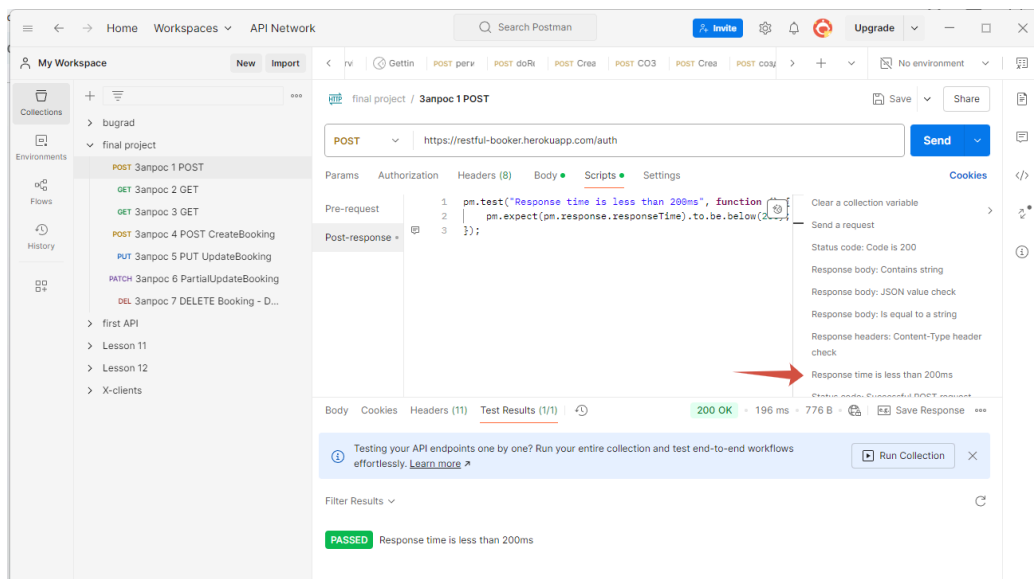


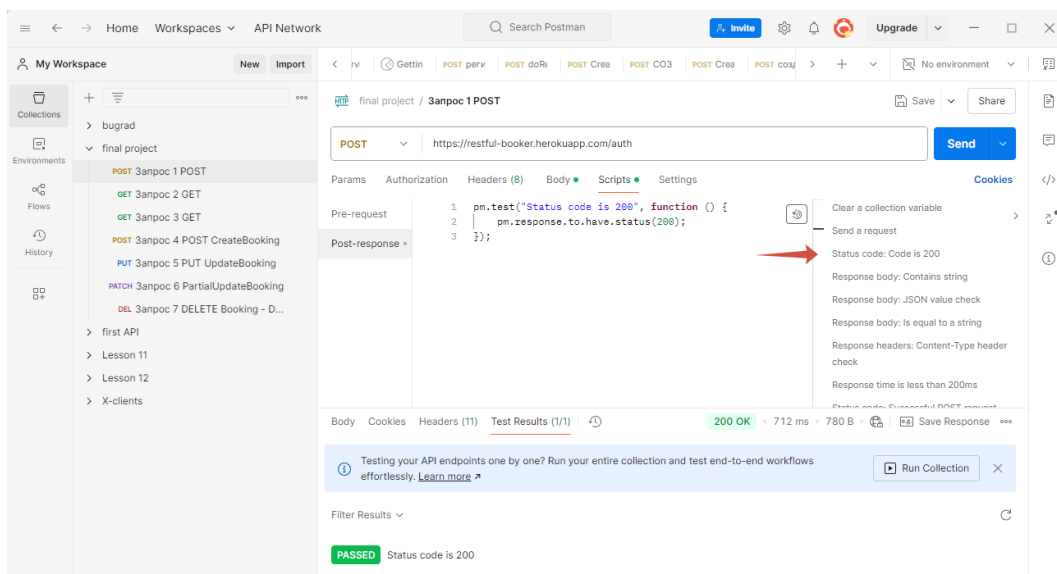
1. Создать в POSTMAN новую коллекцию на основе свагера:
[Загрузка...](#) Минимум 4 запроса (последовательность не важна) (и какие запросы — тоже не важно, но хотелось бы стандартные: GET, POST, PUT, DELETE)

- Покрытие минимальными проверками с помощью тестов (по 3–4 на каждый API-запрос)
- Задание со звездочкой: уметь запускать коллекцию из командной строки (придется погуглить)

Запрос 1. Тест “Валидация времени ответа”.



Тест “Валидация статус кода”.



Поменяла статус код с 200 на 404.

The screenshot shows the Postman interface with a collection named 'final project' containing a sub-collection '3аппоч 1 POST'. The selected item is a POST request to 'https://restful-booker.herokuapp.com/auth'. The 'Scripts' tab is active, showing a pre-request script:

```
1 pm.test("Status code is 404", function () {  
2   pm.response.to.have.status(404);  
3 });
```

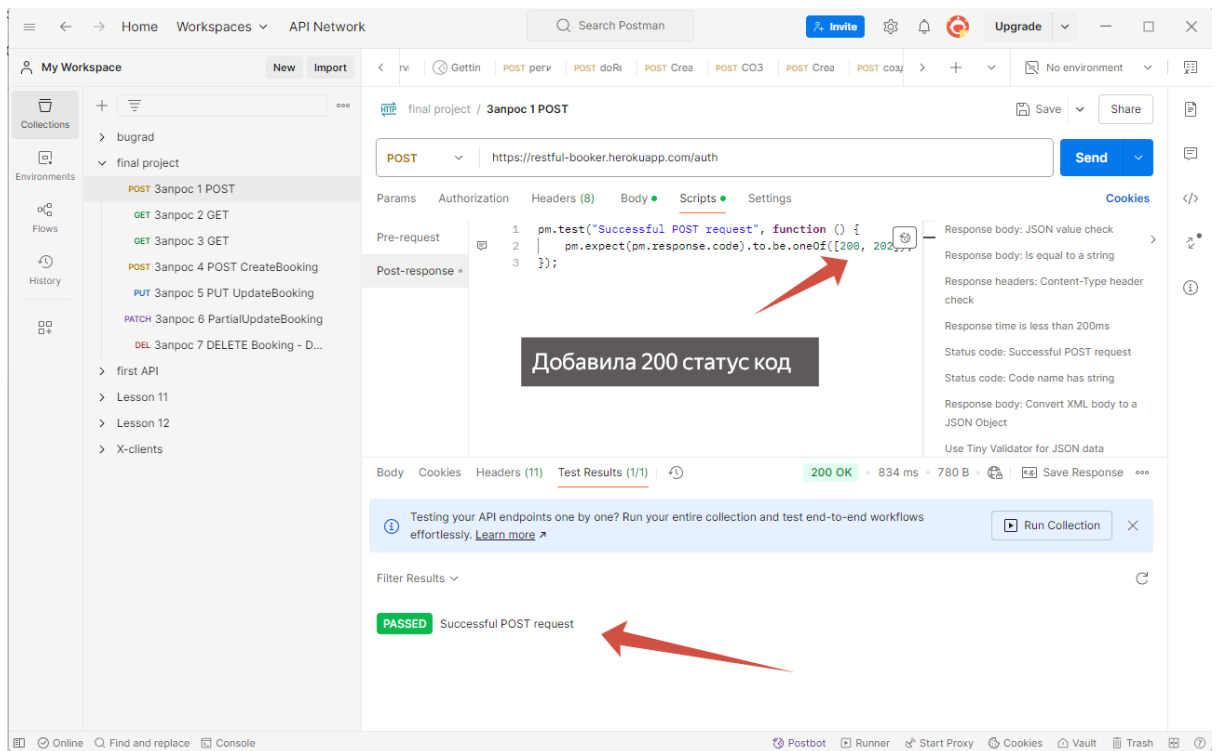
 A red arrow points from the text 'Поменяла статус код с 200 на 404' to the script. The 'Test Results' tab shows a failed assertion: 'Status code: Code is 200'. The status bar at the bottom indicates '200 OK' and '610 ms'. A message at the bottom says 'Testing your API endpoints one by one? Run your entire collection and test end-to-end workflows effortlessly. Learn more'.

Тест “Является ли код состояния одним из набора”.

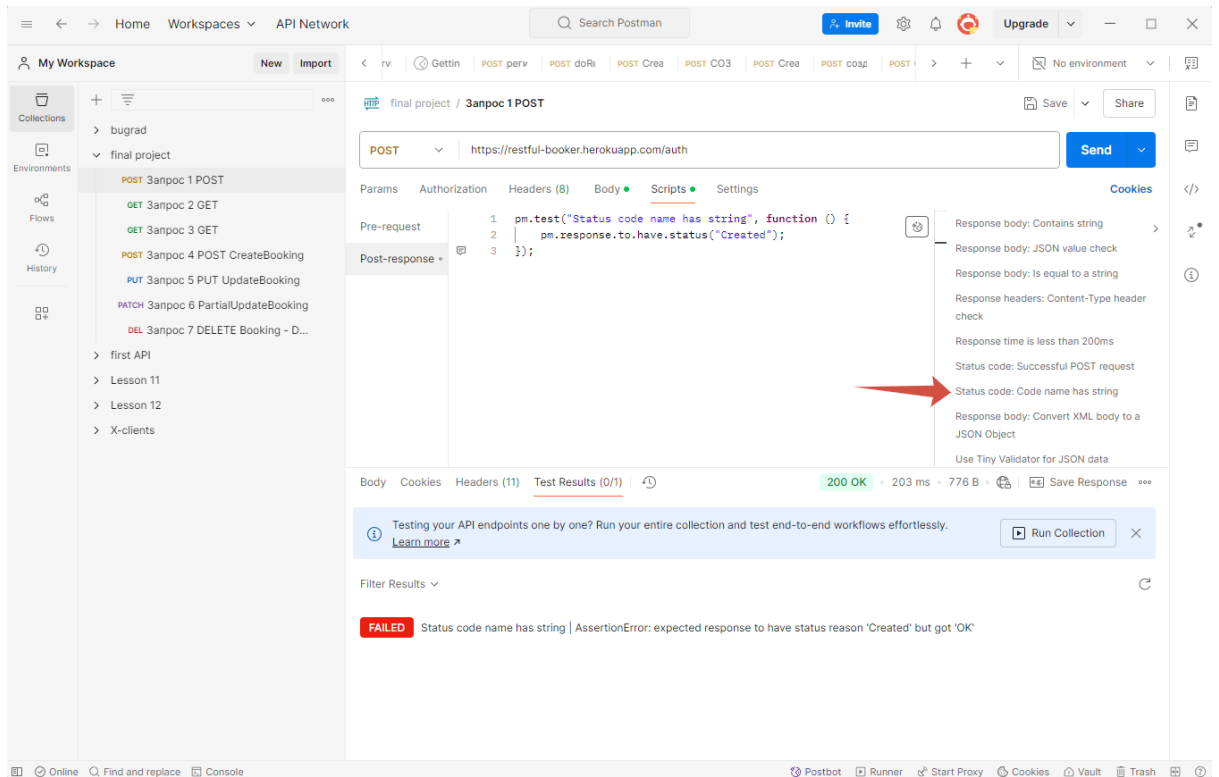
The screenshot shows the Postman interface with the same collection and item. The 'Scripts' tab is active, showing a pre-request script:

```
1 pm.test("Successful POST request", function () {  
2   pm.expect(pm.response.code).to.be.oneOf([201, 202]);  
3 });
```

 A red arrow points from the text 'Тест “Является ли код состояния одним из набора”.' to the script. The 'Test Results' tab shows a failed assertion: 'Status code: Successful POST request'. The status bar at the bottom indicates '200 OK' and '716 ms'. A message at the bottom says 'Testing your API endpoints one by one? Run your entire collection and test end-to-end workflows effortlessly. Learn more'.



Тест “Проверка текста кода состояния”.



Home Workspaces API Network Search Postman Invite Upgrade

My Workspace New Import

bugrad
final project

POST 3anpoc 1 POST
GET 3anpoc 2 GET
GET 3anpoc 3 GET
POST 3anpoc 4 POST CreateBooking
PUT 3anpoc 5 PUT UpdateBooking
PATCH 3anpoc 6 PartialUpdateBooking
DEL 3anpoc 7 DELETE Booking - D...

first API
Lesson 11
Lesson 12
X-clients

final project / 3anpoc 1 POST

POST https://restful-booker.herokuapp.com/auth Send

Params Authorization Headers (8) Body Scripts Settings Cookies

```
1 pm.test("Status code name has string", function () {  
2   pm.response.to.have.status("Created");  
3 });  
4 pm.test("Response time is less than 200ms", function () {  
5   pm.expect(pm.response.responseTime).to.be.below(200);  
6 });  
7 pm.test("Successful POST request", function () {  
8   pm.expect(pm.response.code).to.be.oneOf([200, 202]);  
9 });  
10 pm.test("Status code is 200", function () {  
11   pm.response.to.have.status(200);  
12 });
```

Body Cookies Headers (11) Test Results (3/4) 200 OK 152 ms 776 B Save Response

Testing your API endpoints one by one? Run your entire collection and test end-to-end workflows effortlessly. [Learn more](#) Run Collection

Filter Results

FAILED Status code name has string | AssertionError: expected response to have status reason 'Created' but got 'OK'

PASSED Response time is less than 200ms

PASSED Successful POST request

PASSED Status code is 200

Online Find and replace Console Postbot Runner Start Proxy Cookies Vault Trash

Запрос 2.

Home Workspaces API Network Search Postman Invite Upgrade

My Workspace New Import

bugrad
final project

POST 3anpoc 1 POST
GET 3anpoc 2 GET
GET 3anpoc 3 GET
POST 3anpoc 4 POST CreateBooking
PUT 3anpoc 5 PUT UpdateBooking
PATCH 3anpoc 6 PartialUpdateBooking
DEL 3anpoc 7 DELETE Booking - D...

first API
Lesson 11
Lesson 12
X-clients

final project / 3anpoc 2 GET

GET https://restful-booker.herokuapp.com/booking Send

Params Authorization Headers (6) Body Scripts Settings Cookies

```
1 pm.test("Response time is less than 200ms", function () {  
2   pm.expect(pm.response.responseTime).to.be.below(200);  
3 });  
4 pm.test("Successful POST request", function () {  
5   pm.expect(pm.response.code).to.be.oneOf([201, 202]);  
6 });  
7 pm.test("Status code name has string", function () {  
8   pm.response.to.have.status("Created");  
9 });  
10 pm.test("Status code is 200", function () {  
11   pm.response.to.have.status(200);  
12 });
```

Body Cookies Headers (11) Test Results (1/4) 200 OK 742 ms 252.36 KB Save Response

Testing your API endpoints one by one? Run your entire collection and test end-to-end workflows effortlessly. [Learn more](#) Run Collection

Filter Results

FAILED Response time is less than 200ms | AssertionError: expected 742 to be below 200

FAILED Successful POST request | AssertionError: expected 200 to be one of [201, 202]

FAILED Status code name has string | AssertionError: expected response to have status reason 'Created' but got 'OK'

PASSED Status code is 200

Online Find and replace Console Postbot Runner Start Proxy Cookies Vault Trash

Запрос 3.

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The 'final project' collection is expanded, showing a 'GET 3anpoc 3 GET' request. The request is a GET method to the URL 'https://restful-booker.herokuapp.com/booking/1'. The 'Scripts' tab is active, showing a series of PM tests. The 'Test Results' section shows four passed tests: 'Content-Type is present', 'Status code is 200', 'Response time is less than 200ms', and 'Successful POST request'. The status bar indicates a 200 OK response with a response time of 165 ms and a body size of 889 B.

```
1 pm.test("Content-Type is present", function () {
2   pm.response.to.have.header("Content-Type");
3 });
4 pm.test("Status code is 200", function () {
5   pm.response.to.have.status(200);
6 });
7 pm.test("Response time is less than 200ms", function () {
8   pm.expect(pm.response.responseTime).to.be.below(200);
9 });
10 pm.test("Successful POST request", function () {
11   pm.expect(pm.response.code).to.be.oneOf([200, 202]);
12 });
```

Test Results (4/4):

- PASSED Content-Type is present
- PASSED Status code is 200
- PASSED Response time is less than 200ms
- PASSED Successful POST request

Запрос 4.

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The 'final project' collection is expanded, showing a 'POST 3anpoc 4 POST CreateBooking' request. The request is a POST method to the URL 'https://restful-booker.herokuapp.com/booking'. The 'Scripts' tab is active, showing a series of PM tests. The 'Test Results' section shows four tests: 'Status code is 200' (PASSED), 'Response time is less than 200ms' (PASSED), 'Successful POST request' (FAILED), and 'Content-Type is present' (PASSED). The failed test has an error message: 'AssertionError: expected 200 to be one of [201, 202]'. The status bar indicates a 200 OK response with a response time of 137 ms and a body size of 942 B.

```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
4 pm.test("Response time is less than 200ms", function () {
5   pm.expect(pm.response.responseTime).to.be.below(200);
6 });
7 pm.test("Successful POST request", function () {
8   pm.expect(pm.response.code).to.be.oneOf([201, 202]);
9 });
10 pm.test("Content-Type is present", function () {
11   pm.response.to.have.header("Content-Type");
12 });
```

Test Results (3/4):

- PASSED Status code is 200
- PASSED Response time is less than 200ms
- FAILED Successful POST request | AssertionError: expected 200 to be one of [201, 202]
- PASSED Content-Type is present

Запрос 5.

The screenshot shows a Postman interface for a workspace named "My Workspace". The selected collection is "final project" and the environment is "No environment". The active request is a PUT request to "https://restful-booker.herokuapp.com/booking/20714". The "Scripts" tab is active, showing a series of PM tests. The "Test Results" section at the bottom shows 2 out of 4 tests failed.

Request: PUT `https://restful-booker.herokuapp.com/booking/20714`

Scripts:

```
1 pm.test("Successful POST request", function () {
2   pm.expect(pm.response.code).to.be.oneOf([200, 201, 202]);
3 });
4 pm.test("Response time is less than 200ms", function () {
5   pm.expect(pm.response.responseTime).to.be.below(200);
6 });
7 pm.test("Status code is 200", function () {
8   pm.response.to.have.status(200);
9 });
10 pm.test("Status code name has string", function () {
11   pm.response.to.have.status("Created");
12 });
```

Test Results (2/4):

- FAILED** Response time is less than 200ms | AssertionError: expected 243 to be below 200
- PASSED** Status code is 200
- FAILED** Status code name has string | AssertionError: expected response to have status reason 'Created' but got 'OK'

Запрос 6.

The screenshot shows a Postman interface for a workspace named "My Workspace". The selected collection is "final project" and the environment is "No environment". The active request is a PATCH request to "https://restful-booker.herokuapp.com/booking/2070". The "Scripts" tab is active, showing a series of PM tests. The "Test Results" section at the bottom shows 4 out of 4 tests passed.

Request: PATCH `https://restful-booker.herokuapp.com/booking/2070`

Scripts:

```
1 pm.test("Response time is less than 200ms", function () {
2   pm.expect(pm.response.responseTime).to.be.below(200);
3 });
4 pm.test("Status code is 200", function () {
5   pm.response.to.have.status(200);
6 });
7 pm.test("Successful POST request", function () {
8   pm.expect(pm.response.code).to.be.oneOf([200, 201, 202]);
9 });
10 pm.test("Content-Type is present", function () {
11   pm.response.to.have.header("Content-Type");
12 });
```

Test Results (4/4):

- PASSED** Response time is less than 200ms
- PASSED** Status code is 200
- PASSED** Successful POST request
- PASSED** Content-Type is present

Запрос 7.

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection of requests under 'final project', including '3anpoc 1 POST', '3anpoc 2 GET', '3anpoc 3 GET', '3anpoc 4 POST CreateBooking', '3anpoc 5 PUT UpdateBooking', '3anpoc 6 PartialUpdateBooking', and '3anpoc 7 DELETE Booking - D...'. The main panel shows the '3anpoc 7 DELETE Booking - DeleteBooking' request. The URL is 'https://restful-booker.herokuapp.com/booking/5182'. The method is 'DELETE'. The 'Test' tab is active, showing the following test scripts:

```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
4 pm.test("Response time is less than 200ms", function () {
5   pm.expect(pm.response.responseTime).to.be.below(200);
6 });
7 pm.test("Successful POST request", function () {
8   pm.expect(pm.response.code).to.be.oneOf([201, 202]);
9 });
```

The 'Test Results' tab shows the following results:

- FAILED** Status code is 200 | AssertionError: expected response to have status code 200 but got 201
- PASSED** Response time is less than 200ms
- PASSED** Successful POST request

The status bar at the bottom indicates '201 Created', '161 ms', and '749 B'. The bottom of the interface shows various toolbars including 'Online', 'Find and replace', 'Console', 'Postbot', 'Runner', 'Start Proxy', 'Cookies', 'Vault', 'Trash', and 'Help'.