Music Library

Learning objective: Build a REST web API in Django REST Framework and test each endpoint in Postman.

Technologies: Python, Django REST Framework, Postman

Total points: /55 -> /5 weighted

(5 points): As a developer, I want to make good, consistent commits.

(2.5 points) As a developer, I want to create a Song model

Property names must be in snake_case and match the following exactly!

- title - CharField
- artist - CharField
- album - CharField
- release_date - DateField
- genre - CharField

(2.5 points) As a developer, I want my API to serve content on the following urls paths:

Paths must match these exactly!

- '127.0.0.1:8000/music/'
- '127.0.0.1:8000/music/<int:pk>/'

(15 points) As a developer, I want to build a REST web API in Django REST Framework, so that I can make HTTP requests interact with the data set.

(5 points) As a developer, I want to create a GET endpoint the responds with a 200 success status code and all of the songs within the Music table.

(5 points) As a developer, I want to create a GET by id endpoint that does the following things:

- Accepts a value from the request's URL (The id of the song to retrieve).
- Returns a 200 status code.
- Responds with the song in the database that has the id that was sent through the URL.

(5 points) As a developer, I want to create a POST endpoint that does the following things:

- Accepts a body object from the request in the form of a Song model.
- Adds the new song to the database.
- Returns a 201 status code.
- Responds with the newly created song object.

(5 points) As a developer, I want to create a PUT endpoint that does the following things:

- Accepts a value from the request's URL (The id of the song to be updated).
- Accepts a body object from the request in the form of a Song model.

- Finds the song in the Music table and updates that song with the properties that were sent in the request’s body.
- Returns a 200 status code.
- Responds with the newly updated song object.

(5 points) As a developer, I want to create a DELETE endpoint that does the following things:

- Accepts a value from the request’s URL.
- Returns a 200 status code.
- Responds with the song in the database that has just been deleted.

(5 points) As a developer, I want to use Postman to make a POST, PUT, DELETE, and both GET requests (get by id and get all) request to my REST web API, save it to a collection, and then export it as a JSON from Postman.

BONUS:

(5 points) As a developer, I want to add the ability to “like” a song through the web API and have the number of likes saved in the database with the song.