

Тестирование эндпоинта /posts (<https://jsonplaceholder.typicode.com/>)

Таблица тест-кейсов для GET /posts

TCID	Название теста / Сценарий	Запрос	Ожидаемый результат
1	Получение всех постов	GET /posts	200 OK, массив из 100 объектов, каждый объект содержит поля id, title, body, userId
2	Попытка получить пост с ID=0	GET /posts/0	404 Not Found
3	Получение поста с ID=1	GET /posts/1	200 OK, объект с id=1, title, body, userId
4	Получение поста с ID=2	GET /posts/2	200 OK, объект с id=2, title, body, userId
5	Получение поста с ID=99	GET /posts/99	200 OK, объект с id=99, title, body, userId
6	Получение поста с ID=100	GET /posts/100	200 OK, объект с id=100, title, body, userId
7	Попытка получить пост с ID=101	GET /posts/101	404 Not Found
8	Получение комментариев для существующего поста (ID=1)	GET /posts/1/comments	200 OK, массив объектов с postId=1, каждый объект имеет поля id, name, email, body
9	Попытка получения комментариев для несуществующего поста (ID=0)	GET /posts/0/comments	200 OK, пустой массив

POST/PUT/PATCH/DELETE нельзя нормально проверить с mock API, поэтому создано по одному запросу с максимально валидными данными для того, чтобы показать навыки работы с каждым методом.

Итоговая коллекция

```
✓ jsonplaceholder /posts testing
  GET TCID = 1 Получение всех постов
  GET TCID = 2 Получение поста с ID=0
  GET TCID = 3 Получение поста с ID=1
  GET TCID = 4 Получение поста с ID=2
  GET TCID = 5 Получение поста с ID=99
  GET TCID = 6 Получение поста с ID=100
  GET TCID = 7 Получение поста с ID=101
  GET TCID = 8 Получение комментариев для существующего поста (ID=1)
  GET TCID = 9 Попытка получения комментариев для несуществующего поста (ID=0)
  POST Пример запроса POST
  PUT Пример запроса PUT
  PATCH Пример запроса PATCH
  DEL Пример запроса DELETE
```

Run

jsonplaceholder /posts testing - Run results

🕒 Ran today at 06:39:03 PM · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	jsonplaceholder	1	4s 292ms	24	166 ms

[All Tests](#) Passed (24) Failed (0) Skipped (0) [View Summary](#)

Iteration 1

1

GET TCID = 1 Получение всех постов
https://jsonplaceholder.typicode.com//posts

PASS Status code is 200
PASS Response is an array of 100 objects
PASS Each object has required fields

2

GET TCID = 2 Получение поста с ID=0
https://jsonplaceholder.typicode.com//posts/0

PASS Status code is 404

3

GET TCID = 3 Получение поста с ID=1
https://jsonplaceholder.typicode.com//posts/1

PASS Status code is 200
PASS Response object has correct id

4

GET TCID = 4 Получение поста с ID=2
https://jsonplaceholder.typicode.com//posts/2

PASS Status code is 200
PASS Response object has correct id

5

GET TCID = 5 Получение поста с ID=99
https://jsonplaceholder.typicode.com//posts/99

PASS Status code is 200
PASS Response object has correct id

6

GET TCID = 6 Получение поста с ID=100
https://jsonplaceholder.typicode.com//posts/100

PASS Status code is 200

GET TCID = 7 Получение поста с ID=101

<https://jsonplaceholder.typicode.com//posts/101>

404 • 45 ms • 1.16 KB • 1

PASS Status code is 404

GET TCID = 8 Получение комментариев для существующего поста (ID=1)

<https://jsonplaceholder.typicode.com//posts/1/comments>

200 • 44 ms • 1.797 KB • 2

PASS Status code is 200

PASS All comments belong to postId=1

GET TCID = 9 Попытка получения комментариев для несуществующего поста (ID=0)

<https://jsonplaceholder.typicode.com//posts/0/comments>

200 • 42 ms • 1.148 KB • 2

PASS Status code is 200

PASS Response is an empty array

POST Пример запроса POST

<https://jsonplaceholder.typicode.com//posts>

201 • 155 ms • 1.378 KB • 1

PASS Status code is 201

PUT Пример запроса PUT

<https://jsonplaceholder.typicode.com//posts/1>

200 • 420 ms • 1.143 KB • 2

PASS Status is 200

PASS Response contains id field

PATCH Пример запроса PATCH

<https://jsonplaceholder.typicode.com//posts/1>

200 • 154 ms • 1.321 KB • 2

PASS Status is 200

PASS Response contains id field

DELETE Пример запроса DELETE

<https://jsonplaceholder.typicode.com//posts/1>

200 • 157 ms • 1.132 KB • 2

PASS Status code is 200

PASS Response body is empty