HASHIM AZIZ MUHAMMAD HA24010012783

Full Stack Web Development L1T12 - HTTP Requests and Responses

hashimazizm@gmail.com +27832521214

1. When a GET request was sent to `/pet/1`, the response was successful with a status code of 200. This indicates that the operation was completed without any errors.

2. The response code, 200, signifies a successful operation. It means that the server successfully processed the request and returned the desired resource, which in this case is the details of the pet with ID 1.

3. ` The response body included several fields such as "id", "category", "name", "photoUrls", "tags", and "status". These fields provide information about the pet, including its unique identifier ("id"), category details such as ID and name, the pet's name, URLs for photos of the pet, any associated tags, and the current status of the pet ("status").

4. When a POST request was sent to /pet/1, the request body included form data specifying the updated name ("fuzzy") and status ("sold") of the pet with ID 1.

5. The response to the POST request returned the response code, 200, signifying a successful operation. and the message contained the ID of the updated pet ("1").

6. The response code of the POST request was 200. This indicates a successful operation.

7. The /pet/{petId} endpoint is used to update specific details of a pet in the store. In this case, it updates the name and status of the pet identified by the provided ID. On the other hand, the /pet endpoint is used for creating or adding new pets to the store.

8. If the /pet/{petId} endpoint returned a 404 status code, it would mean that the pet with the specified ID was not found in the store. This could happen if the ID provided in the request does not correspond to any existing pet, indicating that the pet either doesn't exist or has been removed from the store.

9. The `/pet/{petId}` endpoint in the Petstore API serves the purpose of deleting a pet from the store. By sending a DELETE request to this endpoint with the appropriate pet ID specified in the path, the corresponding pet will be removed from the store's database. This endpoint is useful for managing the inventory of pets in the store, allowing administrators to remove pets that are no longer available or have been sold. To use this endpoint, one would simply construct a DELETE request with the desired pet ID and send it to the API endpoint. For example, if you wanted to delete a pet with ID 5 from the store, you would send a DELETE request to `/pet/5`. This endpoint helps maintain the accuracy and integrity of the pet store's inventory by allowing for the removal of outdated or irrelevant pet entries.

10. Swagger UI is an advantageous tool for API management. It offers an intuitive interface that simplifies the interaction with APIs for both technical and non-technical users, facilitating the testing of API functions without additional coding or tools. The automatic generation of documentation from the API's codebase ensures that the guidance provided remains current with the API's functionality. This feature enhances development efficiency and fosters clear communication among team members and external stakeholders. Furthermore, Swagger UI aids in the clear visualization of the API's structure, promotes a comprehensive understanding of its features, and assists in the prompt identification of any design issues.