1. Откройте ссылку в браузере https://petstore.swagger.io/
2. Найдите запрос Create user. Создайте нового пользователя, заполнив все поля в запросе. Приложите скриншот с запросом и ответом.

Name	Description

body * required
object
(body)
	Created user object
	
	Edit Value | Model
	
	``json
	{
	   "id": 1654,
	   "username": "sveta",
	   "firstName": "sveta",
	   "lastName": "veselova",
	   "email": "www.veselova@yande
	   "password": "ktnj2020",
	   "phone": "89205673421",
	   "userStatus": 1654
	}
	``
	
	Cancel

```
curl -X 'POST' \
  'https://petstore.swagger.io/v2/user' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
"id": 1654,
"username": "sveta",
"firstName": "sveta",
"lastName": "veselova",
"email": "www.veselova@yandex.ru",
"password": "ktnj2020",
"phone": "89205673421",
"userStatus": 1654
}'
```

Request URL

```
https://petstore.swagger.io/v2/user
```

Server response

Code	Details

200	**Response body**
	``json
	{
	  "code": 200,
	  "type": "unknown",
	  "message": "1654"
	}
	``
	Response headers
	``
	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,27 Oct 2021 06:36:46 GMT
	server: Jetty(9.2.9.v20150224)
	``

Responses

3. Найдите запрос Find Pets by Status. Выполните его с любым параметром. Приложите скриншот с запросом и ответом.

GET **/pet/findByStatus** Finds Pets by status

Multiple status values can be provided with comma separated strings

Parameters

Name	Description
status * required **array[string]** *(query)*	Status values that need to be considered for available pending sold

Execute

Responses

Curl

```
curl -X 'GET' \
  'https://petstore.swagger.io/v2/pet/findByStatus?status=pending&status=sold' \
  -H 'accept: application/json'
```

Request URL

```
https://petstore.swagger.io/v2/pet/findByStatus?status=pending&status=sold
```

Server response

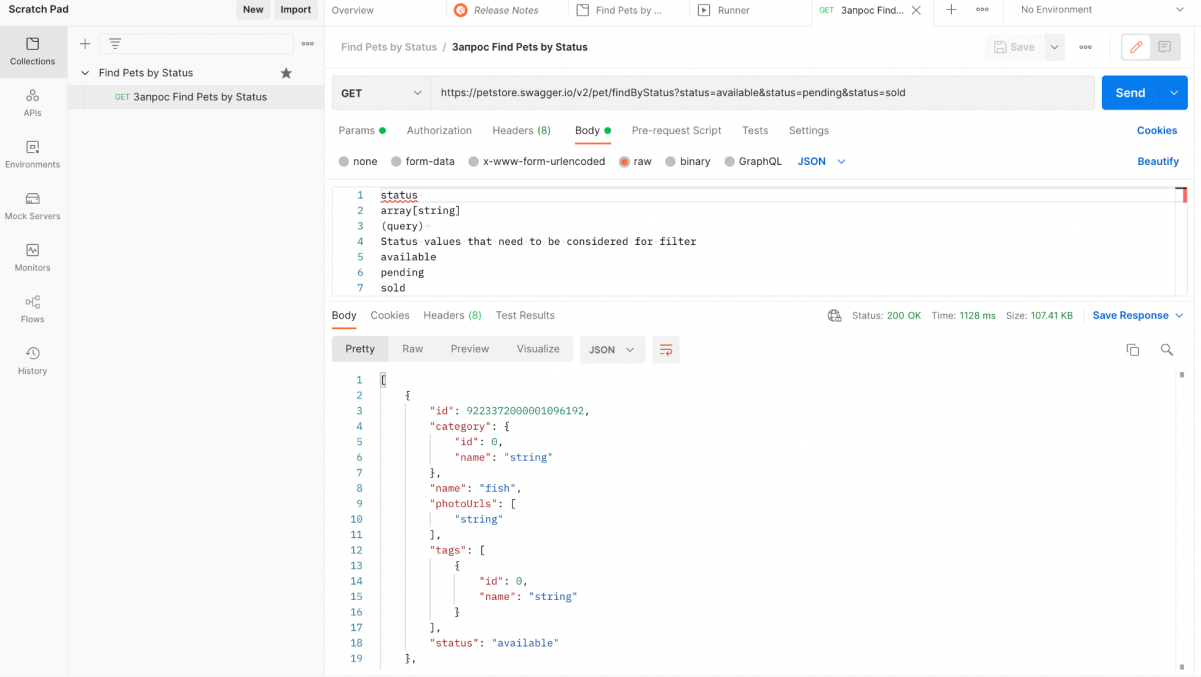| Code	Details
200	**Response body**

```
[
  {
    "id": 5,
    "category": {
      "id": 5,
      "name": "jimmy"
    },
    "name": "jimmy",
    "photoUrls": [],
    "tags": [
      {
        "id": 5,
        "name": "jimmy"
      }
    ],
    "status": "pending"
  },
  {
    "id": 9223372000001094000,
    "category": {
      "id": 0,
      "name": "string"
```

```
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,27 Oct 2021 06:46:49 GMT
server: Jetty(9.2.9.v20150224)
```

Responses

Code	Description
200	successful operation

Example Value | Model

```json
[
  {
    "id": 0,
    "category": {
      "id": 0,
      "name": "string"
    },
    "name": "doggie",
    "photoUrls": [
      "string"
    ],
    "tags": [
      {
        "id": 0,
        "name": "string"
      }
    ],
    "status": "available"
  }
]
```

400	Invalid status value

4. * Создайте запрос Find Pets by Status в Postman (подсказка: тип запроса и ссылку можно найти в сваггере). Выполните негативный тест: передайте в параметр status недопустимое значение. Приложите скриншот с запросом и ответом.

Collections

APIs

Environments

Mock Servers

Monitors

Flows

History

+ ≡   ooo

⌄ Find Pets by Status   ★

GET 3anpoc Find Pets by Status

Find Pets by Status / 3anpoc Find Pets by Status   💾 Save ⌄   ooo   ✎ 💬

GET ⌄   https://petstore.swagger.io/v2/pet/findByStatus?status=available&status=pending&status=sold   Send ⌄

Params ●   Authorization   Headers (8)   Body ●   Pre-request Script   Tests   Settings   Cookies

⬤ none   ⬤ form-data   ⬤ x-www-form-urlencoded   🔴 raw   ⬤ binary   ⬤ GraphQL   JSON ⌄   Beautify

```
1  status
2  array[string]
3  (query)
4  Status values that need to be considered for filter
5  available
6  pending
7  sold
```

Body   Cookies   Headers (8)   Test Results   🌐 Status: 200 OK   Time: 1128 ms   Size: 107.41 KB   Save Response ⌄

Pretty   Raw   Preview   Visualize   JSON ⌄   ⇄   📋 🔍

```
 1  [
 2      {
 3          "id": 9223372000001096192,
 4          "category": {
 5              "id": 0,
 6              "name": "string"
 7          },
 8          "name": "fish",
 9          "photoUrls": [
10              "string"
11          ],
12          "tags": [
13              {
14                  "id": 0,
15                  "name": "string"
16              }
17          ],
18          "status": "available"
19      },
```

```
12537              "string"
12538          ],
12539          "tags": [
12540              {
12541                  "id": 0,
12542                  "name": "string"
12543              }
12544          ],
12545          "status": "available"
12546      },
12547      {
12548          "id": 9223372000001097435,
12549          "category": {
12550              "id": 0,
12551              "name": "string"
12552          },
12553          "name": "doggie",
12554          "photoUrls": [
12555              "string"
12556          ],
12557          "tags": [
12558              {
12559                  "id": 0,
12560                  "name": "string"
12561              }
12562          ],
12563          "status": "available"
12564      }
12565  ]
```