# Swagger Petstore

## 概览

### 描述

This is a sample server Petstore server. You can find out more about Swagger at [http://swagger.io](http://swagger.io) or on [irc.freenode.net, #swagger](http://swagger.io/irc/). For this sample, you can use the api key `special-key` to test the authorization filters.

### 版本

1.0.3

### 联系人

apiteam@swagger.io

### 许可证

Apache 2.0

## 接口列表

### pet

Everything about your Pets

#### Find pet by ID

**GET /pet/{petId}**

##### 描述

Returns a single pet

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | petId\*  必填 | ID of pet to return | integer |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | [Pet](#Pet) |
| 400 | Invalid ID supplied |  |
| 404 | Pet not found |  |

#### Updates a pet in the store with form data

**POST /pet/{petId}**

##### 描述

Updates a pet in the store with form data

##### Produces

* application/json
* application/xml

##### Consumes

* application/x-www-form-urlencoded

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | petId\*  必填 | ID of pet that needs to be updated | integer |
| formData | name | Updated name of the pet | string |
| formData | status | Updated status of the pet | string |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 405 | Invalid input |  |

#### Deletes a pet

**DELETE /pet/{petId}**

##### 描述

Deletes a pet

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| header | api\_key |  | string |
| path | petId\*  必填 | Pet id to delete | integer |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 400 | Invalid ID supplied |  |
| 404 | Pet not found |  |

#### uploads an image

**POST /pet/{petId}/uploadImage**

##### 描述

uploads an image

##### Produces

* application/json

##### Consumes

* multipart/form-data

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | petId\*  必填 | ID of pet to update | integer |
| formData | additionalMetadata | Additional data to pass to server | string |
| formData | file | file to upload | file |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | [ApiResponse](#ApiResponse) |

#### Update an existing pet

**PUT /pet**

##### 描述

Update an existing pet

##### Produces

* application/json
* application/xml

##### Consumes

* application/json
* application/xml

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| body | body\*  必填 | Pet object that needs to be added to the store | [Pet](#Pet) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 400 | Invalid ID supplied |  |
| 404 | Pet not found |  |
| 405 | Validation exception |  |

#### Add a new pet to the store

**POST /pet**

##### 描述

Add a new pet to the store

##### Produces

* application/json
* application/xml

##### Consumes

* application/json
* application/xml

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| body | body\*  必填 | Pet object that needs to be added to the store | [Pet](#Pet) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 405 | Invalid input |  |

#### Finds Pets by status

**GET /pet/findByStatus**

##### 描述

Multiple status values can be provided with comma separated strings

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| query | status\*  必填 | Status values that need to be considered for filter | stringarray(multi) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | <[Pet](#Pet)>array |
| 400 | Invalid status value |  |

#### Finds Pets by tags

**GET /pet/findByTags**

##### 描述

Multiple tags can be provided with comma separated strings. Use tag1, tag2, tag3 for testing.

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| query | tags\*  必填 | Tags to filter by | stringarray(multi) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | <[Pet](#Pet)>array |
| 400 | Invalid tag value |  |

### store

Access to Petstore orders

#### Returns pet inventories by status

**GET /store/inventory**

##### 描述

Returns a map of status codes to quantities

##### Produces

* application/json

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | object |

#### Find purchase order by ID

**GET /store/order/{orderId}**

##### 描述

For valid response try integer IDs with value >= 1 and <= 10. Other values will generated exceptions

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | orderId\*  必填 | ID of pet that needs to be fetched | integer |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | [Order](#Order) |
| 400 | Invalid ID supplied |  |
| 404 | Order not found |  |

#### Delete purchase order by ID

**DELETE /store/order/{orderId}**

##### 描述

For valid response try integer IDs with positive integer value. Negative or non-integer values will generate API errors

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | orderId\*  必填 | ID of the order that needs to be deleted | integer |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 400 | Invalid ID supplied |  |
| 404 | Order not found |  |

#### Place an order for a pet

**POST /store/order**

##### 描述

Place an order for a pet

##### Produces

* application/json
* application/xml

##### Consumes

* application/json

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| body | body\*  必填 | order placed for purchasing the pet | [Order](#Order) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | [Order](#Order) |
| 400 | Invalid Order |  |

### user

Operations about user

#### Get user by user name

**GET /user/{username}**

##### 描述

Get user by user name

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | username\*  必填 | The name that needs to be fetched. Use user1 for testing. | string |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation | [User](#User) |
| 400 | Invalid username supplied |  |
| 404 | User not found |  |

#### Updated user

**PUT /user/{username}**

##### 描述

This can only be done by the logged in user.

##### Produces

* application/json
* application/xml

##### Consumes

* application/json

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | username\*  必填 | name that need to be updated | string |
| body | body\*  必填 | Updated user object | [User](#User) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 400 | Invalid user supplied |  |
| 404 | User not found |  |

#### Delete user

**DELETE /user/{username}**

##### 描述

This can only be done by the logged in user.

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| path | username\*  必填 | The name that needs to be deleted | string |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 400 | Invalid username supplied |  |
| 404 | User not found |  |

#### Logs user into the system

**GET /user/login**

##### 描述

Logs user into the system

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| query | username\*  必填 | The user name for login | string |
| query | password\*  必填 | The password for login in clear text | string |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| 200 | successful operation  **X-Expires-After(string):** date in UTC when token expires  **X-Rate-Limit(integer):** calls per hour allowed by the user | string |
| 400 | Invalid username/password supplied |  |

#### Logs out current logged in user session

**GET /user/logout**

##### 描述

Logs out current logged in user session

##### Produces

* application/json
* application/xml

##### Consumes

无

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| default | successful operation |  |

#### Create user

**POST /user**

##### 描述

This can only be done by the logged in user.

##### Produces

* application/json
* application/xml

##### Consumes

* application/json

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| body | body\*  必填 | Created user object | [User](#User) |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| default | successful operation |  |

#### Creates list of users with given input array

**POST /user/createWithArray**

##### 描述

Creates list of users with given input array

##### Produces

* application/json
* application/xml

##### Consumes

* application/json

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| body | body\*  必填 | List of user object | <[User](#User)>array |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| default | successful operation |  |

#### Creates list of users with given input array

**POST /user/createWithList**

##### 描述

Creates list of users with given input array

##### Produces

* application/json
* application/xml

##### Consumes

* application/json

##### 参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数类型 | 参数名 | 描述 | 数据类型 |
| body | body\*  必填 | List of user object | <[User](#User)>array |

##### 返回值

|  |  |  |
| --- | --- | --- |
| 响应码 | 描述 | 数据类型 |
| default | successful operation |  |

## 数据模型

### Category

|  |  |  |
| --- | --- | --- |
| 属性 | 描述 | 类型 |
| id |  | integer |
| name |  | string |

### Pet

|  |  |  |
| --- | --- | --- |
| 属性 | 描述 | 类型 |
| id |  | integer |
| category |  | <[Category](#Category)> |
| name\* |  | string |
| photoUrls\* |  | array |
| tags |  | <[Tag](#Tag)>array |
| status | pet status in the store | string |

### Tag

|  |  |  |
| --- | --- | --- |
| 属性 | 描述 | 类型 |
| id |  | integer |
| name |  | string |

### ApiResponse

|  |  |  |
| --- | --- | --- |
| 属性 | 描述 | 类型 |
| code |  | integer |
| type |  | string |
| message |  | string |

### Order

|  |  |  |
| --- | --- | --- |
| 属性 | 描述 | 类型 |
| id |  | integer |
| petId |  | integer |
| quantity |  | integer |
| shipDate |  | string |
| status | Order Status | string |
| complete |  | boolean |

### User

|  |  |  |
| --- | --- | --- |
| 属性 | 描述 | 类型 |
| id |  | integer |
| username |  | string |
| firstName |  | string |
| lastName |  | string |
| email |  | string |
| password |  | string |
| phone |  | string |
| userStatus | User Status | integer |

## 关于文档

本文档由poi-tl Word模板引擎(<http://deepoove.com/poi-tl>)生成，poi-tl是一个免费开源的Java类库，你可以非常方便的加入到你的Java项目中。