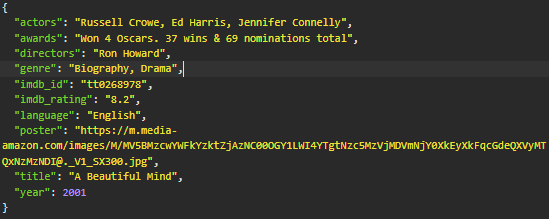
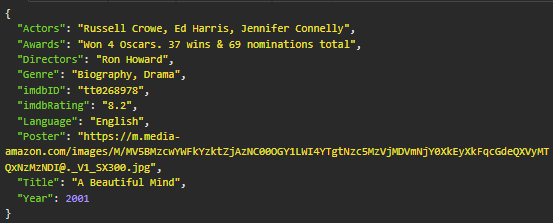
**REST API Documentation**

Below is the REST API documentation for Movie Database API

***JSON fields that the endpoints accept for POST and PUT***

The endpoints accept two different format (OMDb like and the app itself like):



All the key doesn’t need to be there to do a POST or an UPDATE, these are just references on how the key of the JSON should be looked like.

***GET all movies in the database ENDPOINT***

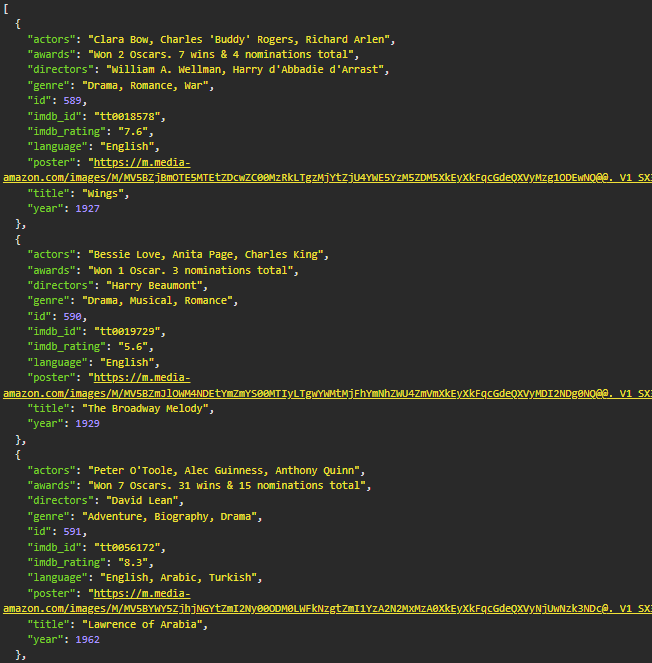
ENDPOINT URL: http://127.0.0.1:8080/api/v1/movies

This endpoint will return all movies in the database, no parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| Empty database | {"error": "there is no movie in the database"} | 404 |

Expect output with response code 200 OK:



***GET specific movie in the database using id ENDPOINT***

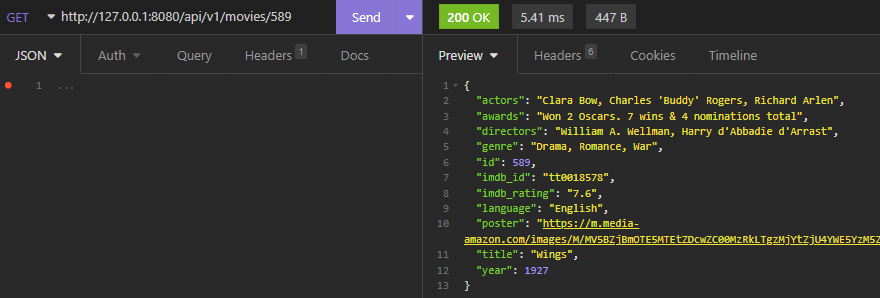
ENDPOINT URL: http://127.0.0.1:8080/api/v1/movies/<id>

This endpoint will return movie with matching in the database, <id> parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No matching movie | {"error":"no movie found for id (<id>)"} | 404 |

Example for the query and expect output with response code 200 OK:



***POST (create) a movie into the database ENDPOINT***

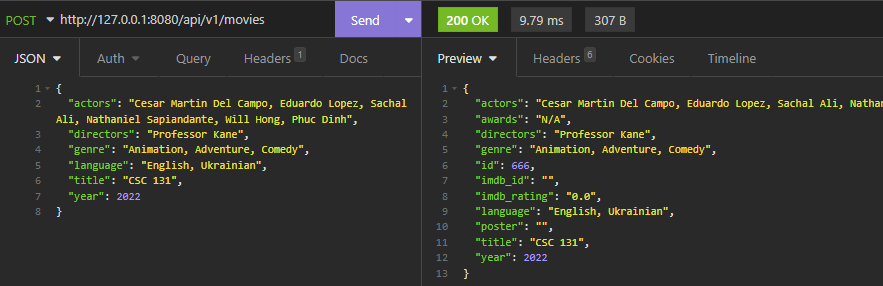
ENDPOINT URL: http://127.0.0.1:8080/api/v1/movies

This endpoint will create a new movie in the database and return to the user. If movie is already existed in the database, return movie information from database. No parameter is needed, but a JSON POST is required.

Error cases:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| Missing required fields | {"error": "malformed request, missing required fields"} | 400 |
| Request is not a JSON | {"error": "request must be json"} | 415 |

Example for the query and expect output with response code 200 OK:



***PUT (update) a movie in the database ENDPOINT***

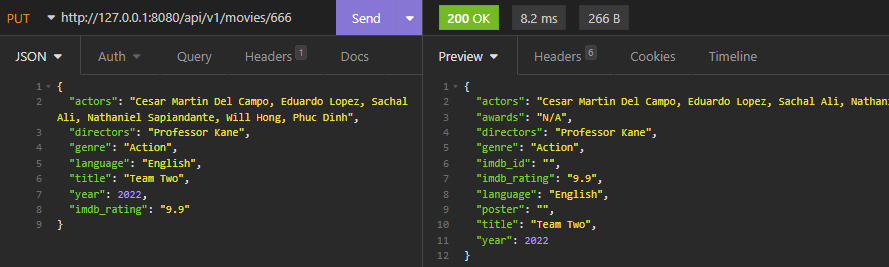
ENDPOINT URL: http://127.0.0.1:8080/api/v1/movies/<id>

This endpoint will update existing movie in the database and return to the user updated information from database. <id> parameter is needed, also a JSON PUT is required.

Error cases:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| Movie doesn’t exist in the database | {"error":"no movie found for id (<id>)"} | 404 |
| Request is not a JSON | {"error": "request must be json"} | 415 |

Example for the query and expect output with response code 200 OK:



***DELETE specific movie in the database using id ENDPOINT***

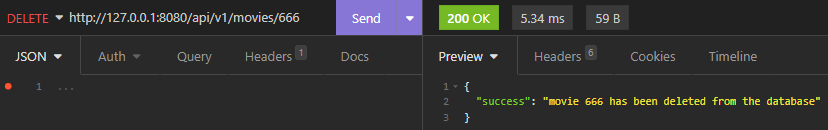
ENDPOINT URL: http://127.0.0.1:8080/api/v1/movies/<id>

This endpoint will delete movie with matching in the database, <id> parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No matching movie | {"error":"no movie found for id (<id>), no deletion is performed"} | 404 |

Example for the query and expect output with response code 200 OK:



***GET all academy awards nominees by year ENDPOINT***

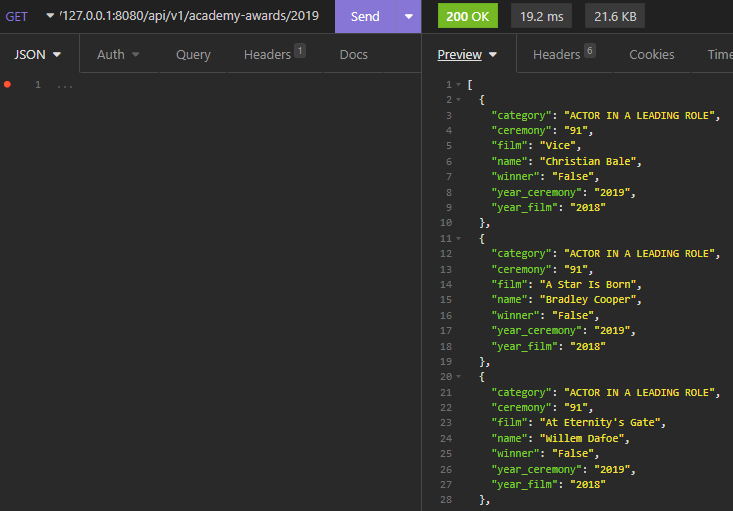
ENDPOINT URL: http://127.0.0.1:8080/api/v1/academy-awards/<year>

This endpoint will return all academy awards nominees by year, <year> parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No data found | {"error":"no data found for this year (<year>)"} | 404 |

Example for the query and expect output with response code 200 OK:



***GET all academy awards Best Picture Winner by year ENDPOINT***

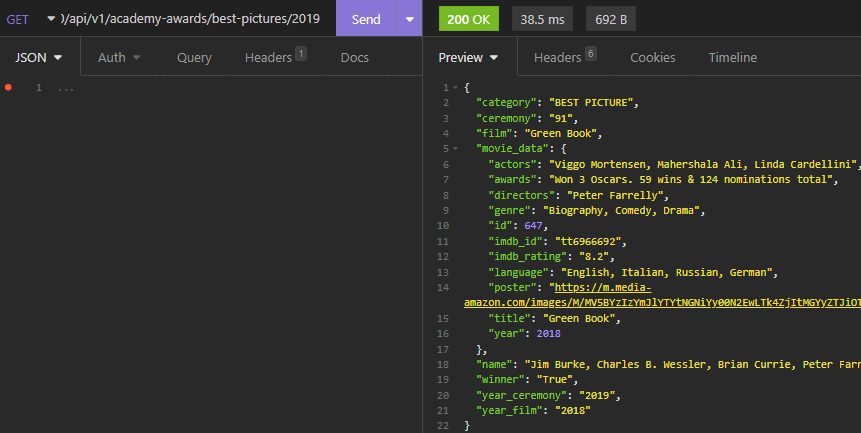
ENDPOINT URL: http://127.0.0.1:8080/api/v1/academy-awards/best-pictures/<year>

This endpoint will return academy awards best picture winner and matching movie by year, <year> parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No data found | {"error":"no data found for best picture in this year (<year>)"} | 404 |

Example for the query and expect output with response code 200 OK:



***GET all academy awards Best Actors Winners by year ENDPOINT***

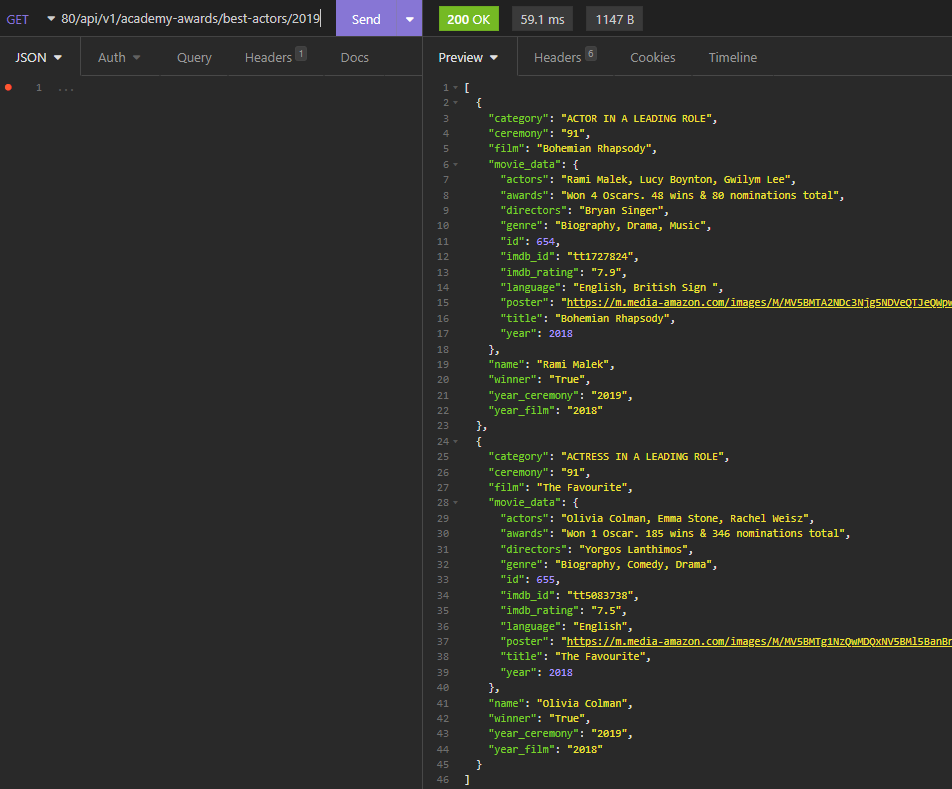
ENDPOINT URL: http://127.0.0.1:8080/api/v1/academy-awards/best-actors/<year>

This endpoint will return academy awards best actors winners and matching movie by year, <year> parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No data found | {"error":"no data found for best actors in this year (<year>)"} | 404 |

Example for the query and expect output with response code 200 OK:



***GET movies recommendation ENDPOINT***

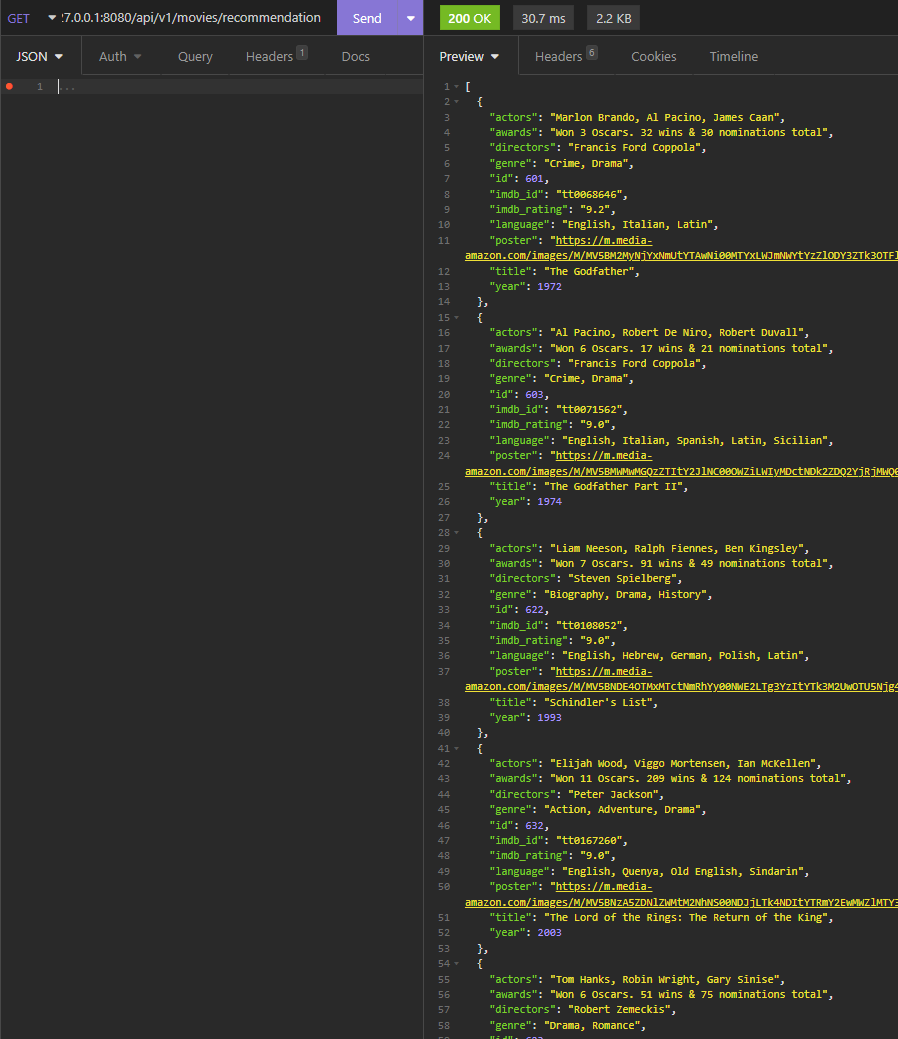
ENDPOINT URL: http://127.0.0.1:8080/api/v1/movies/recommendation

This endpoint will return a list of recommended movies based on past queries, no parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No matching movie to recommend | {"error": "there is no movie to be recommended for now"} | 404 |

Example for the query and expect output with response code 200 OK:



***GET search for specific movie using title ENDPOINT***

ENDPOINT URL: http://127.0.0.1:8080/api/v1/search?title=<title>

This endpoint will search the local database for movie using title, if no movie is found in the local database, continue to search in the OMDb and import to the database. <title> parameter is needed.

Error case:

|  |  |  |
| --- | --- | --- |
|  | Response | Response Status Code |
| No title parameter in the request | {"error": "malformed request, no title parameter is found"} | 400 |

Example for the query and expect output with response code 200 OK:

