

# **Swagger Petstore - OpenAPI 3.0**

# Contents

Section	
<b>General Information</b>	<a href="#">Skip</a>
• Terms of Service	<a href="#">Skip</a>
• Contact	<a href="#">Skip</a>
• License	<a href="#">Skip</a>
<b>Servers</b>	<a href="#">Skip</a>
<b>Security</b>	<a href="#">Skip</a>
• API Key Authentication	<a href="#">Skip</a>
•   ◦ Api Key	<a href="#">Skip</a>
• OAuth 2.0 Authorization	<a href="#">Skip</a>
•   ◦ Implicit Flow	<a href="#">Skip</a>
<b>Operations</b>	<a href="#">Skip</a>
• pet	<a href="#">Skip</a>
•   ◦ PUT /pet (updatePet)	<a href="#">Skip</a>
•   ◦   ■ Required Scopes	<a href="#">Skip</a>
•   ◦   ■ Path Parameters	<a href="#">Skip</a>
•   ◦   ■ Query Parameters	<a href="#">Skip</a>
•   ◦   ■ Request Headers	<a href="#">Skip</a>
•   ◦   ■ Cookies	<a href="#">Skip</a>
•   ◦   ■ Request Body	<a href="#">Skip</a>
•   ◦   ■ Responses	<a href="#">Skip</a>
•   ◦   ■   ■ 200 OK	<a href="#">Skip</a>
•   ◦   ■   ■   ■ Response Headers	<a href="#">Skip</a>

Section	
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    ■    ■    400 Bad Request	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    ■    ■    404 Not Found	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    ■    ■    422 Unprocessable Content	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    POST /pet (addPet)	<a href="#">Skip</a>
•    ◦    ■ Required Scopes	<a href="#">Skip</a>
•    ◦    ■ Path Parameters	<a href="#">Skip</a>
•    ◦    ■ Query Parameters	<a href="#">Skip</a>
•    ◦    ■ Request Headers	<a href="#">Skip</a>
•    ◦    ■ Cookies	<a href="#">Skip</a>
•    ◦    ■ Request Body	<a href="#">Skip</a>
•    ◦    ■ Responses	<a href="#">Skip</a>
•    ◦    ■    ■ 200 OK	<a href="#">Skip</a>

Section	
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    ■    ■    400 Bad Request	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    ■    ■    422 Unprocessable Content	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>
•    ◦    GET /pet/findByStatus (findPetsByStatus)	<a href="#">Skip</a>
•    ◦    ■ Required Scopes	<a href="#">Skip</a>
•    ◦    ■ Path Parameters	<a href="#">Skip</a>
•    ◦    ■ Query Parameters	<a href="#">Skip</a>
•    ◦    ■ Request Headers	<a href="#">Skip</a>
•    ◦    ■ Cookies	<a href="#">Skip</a>
•    ◦    ■ Request Body	<a href="#">Skip</a>
•    ◦    ■ Responses	<a href="#">Skip</a>
•    ◦    ■    ■    200 OK	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Headers	<a href="#">Skip</a>
•    ◦    ■    ■    ■ Response Body	<a href="#">Skip</a>

Section	
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ GET /pet/findByTags (findPetsByTags)	<a href="#">Skip</a>
• ○ ■ Required Scopes	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ GET /pet/{petId} (getPetById)	<a href="#">Skip</a>
• ○ ■ Required Scopes	<a href="#">Skip</a>

Section	
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ ■ 404 Not Found	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ POST /pet/{petId} (updatePetWithForm)	<a href="#">Skip</a>
• ○ ■ Required Scopes	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>

Section	
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ DELETE /pet/{petId} (deletePet)	<a href="#">Skip</a>
• ○ ■ Required Scopes	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ POST /pet/{petId}/uploadImage (uploadFile)	<a href="#">Skip</a>
• ○ ■ Required Scopes	<a href="#">Skip</a>

Section	
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• store	<a href="#">Skip</a>
• ○ GET /store/inventory (getInventory)	<a href="#">Skip</a>
• ○ ■ Required Scopes	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>



Section	
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Body</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>POST /store/order (placeOrder)</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Path Parameters</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Query Parameters</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Request Headers</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Cookies</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Request Body</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Responses</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>200 OK</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Headers</li> </ul> </li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Body</li> </ul> </li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>400 Bad Request</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Headers</li> </ul> </li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Body</li> </ul> </li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>422 Unprocessable Content</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Headers</li> </ul> </li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Response Body</li> </ul> </li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>GET /store/order/{orderId} (getOrderById)</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>Path Parameters</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>

Section	
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Query Parameters</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Request Headers</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Cookies</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Request Body</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Responses</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ 200 OK</li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Response Headers</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Response Body</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ 400 Bad Request</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Response Headers</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Response Body</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ 404 Not Found</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Response Headers</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Response Body</li> </ul> </li> </ul> </li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>DELETE /store/order/{orderId} (deleteOrder)</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Path Parameters</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Query Parameters</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Request Headers</li> </ul> </li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li> <ul style="list-style-type: none"> <li>■ Cookies</li> </ul> </li> </ul>	<a href="#">Skip</a>

Section	
<ul style="list-style-type: none"> <li>o ■ Request Body</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Responses</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ 400 Bad Request</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ ■ Response Headers</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ ■ Response Body</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ 404 Not Found</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ ■ Response Headers</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ ■ Response Body</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>user</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o POST /user (createUser)</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Path Parameters</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Query Parameters</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Request Headers</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Cookies</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Request Body</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ Responses</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ default Unknown Status Code</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ ■ Response Headers</li> </ul>	<a href="#">Skip</a>
<ul style="list-style-type: none"> <li>o ■ ■ ■ Response Body</li> </ul>	<a href="#">Skip</a>

Section	
• ○ POST /user/createWithList (createUsersWithListInput)	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ default Unknown Status Code	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ GET /user/login (loginUser)	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>

Section	
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ GET /user/logout (logoutUser)	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ default Unknown Status Code	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ GET /user/{username} (getUserByName)	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>

Section	
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 200 OK	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ 404 Not Found	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ PUT /user/{username} (updateUser)	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>

Section	
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ default Unknown Status Code	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ DELETE /user/{username} (deleteUser)	<a href="#">Skip</a>
• ○ ■ Path Parameters	<a href="#">Skip</a>
• ○ ■ Query Parameters	<a href="#">Skip</a>
• ○ ■ Request Headers	<a href="#">Skip</a>
• ○ ■ Cookies	<a href="#">Skip</a>
• ○ ■ Request Body	<a href="#">Skip</a>
• ○ ■ Responses	<a href="#">Skip</a>
• ○ ■ ■ 400 Bad Request	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>
• ○ ■ ■ 404 Not Found	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Headers	<a href="#">Skip</a>
• ○ ■ ■ ■ Response Body	<a href="#">Skip</a>

# General Information

Swagger Petstore - OpenAPI 3.0 v1.0.11

This is a sample Pet Store Server based on the OpenAPI 3.0 specification. You can find out more about Swagger at <https://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach!

## Terms of Service

Usage of this API is subject to its Terms of Service which you may access at <http://swagger.io/terms/>.

## Contact

- Email: [apiteam@swagger.io](mailto:apiteam@swagger.io)

## License

This API is under an Apache 2.0 license.

Read the full license here: <http://www.apache.org/licenses/LICENSE-2.0.html>

## Servers

The following servers may be utilized for this API:

URL	Description	Variables
<a href="https://petstore3.swagger.io/api/v3">https://petstore3.swagger.io/api/v3</a>	-	-

## Security

The following security schemes are used by this API for authentication and authorization. These must be taken into account in every request unless otherwise stated.

### API Key Authentication

This API is secured through API Keys which you must obtain separately.

#### Api Key

To authenticate requests, include `api_key` as a request header.

```
GET /example
Host: example.com
api_key: SAMPLE-API-KEY-a29@102&3djSF9
```

### OAuth 2.0 Authorization

This API is secured using the [OAuth 2.0 authorization framework](#).

#### Implicit Flow

This API supports the implicit flow.

- Authorization URL: <https://petstore3.swagger.io/oauth/authorize>
- Scopes:



- `write:pets` : modify pets in your account
- `read:pets` : read your pets

# Operations

- [pet](#)
- [store](#)
- [user](#)

## pet

Everything about your Pets

Read more about this category here: <http://swagger.io>

- [PUT /pet \(updatePet\)](#)
- [POST /pet \(addPet\)](#)
- [GET /pet/findByStatus \(findPetsByStatus\)](#)
- [GET /pet/findByTags \(findPetsByTags\)](#)
- [GET /pet/{petId} \(getPetById\)](#)
- [POST /pet/{petId} \(updatePetWithForm\)](#)
- [DELETE /pet/{petId} \(deletePet\)](#)
- [POST /pet/{petId}/uploadImage \(uploadFile\)](#)

### PUT /pet (updatePet)

Update an existing pet

Update an existing pet by Id

#### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

#### Path Parameters

No Path Parameters.

#### Query Parameters

No Query Parameters.

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

Required.

Update an existent pet in the store

Media Type: application/json

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
name	string	Yes	doggie	-

Key	Type	Required	Example	Notes
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

Media Type: `application/x-www-form-urlencoded`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-

Key	Type	Required	Example	Notes
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

## Responses

- [200 OK](#)
- [400 Bad Request](#)
- [404 Not Found](#)
- [422 Unprocessable Content](#)

### 200 OK

Successful operation

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-

Key	Type	Required	Example	Notes
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

Media Type: application/xml

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

#### 400 Bad Request

Invalid ID supplied

##### RESPONSE HEADERS

No Specific Response Headers.

##### RESPONSE BODY

No Response Body

#### 404 Not Found

Pet not found

##### RESPONSE HEADERS

No Specific Response Headers.

##### RESPONSE BODY

No Response Body

#### 422 Unprocessable Content

Validation exception

##### RESPONSE HEADERS

No Specific Response Headers.

##### RESPONSE BODY

No Response Body

## POST /pet (addPet)

Add a new pet to the store

Add a new pet to the store

### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

### Path Parameters

No Path Parameters.

### Query Parameters

No Query Parameters.

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

Required.

Create a new pet in the store

Media Type: application/json

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>• Format: int64</li></ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"><li>• Format: int64</li></ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"><li>• Format: int64</li></ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"><li>• Allowed Values: available, pending, sold</li></ul>

Media Type: application/xml

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

Media Type: `application/x-www-form-urlencoded`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

## Responses

- [200 OK](#)
- [400 Bad Request](#)
- [422 Unprocessable Content](#)

**200 OK**

Successful operation

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"><li>Format: int64</li></ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"><li>Format: int64</li></ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"><li>Allowed Values: available, pending, sold</li></ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"><li>Format: int64</li></ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"><li>Format: int64</li></ul>
tags[*].name	string	No	-	-



Key	Type	Required	Example	Notes
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

#### 400 Bad Request

Invalid input

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

#### 422 Unprocessable Content

Validation exception

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

### GET /pet/findByStatus (findPetsByStatus)

Finds Pets by status

Multiple status values can be provided with comma separated strings

#### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

#### Path Parameters

No Path Parameters.

#### Query Parameters

Key	Type	Required	Example	Notes
status	string	No	-	<ul style="list-style-type: none"> <li>Default: available</li> <li>Allowed Values: available, pending, sold</li> </ul>

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

No Request Body.

#### Responses

- [200 OK](#)
- [400 Bad Request](#)

#### 200 OK

successful operation

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
	array			The root schema is an array.
[*]	object	No	-	-
[*].id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].name	string	Yes	doggie	-
[*].category	object	No	-	-
[*].category.id	integer	No	1	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].category.name	string	No	Dogs	-
[*].photoUrls	array	Yes	-	-
[].photoUrls[]	string	No	-	-
[*].tags	array	No	-	-
[].tags[]	object	No	-	-
[].tags[].id	integer	No	-	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[].tags[].name	string	No	-	-
[*].status	string	No	-	pet status in the store <ul style="list-style-type: none"><li>Allowed Values: available, pending, sold</li></ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
	array			The root schema is an array.
[*]	object	No	-	-
[*].id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].name	string	Yes	doggie	-
[*].category	object	No	-	-
[*].category.id	integer	No	1	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].category.name	string	No	Dogs	-
[*].photoUrls	array	Yes	-	-
[].photoUrls[]	string	No	-	-
[*].tags	array	No	-	-

Key	Type	Required	Example	Notes
[].tags[]	object	No	-	-
[].tags[].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
[].tags[].name	string	No	-	-
[*].status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

#### 400 Bad Request

Invalid status value

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

### GET /pet/findByTags (findPetsByTags)

Finds Pets by tags

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

#### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

#### Path Parameters

No Path Parameters.

#### Query Parameters

Key	Type	Required	Example	Notes
tags	array	No	-	-
tags[*]	string	No	-	-

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

No Request Body.

#### Responses

- [200 OK](#)
- [400 Bad Request](#)

#### 200 OK

successful operation

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
	array			The root schema is an array.
[*]	object	No	-	-
[*].id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].name	string	Yes	doggie	-
[*].category	object	No	-	-
[*].category.id	integer	No	1	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].category.name	string	No	Dogs	-
[*].photoUrls	array	Yes	-	-
[].photoUrls[]	string	No	-	-
[*].tags	array	No	-	-
[].tags[]	object	No	-	-
[].tags[].id	integer	No	-	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[].tags[].name	string	No	-	-
[*].status	string	No	-	pet status in the store <ul style="list-style-type: none"><li>Allowed Values: available, pending, sold</li></ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
	array			The root schema is an array.
[*]	object	No	-	-
[*].id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].name	string	Yes	doggie	-
[*].category	object	No	-	-
[*].category.id	integer	No	1	<ul style="list-style-type: none"><li>Format: int64</li></ul>
[*].category.name	string	No	Dogs	-
[*].photoUrls	array	Yes	-	-
[].photoUrls[]	string	No	-	-
[*].tags	array	No	-	-
[].tags[]	object	No	-	-
[].tags[].id	integer	No	-	<ul style="list-style-type: none"><li>Format: int64</li></ul>

Key	Type	Required	Example	Notes
[].tags[].name	string	No	-	-
[*].status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

#### 400 Bad Request

Invalid tag value

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

### GET /pet/{petId} (getPetById)

Find pet by ID

Returns a single pet

#### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

#### Path Parameters

Key	Type	Required	Example	Notes
petId	integer	Yes	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>

#### Query Parameters

No Query Parameters.

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

No Request Body.

#### Responses

- [200 OK](#)
- [400 Bad Request](#)
- [404 Not Found](#)

#### 200 OK

successful operation

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

Media Type: application/json

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
name	string	Yes	doggie	-
category	object	No	-	-
category.id	integer	No	1	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
category.name	string	No	Dogs	-
photoUrls	array	Yes	-	-
photoUrls[*]	string	No	-	-
tags	array	No	-	-
tags[*]	object	No	-	-
tags[*].id	integer	No	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
tags[*].name	string	No	-	-
status	string	No	-	pet status in the store <ul style="list-style-type: none"> <li>Allowed Values: available, pending, sold</li> </ul>

#### 400 Bad Request

Invalid ID supplied

#### RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

### 404 Not Found

Pet not found

## RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

## POST /pet/{petId} (updatePetWithForm)

Updates a pet in the store with form data

### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

### Path Parameters

Key	Type	Required	Example	Notes
petId	integer	Yes	-	<ul style="list-style-type: none"><li>• Format: int64</li></ul>

### Query Parameters

Key	Type	Required	Example	Notes
name	string	No	-	-
status	string	No	-	-

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

No Request Body.

### Responses

- [400 Bad Request](#)

### 400 Bad Request

Invalid input

## RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

## DELETE /pet/{petId} (deletePet)

Deletes a pet

delete a pet

### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

### Path Parameters

Key	Type	Required	Example	Notes
petId	integer	Yes	-	<ul style="list-style-type: none"><li>• Format: int64</li></ul>

### Query Parameters

No Query Parameters.

### Request Headers

Key	Type	Required	Example	Notes
api_key	string	No	-	-

### Cookies

No Cookies.

### Request Body

No Request Body.

### Responses

- [400 Bad Request](#)

#### 400 Bad Request

Invalid pet value

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

No Response Body

## POST /pet/{petId}/uploadImage (uploadFile)

uploads an image

### Required Scopes

This endpoint requires the following scopes:

- write:pets
- read:pets

### Path Parameters

Key	Type	Required	Example	Notes
petId	integer	Yes	-	<ul style="list-style-type: none"><li>• Format: int64</li></ul>

### Query Parameters



Key	Type	Required	Example	Notes
additionalMetadata	string	No	-	-

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

Media Type: `application/octet-stream`

Type	Example	Notes
string	-	<ul style="list-style-type: none"> <li>Format: binary</li> </ul>

### Responses

- [200 OK](#)

#### 200 OK

successful operation

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
code	integer	No	-	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
type	string	No	-	-
message	string	No	-	-

## store

Access to Petstore orders

Read more about this category here: <http://swagger.io>

- [GET /store/inventory](#) `(getInventory)`
- [POST /store/order](#) `(placeOrder)`
- [GET /store/order/{orderId}](#) `(getOrderById)`
- [DELETE /store/order/{orderId}](#) `(deleteOrder)`

### GET /store/inventory (getInventory)

Returns pet inventories by status

Returns a map of status codes to quantities

### Required Scopes

This endpoint requires the following scopes:

### Path Parameters

No Path Parameters.

**Query Parameters**

No Query Parameters.

**Request Headers**

No Specific Request Headers

**Cookies**

No Cookies.

**Request Body**

No Request Body.

**Responses**

- [200 OK](#)

**200 OK**

successful operation

**RESPONSE HEADERS**

No Specific Response Headers.

**RESPONSE BODY**

Media Type: `application/json`

Key	Type	Required	Example	Notes
{attribute}	integer	No	-	<ul style="list-style-type: none"><li>• Format: int32</li></ul>

**POST /store/order (placeOrder)**

Place an order for a pet

Place a new order in the store

**Path Parameters**

No Path Parameters.

**Query Parameters**

No Query Parameters.

**Request Headers**

No Specific Request Headers

**Cookies**

No Cookies.

**Request Body**

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>• Format: int64</li></ul>
petId	integer	No	198772	<ul style="list-style-type: none"><li>• Format: int64</li></ul>

Key	Type	Required	Example	Notes
quantity	integer	No	7	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
shipDate	string	No	-	<ul style="list-style-type: none"> <li>Format: date-time</li> </ul>
status	string	No	approved	Order Status <ul style="list-style-type: none"> <li>Allowed Values: placed, approved, delivered</li> </ul>
complete	boolean	No	-	-

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
petId	integer	No	198772	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
quantity	integer	No	7	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
shipDate	string	No	-	<ul style="list-style-type: none"> <li>Format: date-time</li> </ul>
status	string	No	approved	Order Status <ul style="list-style-type: none"> <li>Allowed Values: placed, approved, delivered</li> </ul>
complete	boolean	No	-	-

Media Type: `application/x-www-form-urlencoded`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
petId	integer	No	198772	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
quantity	integer	No	7	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
shipDate	string	No	-	<ul style="list-style-type: none"> <li>Format: date-time</li> </ul>
status	string	No	approved	Order Status <ul style="list-style-type: none"> <li>Allowed Values: placed, approved, delivered</li> </ul>
complete	boolean	No	-	-

## Responses

- [200 OK](#)
- [400 Bad Request](#)
- [422 Unprocessable Content](#)

### 200 OK

successful operation

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
petId	integer	No	198772	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
quantity	integer	No	7	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
shipDate	string	No	-	<ul style="list-style-type: none"> <li>Format: date-time</li> </ul>
status	string	No	approved	Order Status <ul style="list-style-type: none"> <li>Allowed Values: placed, approved, delivered</li> </ul>
complete	boolean	No	-	-

#### 400 Bad Request

Invalid input

##### RESPONSE HEADERS

No Specific Response Headers.

##### RESPONSE BODY

No Response Body

#### 422 Unprocessable Content

Validation exception

##### RESPONSE HEADERS

No Specific Response Headers.

##### RESPONSE BODY

No Response Body

#### GET /store/order/{orderId} (getOrderById)

Find purchase order by ID

For valid response try integer IDs with value  $\leq 5$  or  $> 10$ . Other values will generate exceptions.

##### Path Parameters

Key	Type	Required	Example	Notes
orderId	integer	Yes	-	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>

##### Query Parameters

No Query Parameters.

##### Request Headers

No Specific Request Headers

##### Cookies

No Cookies.

##### Request Body

No Request Body.

##### Responses

- [200 OK](#)
- [400 Bad Request](#)
- [404 Not Found](#)

#### 200 OK

successful operation

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
petId	integer	No	198772	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
quantity	integer	No	7	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
shipDate	string	No	-	<ul style="list-style-type: none"> <li>Format: date-time</li> </ul>
status	string	No	approved	Order Status <ul style="list-style-type: none"> <li>Allowed Values: placed, approved, delivered</li> </ul>
complete	boolean	No	-	-

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
petId	integer	No	198772	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
quantity	integer	No	7	<ul style="list-style-type: none"> <li>Format: int32</li> </ul>
shipDate	string	No	-	<ul style="list-style-type: none"> <li>Format: date-time</li> </ul>
status	string	No	approved	Order Status <ul style="list-style-type: none"> <li>Allowed Values: placed, approved, delivered</li> </ul>
complete	boolean	No	-	-

#### 400 Bad Request

Invalid ID supplied

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

#### 404 Not Found

Order not found

#### RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

## DELETE /store/order/{orderId} (deleteOrder)

Delete purchase order by ID

For valid response try integer IDs with value < 1000. Anything above 1000 or nonintegers will generate API errors

### Path Parameters

Key	Type	Required	Example	Notes
orderId	integer	Yes	-	<ul style="list-style-type: none"><li>Format: int64</li></ul>

### Query Parameters

No Query Parameters.

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

No Request Body.

### Responses

- [400 Bad Request](#)
- [404 Not Found](#)

#### 400 Bad Request

Invalid ID supplied

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

No Response Body

#### 404 Not Found

Order not found

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

No Response Body

## user

Operations about user

- [POST /user \(createUser\)](#)
- [POST /user/createWithList \(createUsersWithListInput\)](#)
- [GET /user/login \(loginUser\)](#)
- [GET /user/logout \(logoutUser\)](#)
- [GET /user/{username} \(getUserByName\)](#)
- [PUT /user/{username} \(updateUser\)](#)

- [DELETE /user/{username}](#) (deleteUser).

## POST /user (createUser)

Create user

This can only be done by the logged in user.

### Path Parameters

No Path Parameters.

### Query Parameters

No Query Parameters.

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

Created user object

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"><li>Format: int32</li></ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-

Key	Type	Required	Example	Notes
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

Media Type: `application/x-www-form-urlencoded`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

## Responses

- [default Unknown Status Code](#)

### default Unknown Status Code

successful operation

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

Media Type: `application/xml`



Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

## POST /user/createWithList (createUsersWithListInput)

Creates list of users with given input array

Creates list of users with given input array

### Path Parameters

No Path Parameters.

### Query Parameters

No Query Parameters.

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

Media Type: `application/json`

Key	Type	Required	Example	Notes
	array			The root schema is an array.
[*]	object	No	-	-
[*].id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
[*].username	string	No	theUser	-
[*].firstName	string	No	John	-
[*].lastName	string	No	James	-
[*].email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
[*].password	string	No	12345	-
[*].phone	string	No	12345	-

Key	Type	Required	Example	Notes
[*].userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

## Responses

- [200 OK](#)
- [default Unknown Status Code](#)

### 200 OK

Successful operation

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

### default Unknown Status Code

successful operation

## RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

### GET /user/login (loginUser)

Logs user into the system

#### Path Parameters

No Path Parameters.

#### Query Parameters

Key	Type	Required	Example	Notes
username	string	No	-	-
password	string	No	-	-

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

No Request Body.

#### Responses

- [200 OK](#)
- [400 Bad Request](#)

#### 200 OK

successful operation

#### RESPONSE HEADERS

Key	Type	Required	Example
X-Rate-Limit	integer	No	-
X-Expires-After	string	No	-

#### RESPONSE BODY

Media Type: `application/xml`

Type	Example	Notes
string	-	-

Media Type: `application/json`

Type	Example	Notes
string	-	-

#### 400 Bad Request

Invalid username/password supplied

## RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

### GET /user/logout (logoutUser)

Logs out current logged in user session

#### Path Parameters

No Path Parameters.

#### Query Parameters

No Query Parameters.

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

No Request Body.

#### Responses

- [default Unknown Status Code](#)

#### default Unknown Status Code

successful operation

## RESPONSE HEADERS

No Specific Response Headers.

## RESPONSE BODY

No Response Body

### GET /user/{username} (getUserByName)

Get user by user name

#### Path Parameters

Key	Type	Required	Example	Notes
username	string	Yes	-	-

#### Query Parameters

No Query Parameters.

#### Request Headers

No Specific Request Headers

#### Cookies

No Cookies.

#### Request Body

No Request Body.

## Responses

- [200 OK](#)
- [400 Bad Request](#)
- [404 Not Found](#)

### 200 OK

successful operation

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

Media Type: `application/json`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"><li>Format: int32</li></ul>

Media Type: `application/xml`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"><li>Format: int32</li></ul>

### 400 Bad Request

Invalid username supplied

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

## 404 Not Found

User not found

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

No Response Body

## PUT /user/{username} (updateUser)

Update user

This can only be done by the logged in user.

### Path Parameters

Key	Type	Required	Example	Notes
username	string	Yes	-	-

### Query Parameters

No Query Parameters.

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

Update an existent user in the store

Media Type: application/json

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"><li>Format: int32</li></ul>

Media Type: application/xml

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"><li>Format: int64</li></ul>

Key	Type	Required	Example	Notes
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

Media Type: `application/x-www-form-urlencoded`

Key	Type	Required	Example	Notes
id	integer	No	10	<ul style="list-style-type: none"> <li>Format: int64</li> </ul>
username	string	No	theUser	-
firstName	string	No	John	-
lastName	string	No	James	-
email	string	No	<a href="mailto:john@email.com">john@email.com</a>	-
password	string	No	12345	-
phone	string	No	12345	-
userStatus	integer	No	1	User Status <ul style="list-style-type: none"> <li>Format: int32</li> </ul>

## Responses

- [default Unknown Status Code](#)

### default Unknown Status Code

successful operation

### RESPONSE HEADERS

No Specific Response Headers.

### RESPONSE BODY

No Response Body

## DELETE /user/{username} (deleteUser)

Delete user

This can only be done by the logged in user.

### Path Parameters

Key	Type	Required	Example	Notes
username	string	Yes	-	-

### Query Parameters

No Query Parameters.

### Request Headers

No Specific Request Headers

### Cookies

No Cookies.

### Request Body

No Request Body.

### Responses

- [400 Bad Request](#)
- [404 Not Found](#)

#### 400 Bad Request

Invalid username supplied

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body

#### 404 Not Found

User not found

#### RESPONSE HEADERS

No Specific Response Headers.

#### RESPONSE BODY

No Response Body