

ID	1
Author	Мінінєвич І.
Status	New
Assigned	Розробник
Severity	1/5
Priority	1/3
Environment	test env https://petstore.swagger.io/
Summary:	Отримано статус відповіді 200 ОК замість 404 Pet not found після оновлення Pet методом PUT з неіснуючим ID
Pre-conditions:	виконати GET запит https://petstore.swagger.io/v2/pet/9484646 Find by ID запит з ID 9484646 і переконатись, що пета з таким ID не існує в системі
Steps	Надіслати запит PUT https://petstore.swagger.io/v2/pet з даним Body: { "id": 9484646, "category": { "id": 2, "name": "Dog" }, "name": "Droopy Dog", "photoUrls": [" https://upload.wikimedia.org/wikipedia/en/thumb/f/fd/Droopy_dog.png/150px-Droopy_dog.png "], "tags": [{ "id": 2, "name": "Beagle" }], "status": "available" }
Actual Result	Статус код відповіді успішний 200 ОК створюється пет з ID 9484646
Expected result	Статус код відповіді 404 Pet not found
Attachments	
ID	2
Author	Мінінєвич І.
Status	New
Assigned	Розробник
Severity	1/5
Priority	1/3
Environment	
Summary:	Статус відповіді 200 ОК після оновлення Pet методом PUT з порожнім значенням ID
Pre-conditions:	

The screenshot displays the Postman interface with two API requests. The top request is a GET request to 'https://petstore.swagger.io/v2/pet/9484646' which returned a 404 Not Found status. The bottom request is a PUT request to 'https://petstore.swagger.io/v2/pet' which returned a 200 OK status. Both requests have their JSON bodies visible in the 'Body' tab.

Top Request: GET Find pet by id 1

- URL: `https://petstore.swagger.io/v2/pet/9484646`
- Method: GET
- Status: 404 Not Found (455 ms)
- Body (JSON):


```
{
  "code": 1,
  "type": "error",
  "message": "Pet not found"
}
```

Bottom Request: PUT Update pet

- URL: `https://petstore.swagger.io/v2/pet`
- Method: PUT
- Status: 200 OK (605 ms)
- Body (JSON):


```
{
  "id": 9223372016908015087,
  "category": {
    "id": 2,
    "name": "Dog"
  },
  "name": "Droopy Dog",
  "photoUrls": [
    "https://upload.wikimedia.org/wikipedia/en/thumb/f/fd/Droopy_dog.png/150px-Droopy_dog.png"
  ],
  "tags": [
    {
      "id": 2,
      "name": "Beagle"
    }
  ],
  "status": ""
}
```


Steps to reproduce	1. Відправити запит PUT https://petstore.swagger.io/v2/pet на зміну значення параметра "NAME" на 1234 як integer з таким BODY: { "id": 1, "category": { "id": 0, "name": "mouse" }, "name": 1234, "photoUrls": ["string"], "tags": [{ "id": 0, "name": "string" }], "status": "available" }
Expected result	Отримано код 405 Validation exception
Actual result	Отримано код 200 ОК з таким BODY відповіді: { "id": 1, "category": { "id": 0, "name": "mouse" }, "name": "1234", "photoUrls": ["string"], "tags": [{ "id": 0, "name": "string" }], "status": "available" } Значення параметра "name" змінено на "1234"
Severity	3 із 5
Priority	2 із 3
Environment	
Version/Build	

ID	4
Author	Мінінєвич І.
Status	New
Assigned	Розробник
Severity	1/5
Priority	1/3
Environment	
Summary:	Отримання коду 200 при оновленні РЕТ за умови змінення значення параметра "NAME" на "null"
Pre-conditions:	"1. Створено Рет з наступними параметрами:""id"": 1, ""category"": { ""id"": 0, ""name"": ""mouse"" }, ""name"": ""miky""

HW API / Update pet

PUThttps://petstore.swagger.io/v2/pet

Send

ParamsAuthHeaders (7)BodyScriptsTestsSettings

rawJSON

200 OK185 ms447 BSave as example

Body

PrettyRawPreviewVisualizeJSON

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

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


PUT	https://petstore.swagger.io/v2/pet					
Pet{						
id	integer(\$int64)		має бути обов'язковим і приймати лише існуючі ID			
category	Category {id integer(\$int64) name string }					
name*	string example: doggie					
photoUrls*	[xml: OrderedMap { "wrapped": true } string xml: OrderedMap { "name": "photoUrl" } xml: name: photoUrl]					
tags	[xml: OrderedMap { "wrapped": true } Tag {...}]					
status	stringpet status in the store Enum: [available, pending, sold]			має бути обов'язковим і приймати лише існуючі статуси		
}						