

Assignment: Build an aggregated Marvel characters service with Finagle and Marvel API

Objective:

Create a Finagle-based REST API that integrates with Marvel API to fetch and aggregate data. The service should implement a caching or database mechanism to store frequently requested data.

Requirements:

Create a single API Endpoint:

Endpoint: GET /marvel-characters?name={name}&comicYear={year}&orderBy={order}

Query Parameters:

- name: The name of the Marvel character (required).
- comicYear: The year of the comics to filter (optional, defaults to the last 10 years).
- orderBy: The order in which the comics should be returned (e.g., title(default),

issueNumber).

The service should call and aggregate data from two Marvel API endpoints:

- Character Endpoint: This fetches character information based on the character's name.

Endpoint: /v1/public/characters?name={name}

- Comics Endpoint: This fetches comics based on the character's ID and optional filters such as year and order.

Endpoint: /v1/public/characters/{characterId}/comics?startYear={year}&orderBy={order}

Expected Response:

The response should include:

Character data: Name, description, and thumbnail image.

Comics data: List of comics the character appears in, with details like title, issue number, and release year.

Caching or Database:

Implement a caching mechanism to store the aggregated results for character and comic queries based on the parameters (name, comicYear, and orderBy). Alternatively, use a database to store the results and implement expiration after a set time (e.g., 10 minutes).

Bonus Tasks (Optional):

Handle pagination for the comics list returned by the Marvel API and aggregate results across multiple pages.

Marvel API Documentation:

https://developer.marvel.com/docs#!/public/getCreatorCollection_get_0

Please ensure that you develop your API to be as production-ready as possible.

Happy coding@