 

1. {{base\_url}}: <a href="https://qa.demoshopping.ru">https://qa.demoshopping.ru</a>;

id/1 - id продукта в системе, продукт из основного каталога, защищен от удаления

#### Tags

Step

4

**DELETE** 

Step	1
Action	Отправка DELETE запроса на <a href="https://qa.demoshopping.ru/products/id/1">https://qa.demoshopping.ru/products/id/1</a>
Input data	
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	Удаление этого товара запрещено

Action	Проверить заголовки ответа
Input data	
Expected result	Server: nginx/1.24.0 (Ubuntu) Content-Type: application/json; charset=utf-8 access-control-allow-methods: GET,POST,DELETE,PUT