

Завдання 2

Відкрийте наступний сайт (<https://petstore.swagger.io/#/>) і надішліть кілька запитів усіх типів запитів. Отримайте різні результати – не лише успіхи

pet Everything about your PetsFind out more ^

POST /pet/{petId}/uploadImage uploads an image

POST /pet Add a new pet to the store

Parameters

| Name | Description |
|-----------------|--|
| body * required | Pet object that needs to be added to the store |
| object (body) | Edit Value Model |

```
{  "id": 123,  "name": "Fluffy",  "status": "available"}
```

Cancel

Parameter content typeapplication/json

ExecuteClear

ResponsesResponse content typeapplication/json

Curl

```
curl -X 'POST' \  'https://petstore.swagger.io/v2/pet' \  -H 'accept: application/json' \  -H 'Content-Type: application/json' \  -d '{  "id": 123,  "name": "Fluffy",  "status": "available"  }'
```

Request URLhttps://petstore.swagger.io/v2/pet

Server response

| Code | Details |
|------|--|
| 200 | <div>Undocumented</div> <div>Response body</div> <div><pre>{ "id": 123, "name": "Fluffy", "photoUrls": [], "tags": [], "status": "available"}</pre></div> <div>Download</div> <div>Response headers</div> <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,01 Jan 2025 14:33:51 GMT server: Jetty(9.2.9.v20150224)</pre></div> |

Responses

| Code | Description |
|------|---------------|
| 405 | Invalid input |

pet Everything about your PetsFind out more ^

POST /pet/{petId}/uploadImage uploads an image

POST /pet Add a new pet to the store

Parameters

| Name | Description |
|-----------------|--|
| body * required | Pet object that needs to be added to the store |
| object (body) | Edit Value Model |

```
{  "id": "abc",  "name": 123  }
```

Cancel

Parameter content typeapplication/json

Please correct the following validation errors and try again.

- For 'body' at path 'id': Value must be an integer.
- For 'body' at path 'name': Value must be a string.

ExecuteClear

ResponsesResponse content typeapplication/json

Curl

```
curl -X 'POST' \  'https://petstore.swagger.io/v2/pet' \  -H 'accept: application/json' \  -H 'Content-Type: application/json' \  -d '{  "id": 123,  "name": "Fluffy",  "status": "available"  }'
```

Request URLhttps://petstore.swagger.io/v2/pet

Server response

| Code | Details |
|------|--|
| 200 | <div>Undocumented</div> <div>Response body</div> <div><pre>{ "id": 123, "name": "Fluffy", "photoUrls": [], "tags": [], "status": "available"}</pre></div> <div>Download</div> <div>Response headers</div> <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,01 Jan 2025 14:33:51 GMT server: Jetty(9.2.9.v20150224)</pre></div> |

Responses

| Code | Description |
|------|---------------|
| 405 | Invalid input |

PUT /pet Update an existing pet

Parameters

Cancel

| Name | Description |
|--|--|
| body * required object (body) | Pet object that needs to be added to the store |

Edit Value | Model

```
{
  "id": 123,
  "name": "FluffyUpdated",
  "status": "sold"
}
```

Edit

Parameter content type
application/json

Execute

Responses

Response content type application/json

| Code | Description |
|------|----------------------|
| 400 | Invalid ID supplied |
| 404 | Pet not found |
| 405 | Validation exception |

PUT /pet Update an existing pet

Parameters

Cancel

| Name | Description |
|--|--|
| body * required object (body) | Pet object that needs to be added to the store |

Edit Value | Model

```
{
  "id": @,
  "name": "FluffyUpdated",
  "status": "sold"
}
```

Edit

Parameter content type
application/json

Execute

Clear

Responses

Response content type application/json

Curl

```
curl -X 'PUT' \
  'https://petstore.swagger.io/v2/pet' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 123,
    "name": "FluffyUpdated",
    "status": "sold"
  }'
```

Request URL

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

Server response

| Code | Details |
|----------------------------|--|
| 200 <i>Undocumented</i> | <div><div>Response body</div><div><pre>{ "id": 123, "name": "FluffyUpdated", "photoUrls": [], "tags": [], "status": "sold" }</pre></div><div>Download</div></div> <div><div>Response headers</div><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,01 Jan 2025 15:03:30 GMT server: Jetty(9.2.9.v20150224)</pre></div></div> |

Responses

| Code | Description |
|------|----------------------|
| 400 | Invalid ID supplied |
| 404 | Pet not found |
| 405 | Validation exception |

DELETE

/pet/{petId}

Deletes a pet

Parameters

Cancel

| Name | Description |
|------------------|------------------|
| api_key | |
| string | api_key |
| (header) | |
| petId * required | Pet id to delete |
| integer(int64) | 0 |
| (path) | |

ExecuteClear

Responses

Response content typeapplication/json

Curl

curl -X 'DELETE' \n'https://petstore.swagger.io/v2/pet/0' \n-H 'accept: application/json'

Request URL

https://petstore.swagger.io/v2/pet/0

Server response

| Code | Details |
|------|-------------------------------|
| 404 | Error: response status is 404 |

Response headers

access-control-allow-headers: Content-Type,api_key,Authorization\naccess-control-allow-methods: GET,POST,DELETE,PUT\naccess-control-allow-origin: *\ncontent-length: 0\ndate: Wed,01 Jan 2025 15:09:51 GMT\nserver: Jetty(9.2.9.v20150224)

Responses

| Code | Description |
|------|---------------------|
| 400 | Invalid ID supplied |
| 404 | Pet not found |

DELETE

/pet/{petId}

Deletes a pet

Parameters

Cancel

| Name | Description |
|------------------|------------------|
| api_key | |
| string | api_key |
| (header) | |
| petId * required | Pet id to delete |
| integer(int64) | 123 |
| (path) | |

ExecuteClear

Responses

Response content typeapplication/json

Curl

curl -X 'DELETE' \n'https://petstore.swagger.io/v2/pet/123' \n-H 'accept: application/json'

Request URL

https://petstore.swagger.io/v2/pet/123

Server response

| Code | Details |
|------|---------|
| 200 | |

Undocumented

Response body

{\n "code": 200,\n "type": "unknown",\n "message": "123"\n}

Response headers

access-control-allow-headers: Content-Type,api_key,Authorization\naccess-control-allow-methods: GET,POST,DELETE,PUT\naccess-control-allow-origin: *\ncontent-type: application/json\ndate: Wed,01 Jan 2025 15:08:13 GMT\nserver: Jetty(9.2.9.v20150224)

Responses

| Code | Description |
|------|---------------------|
| 400 | Invalid ID supplied |
| 404 | Pet not found |

GET

/pet/findByStatus Finds Pets by status

Multiple status values can be provided with comma separated strings

Parameters

Cancel

| Name | Description |
|--|---|
| status <small>required</small> array[string] <small>(query)</small> | Status values that need to be considered for filter <div><div>available</div><div>pending</div><div>sold</div></div> |

Execute

Clear

Responses

Response content type application/json

Curl

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

Request URL

https://petstore.swagger.io/v2/pet/findByStatus?status=available

Server response

| Code | Details |
|------|---|
| 200 | <div>Response body</div> <div><pre>{ "id": 9223372036854744000, "category": { "id": 0, "name": "string" }, "name": "fish", "photoUrls": ["string"], "tags": [{ "id": 0, "name": "string" }], "status": "available" }, { "id": 9223372036854744000, "category": { "id": 0, "name": "string" }, "name": "doggie", "photoUrls": ["string"], "tags": [{ "id": 0, "name": "string" }], "status": "available" }]</pre></div> <div>Download</div> |

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, 01 Jan 2025 14:51:42 GMT
server: Jetty(9.2.9.v20150224)
```

Responses

| Code | Description |
|------|--|
| 200 | <div>successful operation</div> <div>Example Value Model</div> <div><pre>{ "id": 0, "category": { "id": 0, "name": "string" }, "name": "doggie", "photoUrls": ["string"], "tags": [{ "id": 0, "name": "string" }], "status": "available" }</pre></div> |
| 400 | Invalid status value |

Завдання 3

Проаналізуйте URL нижче.

[https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL#basics_anatomy_of_a_url)

[US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL#basics_anatomy_of_a_url](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL#basics_anatomy_of_a_url)

1. Чи це URI?
2. Визначте та підпишіть, які компоненти є цієї у URL-адреси
3. Чи наявні всі можливі компоненти URL у прикладі? Якщо ні, то яких не вистачає? Доповніть приклад відсутніми елементами як вони могли б виглядати.

1) Так, цей URL є URI (Uniform Resource Identifier), оскільки URL (Uniform Resource Locator) є одним із типів URI. URL вказує шлях до ресурсу і дозволяє його ідентифікувати.

2) Компоненти URL:

| | |
|-------------------------|---|
| Схема (protocol): | https Визначає протокол доступу до ресурсу (HTTP із шифруванням). |
| Доменне ім'я (host): | developer.mozilla.org Це сервер або домен, на якому розміщено ресурс. |
| Шлях до ресурсу (path): | /en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL Вказує на конкретне розташування файлу або сторінки на сервері. |
| Якір (fragment): | #basics_anatomy_of_a_url Вказує на конкретну частину документа (анкор). |

3) Цей приклад не містить деяких компонентів, які можуть бути у повному URL:

| | |
|---|--|
| Ім'я користувача та пароль (userinfo): | У сучасному використанні це рідкісний компонент. Формат: username:password@. Наприклад: https://user:pass@developer.mozilla.org . |
| Порт (port): | Якщо сервер працює на нестандартному порту (не 80 для HTTP або 443 для HTTPS), порт потрібно вказати. Наприклад: https://developer.mozilla.org:8080 . |
| Параметри запиту (query): | Використовуються для передачі динамічних параметрів у запиті. Формат: ?ключ=значення&ключ2=значення2. Наприклад: https://developer.mozilla.org/en-US/docs?topic=URL . |
| https://user:pass@developer.mozilla.org:8080/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL?topic=URL&lang=en#basics_anatomy_of_a_url | |
| user:pass@ — додає ідентифікацію користувача. :8080 — порт для доступу. ?topic=URL&lang=en — параметри запиту, що передають інформацію про тему і мову. | |