Backend Assignment

Create REST APIs using **Python** (Flask, Django, any other web framework of your choice) for managing the user's data. You can use database(i.e SQL, NOSQL) of your choice to store the data. Take sample data from here.

User should have the following attributes:-

- ID
- First Name
- Last Name
- Company Name
- Age
- City
- State
- Zip
- Email
- Web

An Application should have the following endpoints:-

- **/api/users - GET** - To list the users
    a. Response with HTTP status code **200** on success

```
[
    {
        "id": 1,
        "first_name": "James",
        "last_name": "Butt",
        "company_name": "Benton, John B Jr",
        "city": "New Orleans",
        "state": "LA",
        "zip": 70116,
        "email": "jbutt@gmail.com",
        "web": "http://www.bentonjohnbjr.com",
        "age": 70
    },
    {
        "id": 2,
        "first_name": "Josephine",
        "last_name": "Darakjy",
        "company_name": "Chanay, Jeffrey A Esq",
        "city": "Brighton",
        "state": "MI",
        "zip": 48116,
        "email": "josephine_darakjy@darakjy.org",
        "web": "http://www.chanayjeffreyaesq.com",
        "age": 48
    }
]
```

b. Also, supports some query parameters:-
   i. page - a number for pagination
   ii. limit - no. of items to be returned, default limit is 5
   iii. name - search user by name as a substring in First Name or Last Name (Note, use substring matching algorithm/pattern to match the name). It should be case-insensitive.
   iv. Sort - name of attribute, the items to be sorted. By default it returns items in ascending order if this parameter exist, and if the value of parameter is prefixed with **'-'** character, then it should return items in descending order

Sample query endpoint:- ***`/api/users?page=1&limit=10&name=James&sort=-age`***
This endpoint should return list of 10 users whose first name or last name contains substring given name and sort the users by age in descending order of page 1.

- **/api/users - POST** - To create a new user
  a. Request Payload should be like in json format :-

```
{
    "id": 2,
    "first_name": "Josephine",
    "last_name": "Darakjy",
    "company_name": "Chanay, Jeffrey A Esq",
    "city": "Brighton",
    "state": "MI",
    "zip": 48116,
    "email": "josephine_darakjy@darakjy.org",
    "web": "http://www.chanayjeffreyaesq.com",
    "age": 48
}
```

  b. Response with HTTP status code **201** on success
     `{}`
  c. This endpoint will create a new user inside the database

- **/api/users/{id} - GET** - To get the details of a user
    a. Here {id} will be the id of the user in path parameter
    b. Response with HTTP status code **200** on success

```
{
    "id": 1,
    "first_name": "James",
    "last_name": "Butt",
    "company_name": "Benton, John B Jr",
    "city": "New Orleans",
    "state": "LA",
    "zip": 70116,
    "email": "jbutt@gmail.com",
    "web": "http://www.bentonjohnbjr.com",
    "age": 70
}
```

    Sample query looks like:- `/api/users/1` ***`GET`***

- **/api/users/{id} - PUT** - To update the details of a user
    - Here {id} will be the id of the user in path parameter
    - Request Payload should be like in json format for updating first name, last name and age:-

```
{
    "first_name": "Josephine",
    "last_name": "Darakjy",
    "age": 48
}
```

    - Response with HTTP status code **200** on success
      `{}`

- **/api/users/{id} - DELETE** - To delete the user
    - Here {id} will be the id of the user in path parameter
    - Response with HTTP status code **200** on success
      `{}`

Resources

- For sample data https://datapeace-storage.s3-us-west-2.amazonaws.com/dummy_data/users.json

Instructions

1. If you have github, gitlab or any code hosting service account, create a project repository and push the code in the repo.
2. Project directory should have **README.md** file at the root folder, provide all the details and instructions of project, like how to setup and run the project.
3. Share your hosted project url on email **hiring@truevalueaccess.com**
4. Attention to detail is important in the project. Completing it in less time will not give you any preference.
5. Writing tests for your code, will be a plus point for you.