

Documentation

Ramen Reviews -Tan Jia Min

Section 1: Steps to run the code:

- a. Navigate to the project folder
- b. In terminal, run "source env/bin/activate"
- c. In terminal, run "pip3 install -r requirements.txt"
- d. In terminal, run "python3 main.py"
- e. Test the API using Postman

Section 2: REST Endpoint URL

Summary

#	Endpoints	Method	Description
1	http://127.0.0.1:5000/reviews	GET	Get a list of all reviews in the database
2	http://127.0.0.1:5000/reviews/<id>	GET	Get review of given id
3	http://127.0.0.1:5000/reviews	POST	Create a new review
4	http://127.0.0.1:5000/reviews/<id>	PUT	Update review of given id
5	http://127.0.0.1:5000/reviews/<id>	DELETE	Delete review of given id
6	http://127.0.0.1:5000/reviews/country/<country>	GET	Get all records that match the country given
7	http://127.0.0.1:5000/reviews/text/<partialtext>	GET	Get all records that contain the partial text in 'Type'

1. Get all reviews:

REQUEST	GET http://127.0.0.1:5000/reviews
RESPONSE	<pre>[{ "Brand": "Brand A", "Country": "IDN", "ID": 1, "Package": "Cup", "Rating": 5.0, "Type": "Seaweed Instant Noodle" }, { "Brand": "Brand B", "Country": "KOR", "ID": 2, "Package": "Cup", "Rating": 3.3, "Type": "Beef Ramen" }, ...]</pre>

2. Get review of given id (e.g. 10)

REQUEST	GET http://127.0.0.1:5000/reviews/10
RESPONSE	<pre>{ "Brand": "Brand C", "Country": "JPN", "ID": 10, "Package": "Cup", "Rating": 4.8, "Type": "Beef and Mushroom Instant Noodle" }</pre>
NOTES	If the id given is invalid, an unsuccessful message is returned.

3. Create a new review

REQUEST	POST http://127.0.0.1:5000/reviews/create
JSON BODY	<pre>{ "Brand": "Brand Z", "Country": "SGP", "Package": "Cup", "Rating": 5.0, "Type": "Yummy Tomyam Ramen" }</pre>
RESPONSE	<pre>{ "Brand": "Brand Z", "Country": "SGP", "ID": 2616, "Package": "Cup", "Rating": 5.0, "Type": "Yummy Tomyam Ramen" }</pre>
NOTES	ID is automatically incremented, and does not need to be specified in the JSON body. The JSON body must contain the Brand, Country, Package, Rating and Type.

4. Update review of given id (e.g. 10)

REQUEST	PUT http://127.0.0.1:5000/reviews/modify/10
JSON BODY	<pre>{ "Country": "USA", "Rating": 5.0 }</pre>

RESPONSE	<pre>{ "Brand": "Brand C", "Country": "USA", "ID": 10, "Package": "Cup", "Rating": 5.0, "Type": "Beef and Mushroom Instant Noodle" }</pre>
NOTES	ID cannot be modified. JSON Body can contain any subset of {Brand, Country, Package, Rating, Type}, depending on what the user wishes to modify. If the id given is invalid, an unsuccessful message is returned.

5. Delete review of given id (e.g. 10)

REQUEST	DELETE http://127.0.0.1:5000/reviews/delete/10
RESPONSE	<pre>{ "status": "Review deleted successfully" }</pre>
NOTES	If the id given is invalid, an unsuccessful message is returned.

6. Get all records that match the country given (e.g. USA)

REQUEST	GET http://127.0.0.1:5000/reviews/country/USA
----------------	---

RESPONSE	<pre> { "Brand": "Brand A", "Country": "USA", "ID": 8, "Package": "Cup", "Rating": 4.0, "Type": "Chicken Instant Noodle" }, { "Brand": "Brand C", "Country": "USA", "ID": 17, "Package": "Cup", "Rating": 2.3, "Type": "Cup Noodles Chicken Vegetable" }, ... </pre>
NOTES	Case sensitive. If there are no reviews that match the given country, a corresponding message will be returned.

7. Get all records that contain the partial text in 'Type' (e.g. Seaweed)

REQUEST	GET http://127.0.0.1:5000/reviews/text/Seaweed
RESPONSE	<pre> { "Brand": "Brand A", "Country": "IDN", "ID": 1, "Package": "Cup", "Rating": 5.0, "Type": "Seaweed Instant Noodle" }, { "Brand": "Brand C", "Country": "TWN", "ID": 123, "Package": "Pack", "Rating": 3.5, "Type": "Noodle Snack Wheat Cracks Seaweed Flavor" }, ... </pre>

Section 3: Design Rationale

- Nouns are used for endpoint paths instead of verbs, as the HTTP methods are already verbs that tell us what the endpoint does.
- Logical nesting is used to show relationships. For example, <http://127.0.0.1:5000/reviews/country/<country>>.
- Implementation accepts and responds with JSON, which is the standard for transferring data.
- The plural form, 'reviews' is used instead of review to avoid ambiguity, and to indicate that there is a collection of reviews.