Python API Developer Challenge:

# Description:

We have one file with the name books.csv. The CSV file contains the general information about the books, like book name, author, year of publication, etc. We want you to develop two API endpoints, where you can see the requirements below:

# API Endpoints:

1. API will return number of rows requested from the books.csv file.

* **Input** : rows=3
* **Output:** (JSON)

{

    "books": [

        {

            "id": 7,

            "title": "Introduction to African Oral Literature and Performance",

            "author": "Bayo Ogunjimi",

            "authors":"Bayo Ogunjimi, Abdul Rasheed Na\'allah",

            "isbn13": 9781592211517,

            "isbn10": "1592211518",

            "price": "$23.95",

            "publisher": "Africa World Press",

            "pubyear": 2006,

            "subjects": "Africa - Anthropology & Sociology, African Folklore & Mythology, Oral Tradition & Storytelling, General & Miscellaneous African Literature - Literary Criticism, African Literature Anthologies, Fables, Fairy Tales, & Folk Tales - Literary Criticism",

            "lexile": null,

            "pages": 146.0,

            "dimensions": "8.30 (w) x 5.30 (h) x 0.80 (d)"

        },

        {

            "id": 474,

            "title": "Out on the Porch: An Evocation in Words and Pictures",

            "author": "Reynolds Price",

            "authors": "Reynolds Price, Clifton Dowell, Reynolds Price",

            "isbn13": 9780945575931,

            "isbn10": "945575939",

            "price": "$1.99",

            "publisher": "Algonquin Books of Chapel Hill",

            "pubyear": 1992,

            "subjects": "Outdoor & Recreational Areas, American Literature Anthologies",

            "lexile": null,

            "pages": 128.0,

            "dimensions": "8.36 (w) x 8.28 (h) x 0.50 (d)"

        },

        {

            "id": 424,

            "title": "Friction, Volume 7: Best Gay Erotic Fiction",

            "author": "Jesse Grant",

            "authors": "Jesse Grant",

            "isbn13": 9781555838270,

            "isbn10": "1555838278",

            "price": "$1.99",

            "publisher": "Alyson Books",

            "pubyear": 2004,

            "subjects": "Fiction, American Literature Anthologies, Anthologies, Gay & Lesbian Studies, Fiction Subjects",

            "lexile": null,

            "pages": 368.0,

            "dimensions": "5.40 (w) x 8.50 (h) x 0.80 (d)"

        }

    ]

}

1. 2nd API will give freedom to the user to filter and see any data from the file. The user could only able to filter from the given column list. If a column is not present then a graceful error message should return. Even if the API didn’t find any filter response then the user should get a empty response.

* **Input**: Ex. (JSON)
  + {"authors":"Jesse Grant"}
* **Output**: (JSON)
* {
* "books": [
* {
* "id": 424,
* "title": "Friction, Volume 7: Best Gay Erotic Fiction",
* "author": "Jesse Grant",
* "authors": "Jesse Grant",
* "isbn13": 9781555838270,
* "isbn10": "1555838278",
* "price": "$1.99",
* "publisher": "Alyson Books",
* "pubyear": 2004,
* "subjects": "Fiction, American Literature Anthologies, Anthologies, Gay & Lesbian Studies, Fiction Subjects",
* "lexile": null,
* "pages": 368.0,
* "dimensions": "5.40 (w) x 8.50 (h) x 0.80 (d)"
* }
* ]
* }

## Requirements/Deliveries:

* Python 3+ (Implementation should be done using Python3 and above)
* Framework (Flask)
* Dockerisation of the complete API
* Git (We are expecting full git logs for this exercise so that all steps of how you developed the project is visible)
* Unit Tests (the code developed should include unit tests)
* Requirements.txt (a file containing all necessary libraries to run the project)
* README.md (a basic description of how to build and run the project)