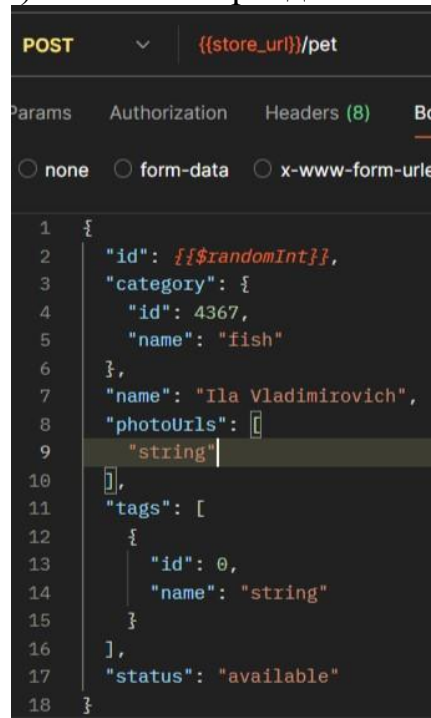


Тема лабораторной работы	lb_7
Выполняющий	Малолетко Александр

Ход Работы

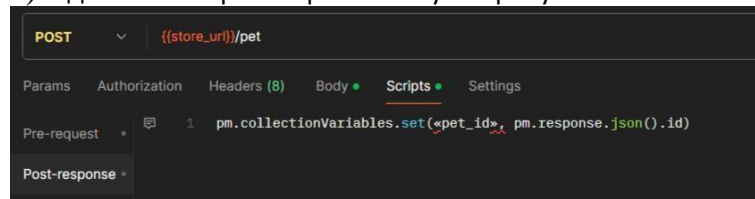
1)меняю id на случайное



The screenshot shows a REST client interface with a POST request to `{{store_url}}/pet`. The request body is a JSON object with the following structure:

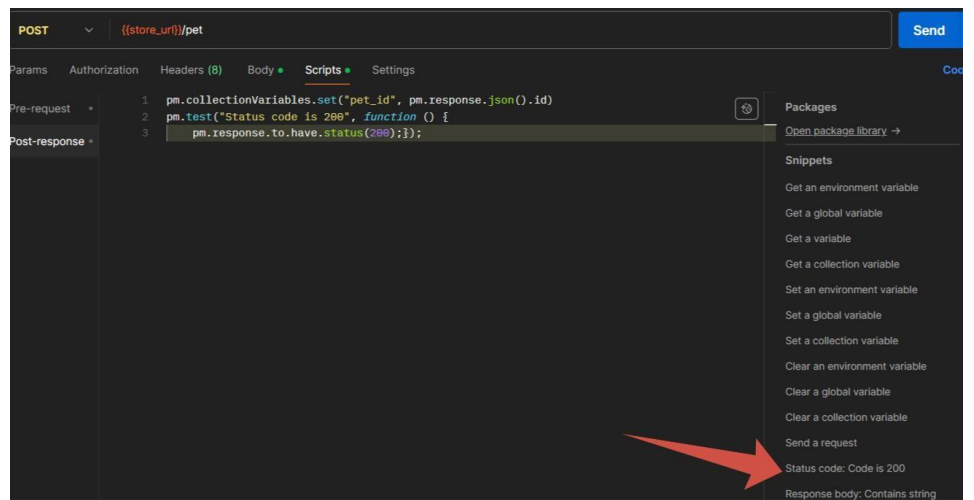
```
1 {
2   "id": {{$randomInt}},
3   "category": {
4     "id": 4367,
5     "name": "fish"
6   },
7   "name": "Ila Vladimirovich",
8   "photoUrls": [],
9   "string"
10 },
11 "tags": [
12   {
13     "id": 0,
14     "name": "string"
15   }
16 ],
17 "status": "available"
18 }
```

2) и добавляю в post-reponse новую строку



The screenshot shows the same REST client interface, but with the 'Scripts' tab selected. A pre-request script is added:

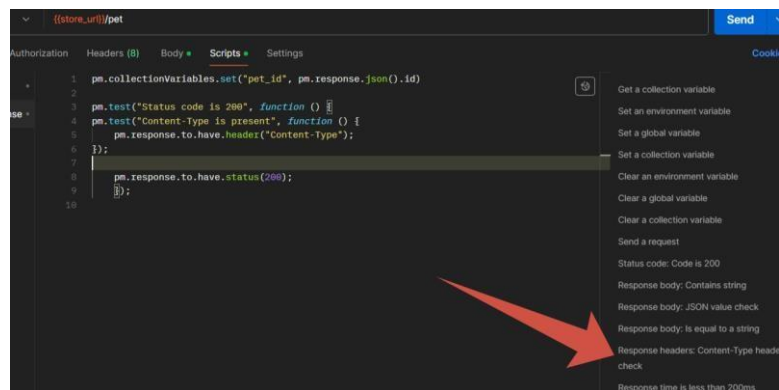
```
1 pm.collectionVariables.set(«pet_id», pm.response.json().id)
```



3) добавляю в код status code is 200

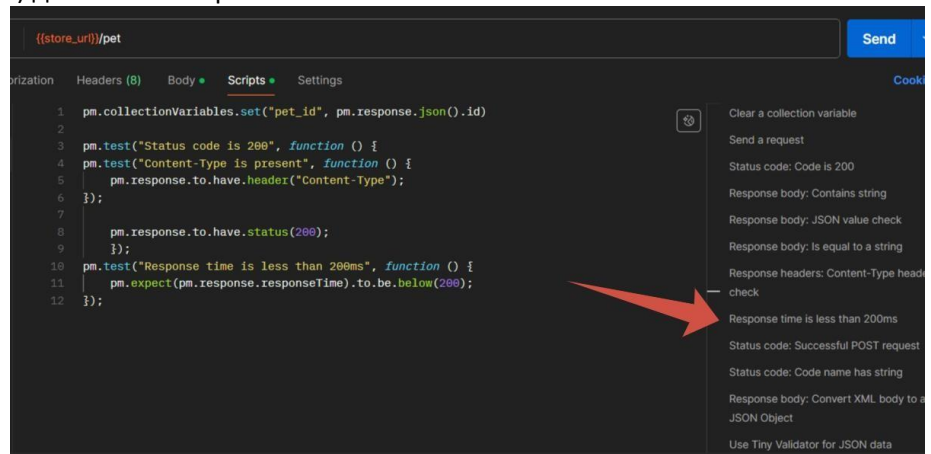
И теперь при каждом запуске автотеста он будет проверять вернулся ли нам код 200

4) добавляю новый снippet Response headers:Content-Type header check



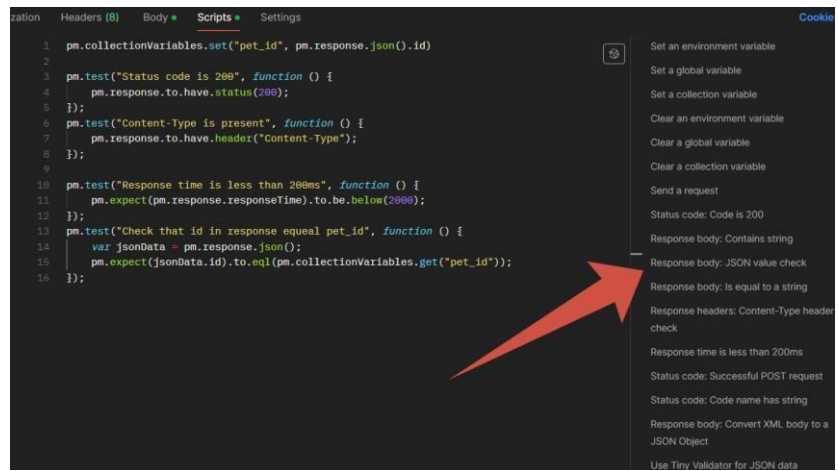
Проверяет есть ли заголовок Content-Type

5) добавляем response time is less than 200ms



Заменяем его значение на 2000 ms

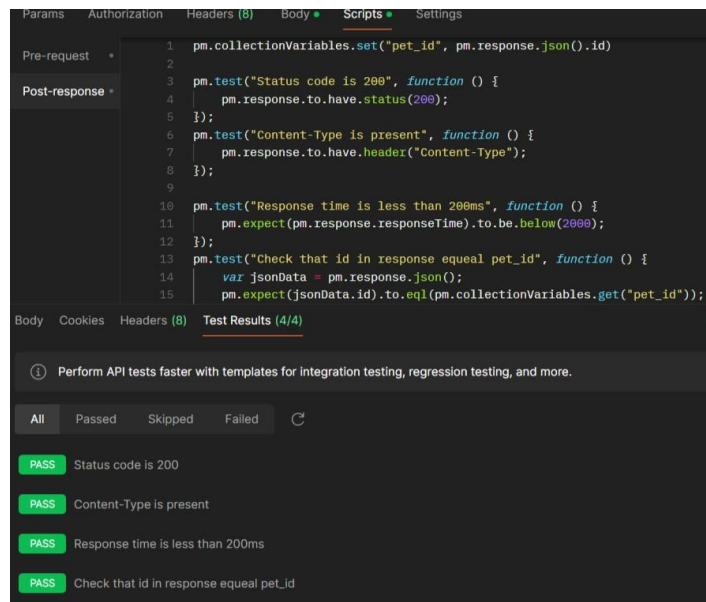
Этот скрипт проверяет ответит ли сервер за определенное время если ему это не удалось то тест провалится



6) добавляю Response body: JSON value check и немного его меняю

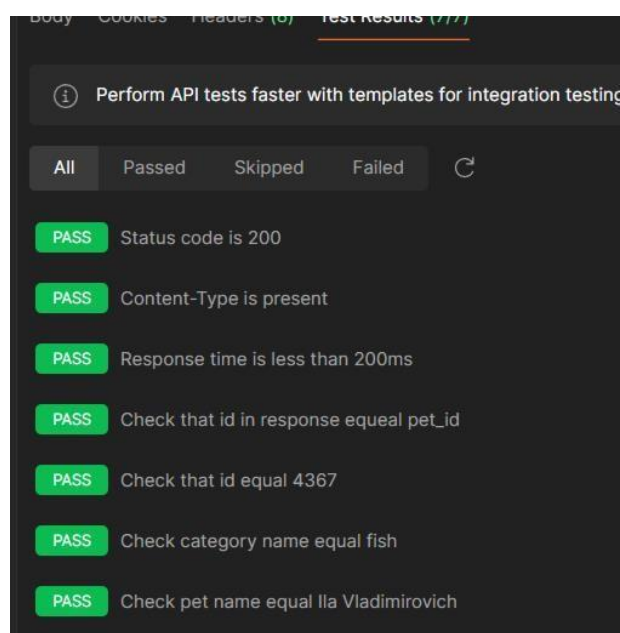
Этот тест будет проверять чтобы id который возвращается в ответе равен id из pet_id

7) убеждаюсь что все тесты прошли успешно



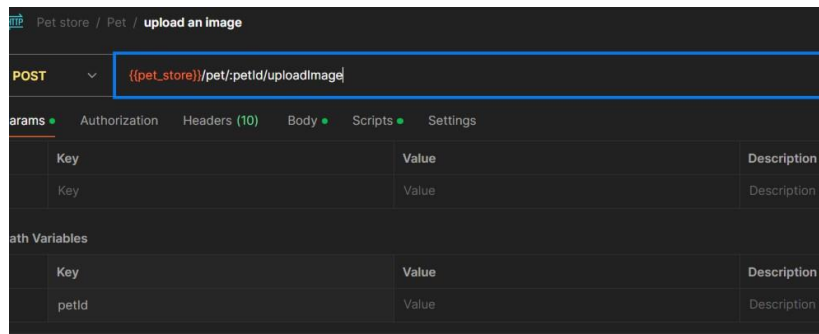
8) добавляю 3 новых теста проверяющих id категории и имя категории и имя питомца

```
2
3 pm.test("Status code is 200", function () {
4   pm.response.to.have.status(200);
5 });
6 pm.test("Content-Type is present", function () {
7   pm.response.to.have.header("Content-Type");
8 });
9
10 pm.test("Response time is less than 200ms", function () {
11   pm.expect(pm.response.responseTime).to.be.below(200);
12 });
13 pm.test("Check that id in response equal pet_id", function () {
14   var jsonData = pm.response.json();
15   pm.expect(jsonData.id).to.eql(pm.collectionVariables.get("pet_id"));
16 });
17 pm.test("Check that id equal 4367", function () {
18   var jsonData = pm.response.json();
19   pm.expect(jsonData.category.id).to.eql(4367);
20 });
21 pm.test("Check category name equal fish", function () {
22   var jsonData = pm.response.json();
23   pm.expect(jsonData.category.name).to.eql("fish");
24 });
25 pm.test("Check pet name equal Ila Vladimirovich", function () {
26   var jsonData = pm.response.json();
27   pm.expect(jsonData.name).to.eql("Ila Vladimirovich");
28 });
```



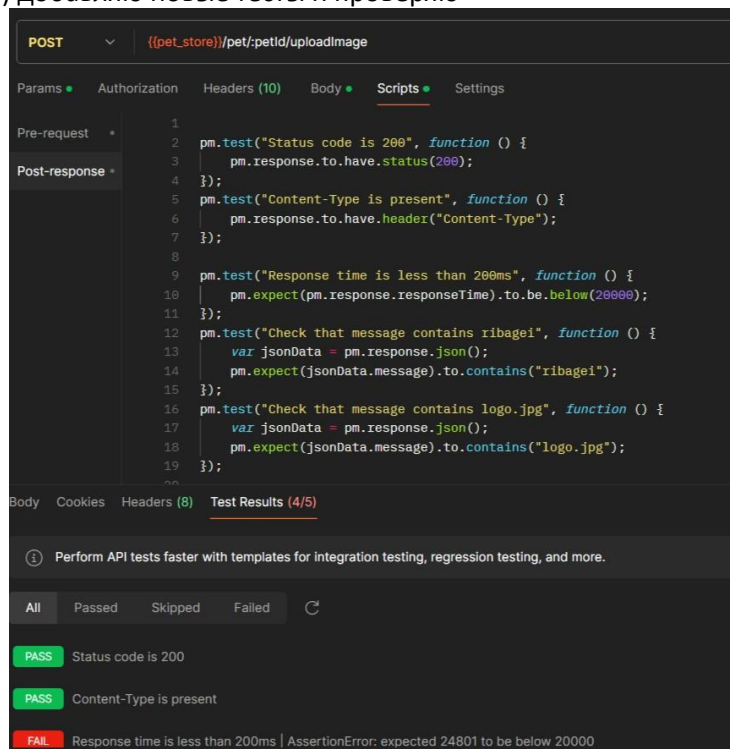
9) убеждаюсь что тесты проходят

--	--



10) добавляем path параметр с ключом petid

11) добавляю новые тесты и проверяю



	<div>12) все тесты</div> <div><div><div>RUN SUMMARY</div><div><div>1</div><div><div>▶ POST Add a new pet to the store7 0</div><div>▶ POST upload an image5 0</div><div>POST Add a new pet to the store Copy0 0</div><div>GET New Request0 0</div></div></div></div><div>положительные</div></div>
Результат	Научился тестировать и пользоваться сниппетами и научулся пользоваться скриптами для проверки запросов