Backend Hiring Task

Deadline : 11:59 pm IST - 02nd February, 2024

Optimal time to learn, solve and code : 2 - 4 hours

Problem Statement

You need to create a sample backend application in FastAPI, Python and MongoDB. The challenge assumes you have basic knowledge of Python and some knowledge of Flask / Django / FastAPI as well as MongoDB.

Brief

You are building an ecommerce application like Flipkart/Amazon. You need to build the following APIs -

1. API to List all available products in the system. You can create some 10-20 dummy products like TV, laptop, etc for reference.
    a. Each product should have these attributes -
        i. Product ID – Default _id (ObjectID) of MongoDB
        ii. Product name
        iii. Product price
        iv. Product available quantity
    b. This API should have pagination enabled using limit/offset query parameters.
    c. Should have filters for *min_price*, *max_price* using query params
    d. This API should return the data in following format -
        i. *data*: A list of records matching filters, if present. Each record should have -
            1. id
            2. name
            3. price
            4. quantity
        ii. page: An object defining metadata
            1. limit: current limit of records
            2. nextOffset: if more records are present
            3. prevOffset: if previous records are present
            4. total: total number of records
    e. Bonus points if the above response object can be created using 1 single DB query. Hint: check Aggregation Pipelines and Facets in MongoDB.

2. API to Create a new order. Each order should have these properties -
    a. createdOn – auto generated, client should not send this.
    b. Items – **list** of items bought in the Order. Each record in this array would have these properties -

        i. productId
        ii. boughtQuantity
    c. Total amount
    d. User Address – nested object having these properties -
        i. City
        ii. Country
        iii. Zip Code

Tech Stack Allowed

- Python - **FastAPI** - Mandatory to be python 3 (3.10 or above)
- Use **MongoDB** as a database with Pymongo or Motor.

How will we Judge

- Code completeness
    - Should be able to test by running
- Code clarity
    - Cleanliness
    - Well Formatted
    - Documentation – README with sample API Calls
- Structure of APIs created
    - Endpoint and REST conventions followed.
- Structure of MongoDB Collections / Database models / Queries
    - Data models - how you are storing the information in MongoDB or your database.
    - Your queries to MongoDB - how optimised they are (could be fairly good, not the best but certainly not very very performant hit)
    - Structuring relationships and lookups/joins in MongoD