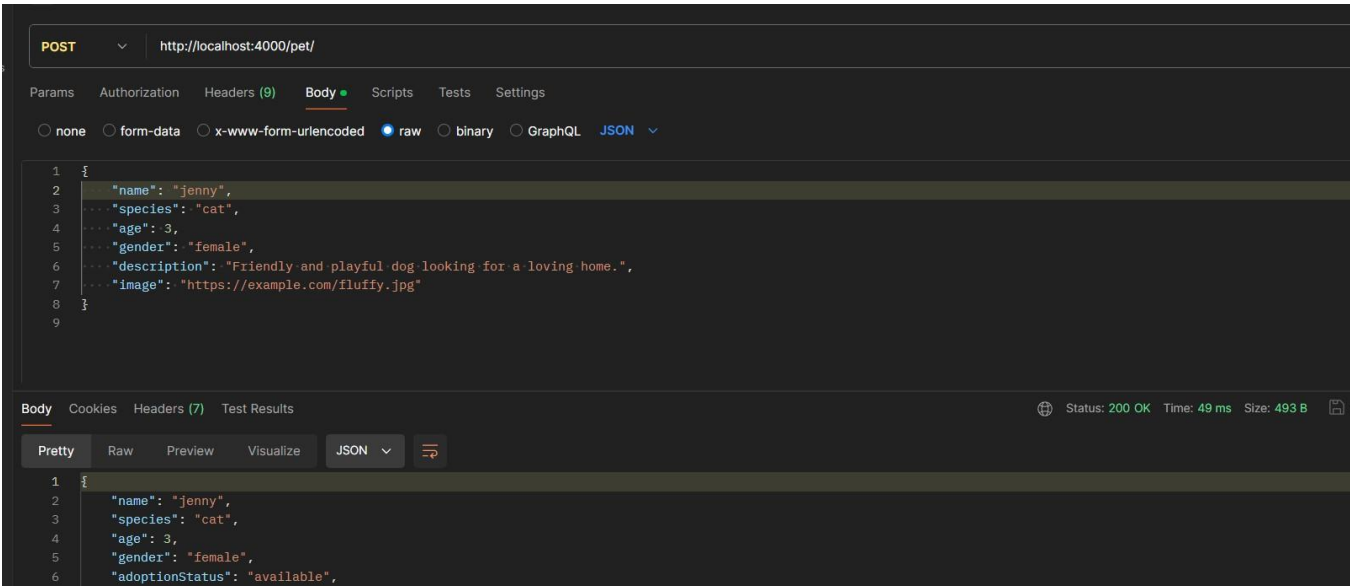
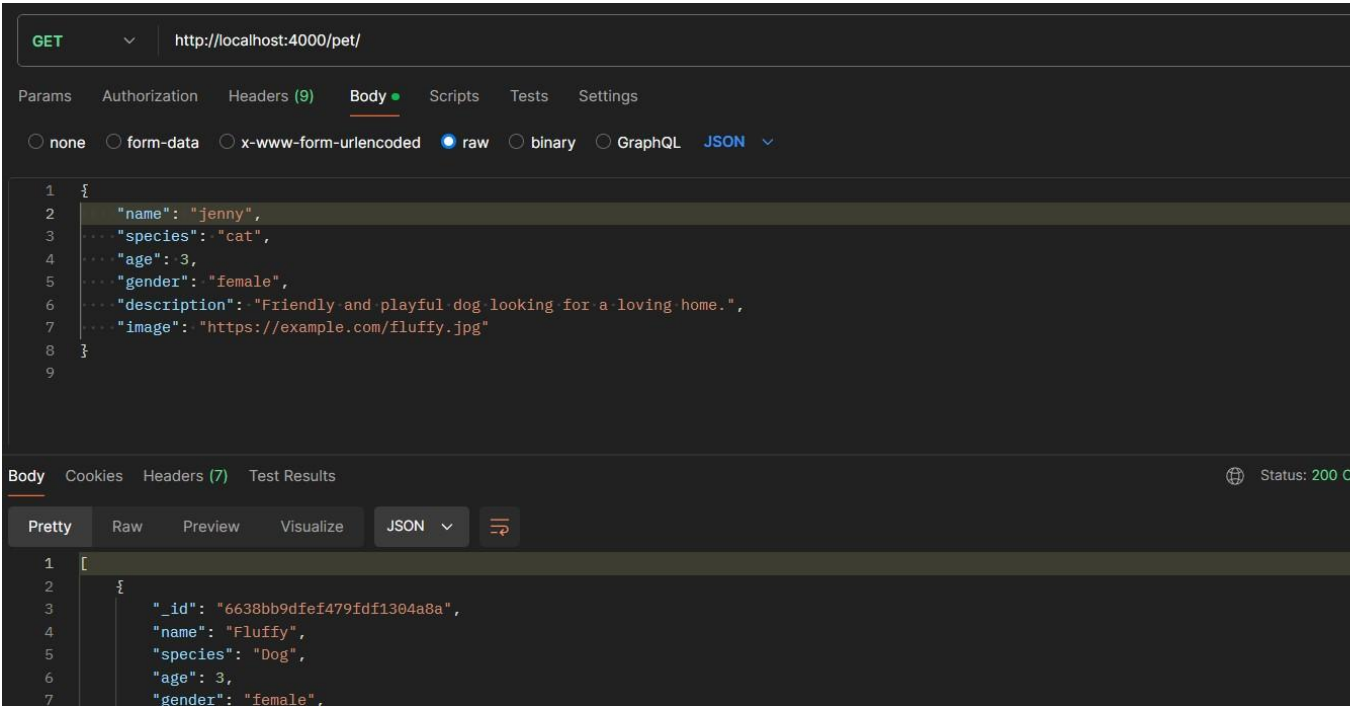


Post-



Get -



Get By Id -

GET

http://localhost:4000/pet/6638bb9dfef479fdf1304a8a

Params

Authorization

Headers (9)

Body

Scripts

Tests

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

```
1 {
2   "name": "jenny",
3   "species": "cat",
4   "age": 3,
5   "gender": "female",
6   "description": "Friendly and playful dog looking for a loving home",
7   "image": "https://example.com/fluffy.jpg"
8 }
9
```

Body

Cookies

Headers (7)

Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1 {
2   "_id": "6638bb9dfef479fdf1304a8a",
3   "name": "Fluffy",
4   "species": "Dog",
5   "age": 3,
6   "gender": "female",
7   "adoptionStatus": "available",

```

Update -

The screenshot shows a REST client interface with a PUT request to `http://localhost:4000/pet/6638bbc3fef479fdf1304a8e`. The 'Body' tab is selected, and the request body is a JSON object. Below the request, the 'Body' tab of the response section is also selected, showing a JSON response with additional fields like `_id`, `adoptionStatus`, and `vaccinated`.

Request:

```
1 {
2   "name": "jenny2",
3   "species": "cat",
4   "age": 3,
5   "gender": "female",
6   "description": "Friendly and playful dog looking for a loving",
7   "image": "https://example.com/fluffy.jpg"
8 }
```

Response:

```
1 {
2   "_id": "6638bbc3fef479fdf1304a8e",
3   "name": "jenny2",
4   "species": "cat",
5   "age": 3,
6   "gender": "female",
7   "adoptionStatus": "available",
8   "vaccinated": false,
```

Delete -

The screenshot displays a REST client interface with a DELETE request configured. The URL is `http://localhost:4000/pet/6638bbc3fef479fdf1304a8e`. The request body is a JSON object representing a pet. The response is also in JSON format, indicating a successful deletion.

DELETE `http://localhost:4000/pet/6638bbc3fef479fdf1304a8e`

Params Authorization Headers (9) **Body** Scripts Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL

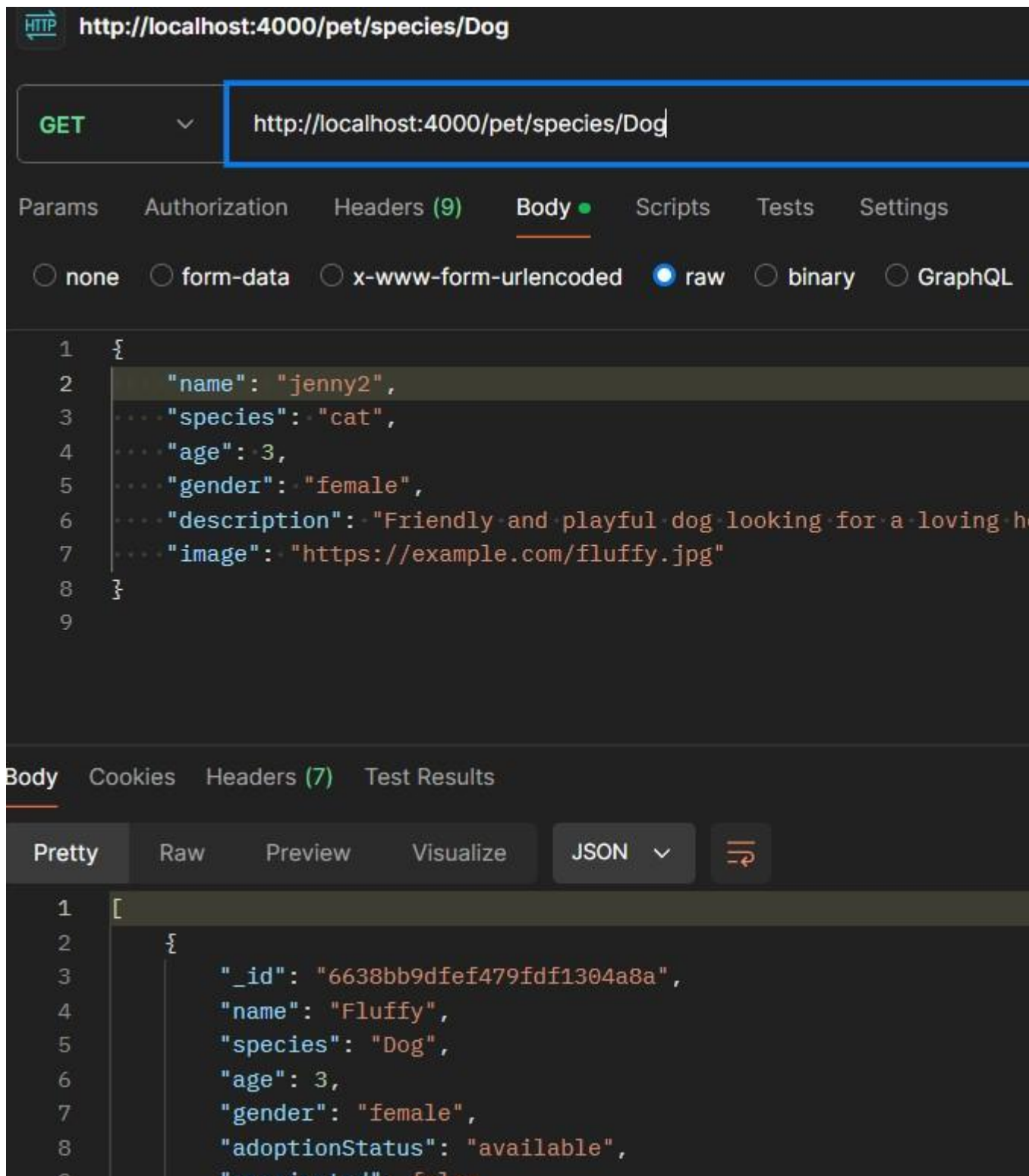
```
1 {
2   "name": "jenny2",
3   "species": "cat",
4   "age": 3,
5   "gender": "female",
6   "description": "Friendly and playful dog looking for a loving",
7   "image": "https://example.com/fluffy.jpg"
8 }
9
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Pet deleted Successfully"
3 }
```

Get by species -



Age -

GET ▼ http://localhost:4000/pet/age/1/3

Params Authorization Headers (9) **Body** ● Scripts Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL

```
1 {
2   "name": "jenny2",
3   "species": "cat",
4   "age": 3,
5   "gender": "female",
6   "description": "Friendly and playful dog looking for a loving
7   "image": "https://example.com/fluffy.jpg"
8 }
9
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼ ↺

```
1 [
2   {
3     "_id": "6638bb9dfef479fdf1304a8a",
4     "name": "Fluffy",
5     "species": "Dog",
6     "age": 3,
7     "gender": "female",
8     "adoptionStatus": "available",
9     "vaccinated": false,
```

Change adoptionStatus -

PUT http://localhost:4000/pet/6638bb9dfef479fdf1304a8a/adopt

Params Authorization Headers (9) **Body** Scripts Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL

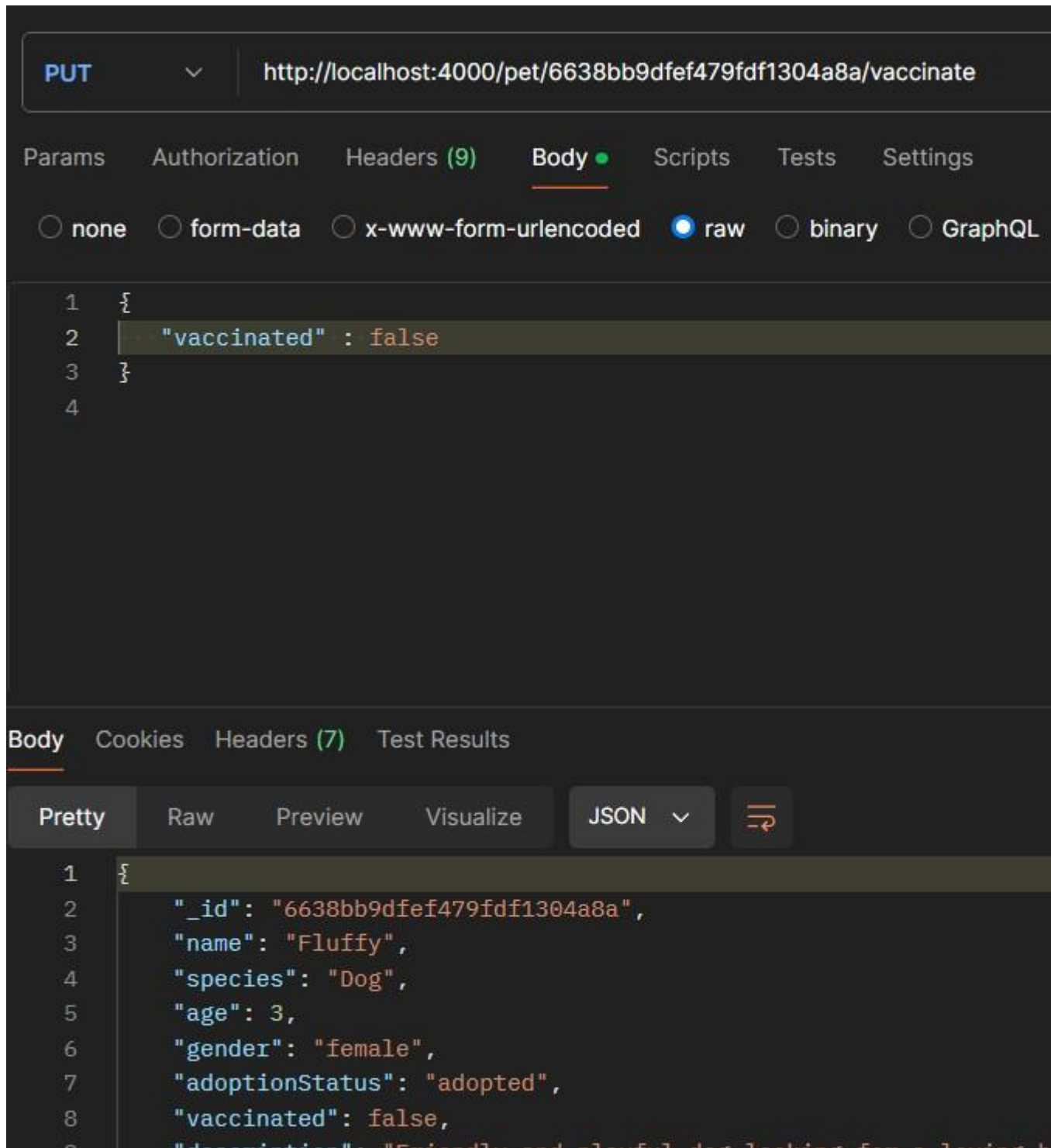
```
1 {
2   "name": "jenny2",
3   "species": "cat",
4   "age": 3,
5   "gender": "female",
6   "description": "Friendly and playful dog looking for a loving",
7   "image": "https://example.com/fluffy.jpg"
8 }
9
```

Body Cookies Headers (7) Test Results

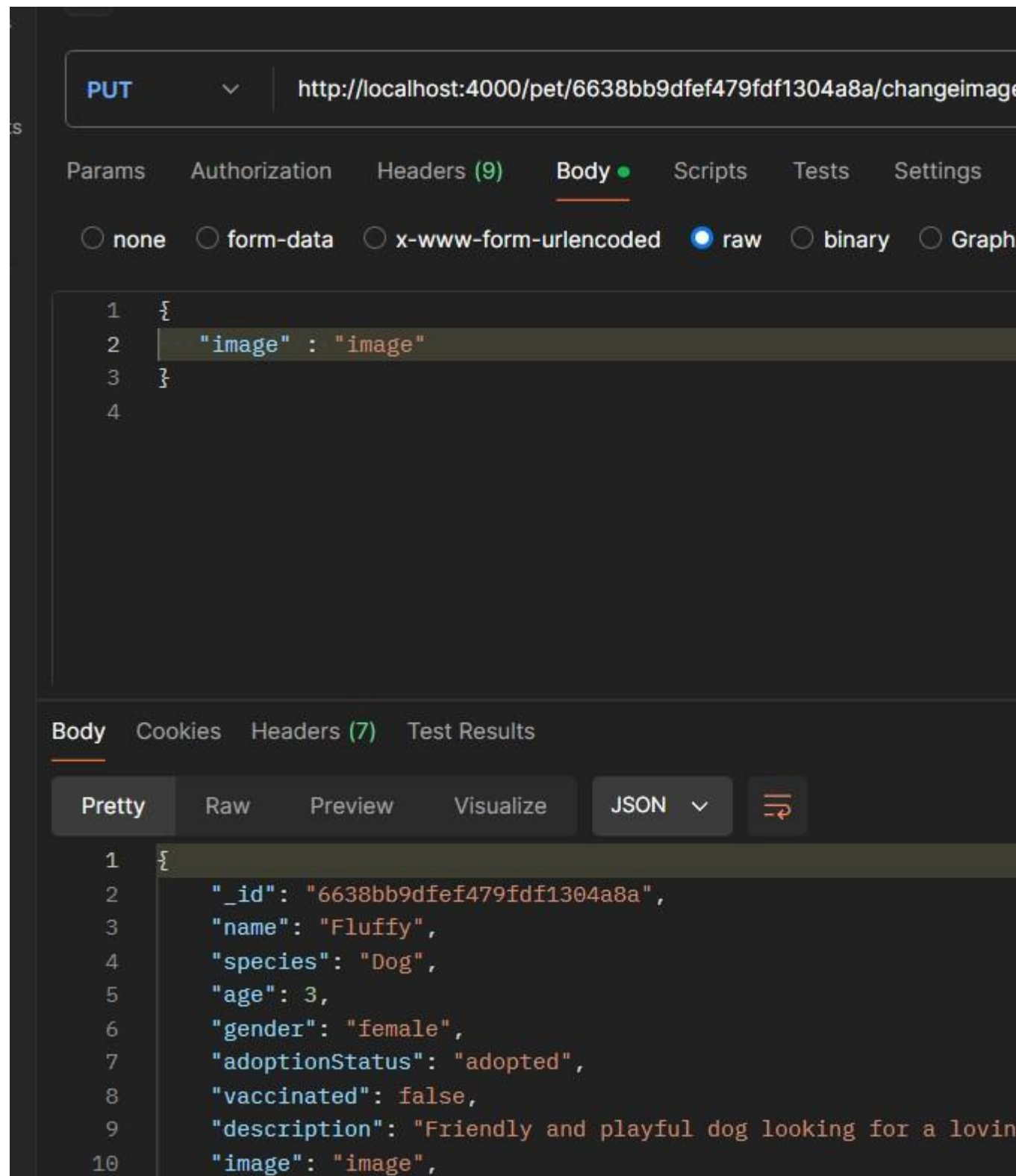
Pretty Raw Preview Visualize JSON

```
1 {
2   "_id": "6638bb9dfef479fdf1304a8a",
3   "name": "Fluffy",
4   "species": "Dog",
5   "age": 3,
6   "gender": "female",
7   "adoptionStatus": "adopted",
8   "vaccinated": false,
9   "description": "Friendly and playful dog looking for a loving"
```

Change vaccinated status -



Update Image Url -



Filter by adoption status -

The screenshot shows a REST client interface with a GET request to `http://localhost:4000/pets/available`. The request body is a JSON object: `{ "image": "image" }`. The response is also in JSON format, showing a list of pet objects. The response body is displayed in the 'Pretty' view, showing a single pet object with various attributes including `_id`, `name`, `species`, `age`, `gender`, `adoptionStatus`, `vaccinated`, and `description`.

GET `http://localhost:4000/pets/available`

Params Authorization Headers (9) **Body** Scripts Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL

```
1 {
2   "image" : "image"
3 }
4
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "_id": "6638bbb5fef479fdf1304a8c",
4     "name": "Tommay",
5     "species": "rabbit",
6     "age": 3,
7     "gender": "female",
8     "adoptionStatus": "available",
9     "vaccinated": false,
10    "description": "Friendly and playful dog looking for a lovi
```

Filter by vaccinated Status -

The screenshot displays a REST client interface. At the top, the method is set to **GET** and the URL is `http://localhost:4000/pets/vaccinated`. Below the URL bar, tabs for **Params**, **Authorization**, **Headers (9)**, **Body**, **Scripts**, **Tests**, and **Settings** are visible. The **Body** tab is selected, showing radio buttons for **none**, **form-data**, **x-www-form-urlencoded**, **raw** (selected), **binary**, and **GraphQL**. The raw body content is a JSON object:

```
{  
  "image" : "image"  
}
```

. The bottom section shows tabs for **body**, **Cookies**, **Headers (7)**, and **Test Results**. The **body** tab is selected, with sub-tabs for **Pretty**, **Raw**, **Preview**, and **Visualize**. The **Pretty** sub-tab is active, and the response is shown as an empty array:

```
[ ]
```

. A **JSON** dropdown and a refresh icon are also present.