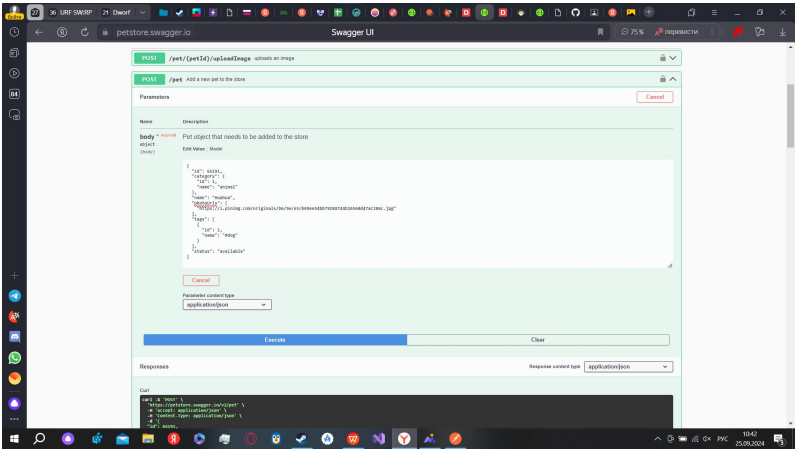
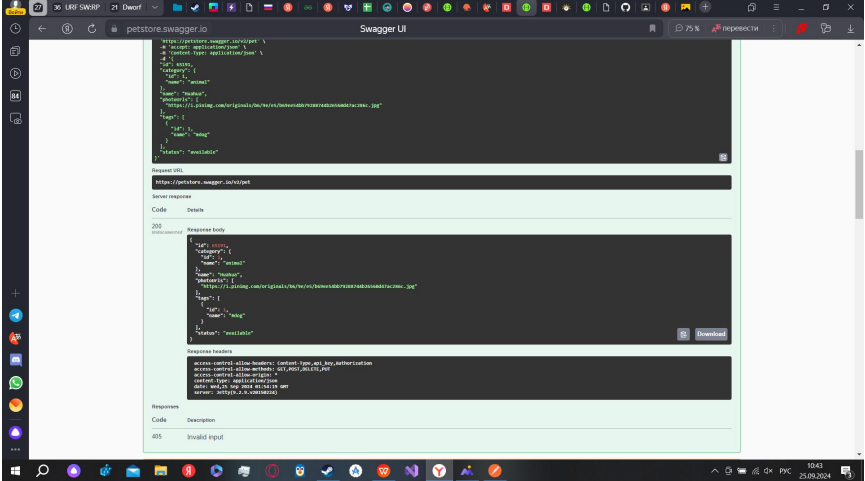
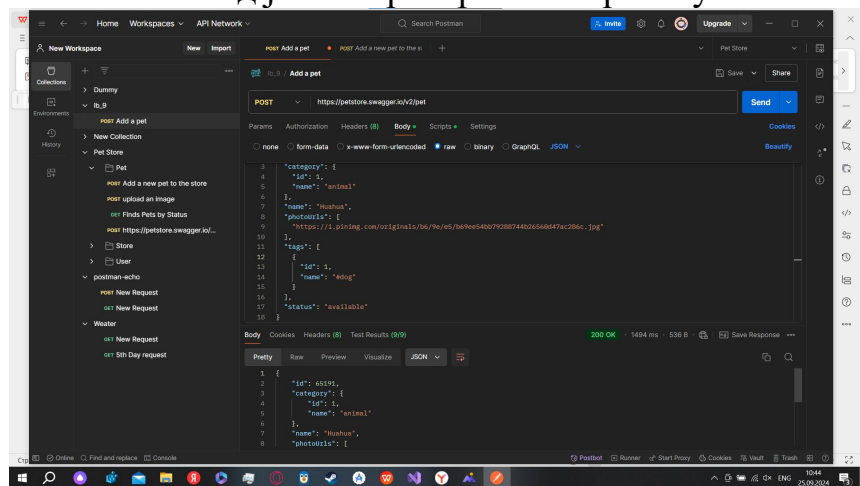


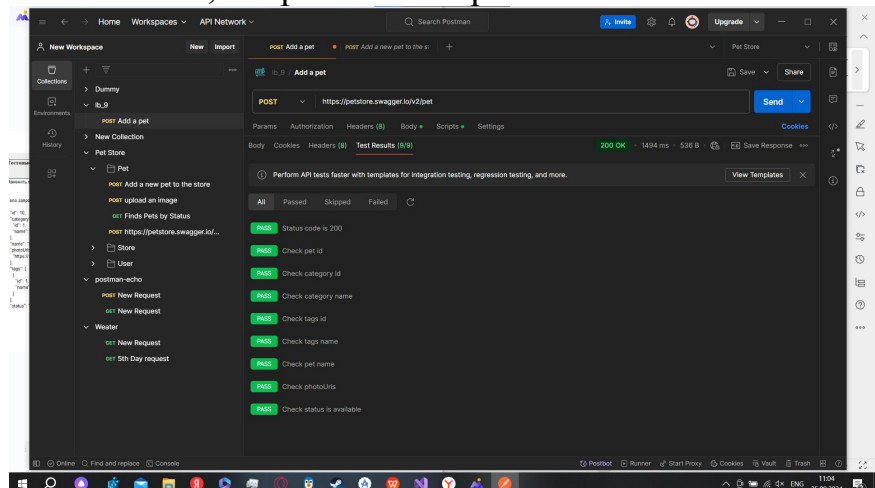
Тема лабораторной работы	lb_9
Выполняющий 1-го тест кейса	Кутенков Константин
Выполняющий 2-го тест кейса	Наглов Виктор
Ход работы	<p>Тест Кейс №1</p> <ol style="list-style-type: none"> 1. Просмотрев таблицу лабораторной, мы приступили к ее выполнению 2. Заходим в swagger и добавляем нвоого питомца, заменив данные на тестовые данные  <p>3. И отправляем</p>  <p>Запрос успешно отправлен и получен код 200</p>

4. Переходим в postman, вставляем полученный на сайте код json и проверяем его работу



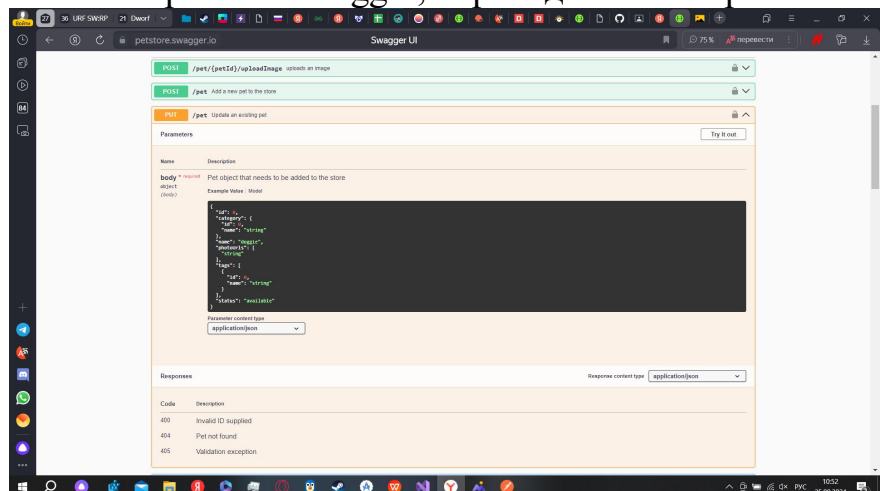
Как видим нам вернулся код 200, значит все в порядке

5. Для проверки создадим авто-тесты на правильность ответа, отправляем запрос



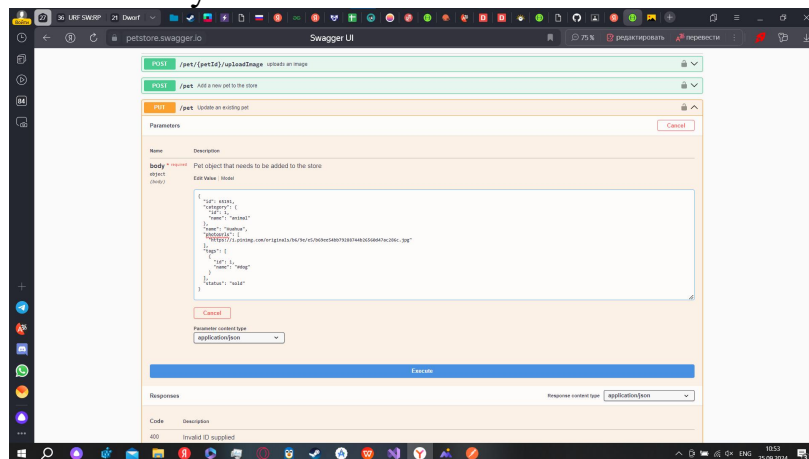
Тест Кейс №2

1. Открываем swagger, переходим к PUT/pet

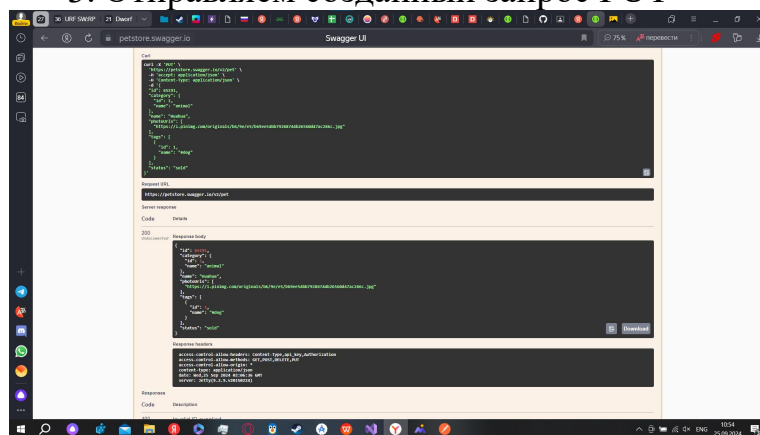


2. Меняю данные в запросе на свои, которые мы

использовали в тест кейсе №1, а также меняем статус на sold

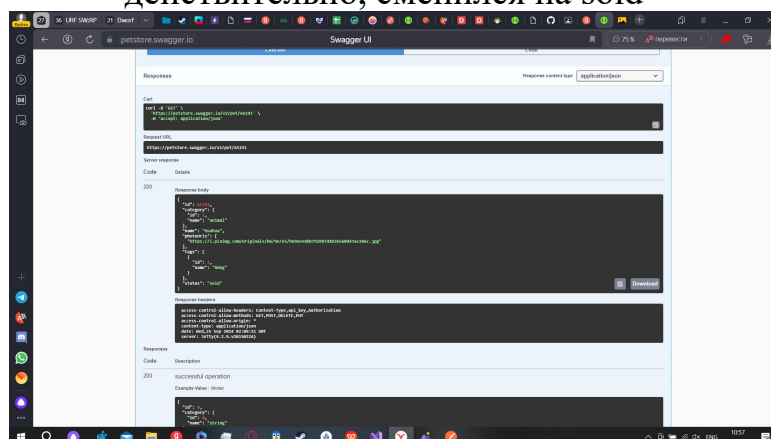


3. Отправляем созданный запрос PUT

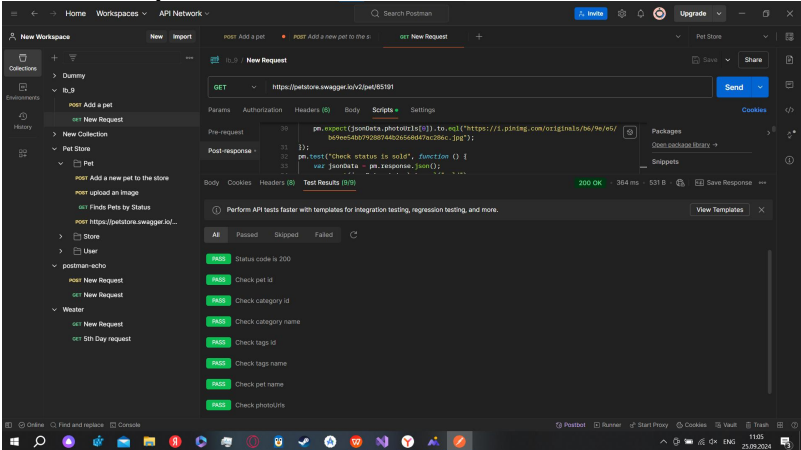


Как видим нам вернулся ответ с кодом 200, следовательно все прошло успешно и наш запрос должен был сменить статус на sold.

4. Отправив, запрос GET /pet{Id}, мы видим что статус действительно, сменился на sold



5. Внимательно присмотревшись к заголовкам ответов:

	<div>Response headers</div> <pre>access-control-allow-headers: Content-Type,api_key,Authorization access-control-allow-methods: GET,POST,DELETE,PUT access-control-allow-origin: * content-type: application/json date: Wed,25 Sep 2024 02:06:36 GMT server: Jetty(9.2.9.v20150224)</pre> <p>Мы видим все практически сходится, за исключением, естественно, наших данных (время, дата)</p> <p>6. Для доп. Проверки создадим авто тесты, как и в первом тест кейсе</p>  <p>Как видим все тесты прошли проверку, следовательно все работает.</p>
Результат	<p>После проделанной нами/мною работы, по итогу рассмотрели:1. Интерфейс 2. Рабочие поля для дальнейшей работы 3. Научились скидывать проекты с github</p>

Программный код тест кейса №1:

Body запроса post:

```
{
  "id": 65191,
  "category": {
    "id": 1,
    "name": "animal"
  },
  "name": "Huahua",
  "photoUrls": [
    "https://i.pinimg.com/originals/b6/9e/e5/b69ee54bb79288744b26560d47ac286c.jpg"
  ],
}
```

```
    "tags": [
      {
        "id": 1,
        "name": "#dog"
      }
    ],
    "status": "available"
  }
}
```

Код тестов вкладки Post Response запроса GET, для проверки:

```
pm.test("Status code is 200", function () {
  pm.response.to.have.status(200);
});

pm.test("Check pet id", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.id).to.eql(65191);
});

pm.test("Check category id", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.category.id).to.eql(1);
});

pm.test("Check category name", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.category.name).to.eql("animal");
});

pm.test("Check tags id", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.tags[0].id).to.eql(1);
});

pm.test("Check tags name", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.tags[0].name).to.eql("#dog");
});

pm.test("Check pet name", function () {
  var jsonData = pm.response.json();
  pm.expect(jsonData.name).to.eql("Huahua");
});

pm.test("Check photoUrls", function () {
```

```

    var jsonData = pm.response.json();

    pm.expect(jsonData.photoUrls[0]).to.eql("https://i.pinimg.com/originals/b6/9e/e5/b69ee54bb79288744b26560d47ac286c.jpg");
  });

  pm.test("Check status is available", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.status).to.eql("available");

  });

```

Программный код тест кейса №2:

Body запроса put

```

{
  "id": 65191,
  "category": {
    "id": 1,
    "name": "animal"
  },
  "name": "Huahua",
  "photoUrls": [
    "https://i.pinimg.com/originals/b6/9e/e5/b69ee54bb79288744b26560d47ac286c.jpg"
  ],
  "tags": [
    {
      "id": 1,
      "name": "#dog"
    }
  ],
  "status": "sold"
}

```

Код тестов вкладки Post Response запроса GET, для проверки:

```

pm.test("Status code is 200", function () {

  pm.response.to.have.status(200);

});

pm.test("Check pet id", function () {

  var jsonData = pm.response.json();

  pm.expect(jsonData.id).to.eql(65191);

```

```
});

pm.test("Check category id", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.category.id).to.eql(1);

});

pm.test("Check category name", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.category.name).to.eql("animal");

});

pm.test("Check tags id", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.tags[0].id).to.eql(1);

});

pm.test("Check tags name", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.tags[0].name).to.eql("#dog");

});

pm.test("Check pet name", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.name).to.eql("Huahua");

});

pm.test("Check photoUrls", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.photoUrls[0]).to.eql("https://i.pinimg.com/originals/b6/9e/e5/b69ee54bb79288744b26560d47ac286c.jpg");

});

pm.test("Check status is sold", function () {

    var jsonData = pm.response.json();

    pm.expect(jsonData.status).to.eql("sold");

});
```