**Ecorus Assignment Notes: Questions and Answers**

How long did it take for assignments?

For 1st assignment less than 2 hours ( 2 different approaches)

For 2nd assignment- took 3-4 days

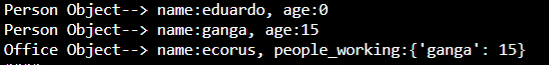
In between I was not feeling well, so could not submit it

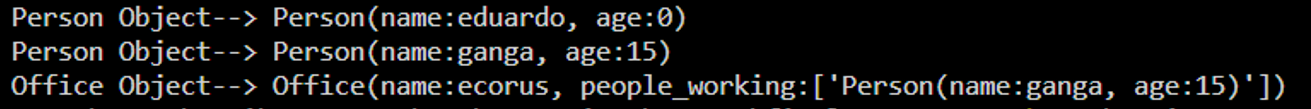
**Output:**

Python code and console screenshot (printing the objects) after every exercise.

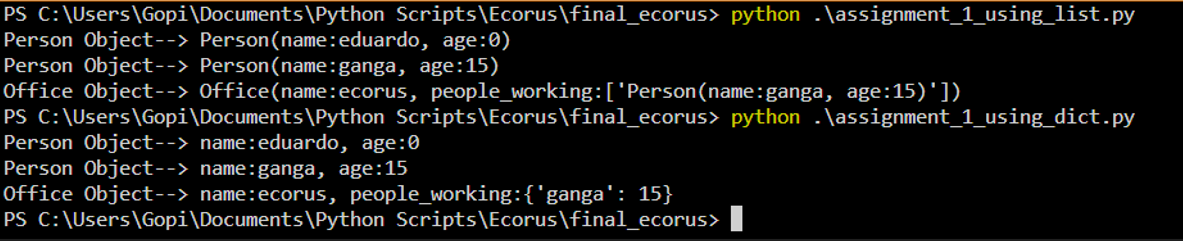
**Assignment 1**:

Output Assignment 1 (using dictionary):



Output Assignment 1 (using list):

Together outputs:

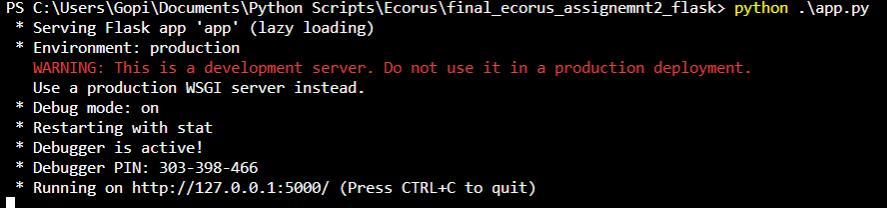


**2nd Assignment**:

Notes:

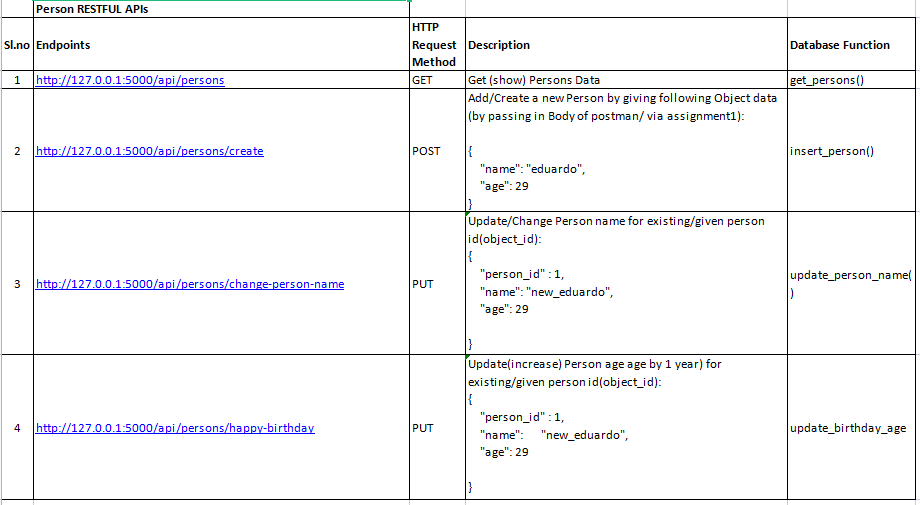
Used Python Flask Framework, sqlite3 Database, flask\_restful api,

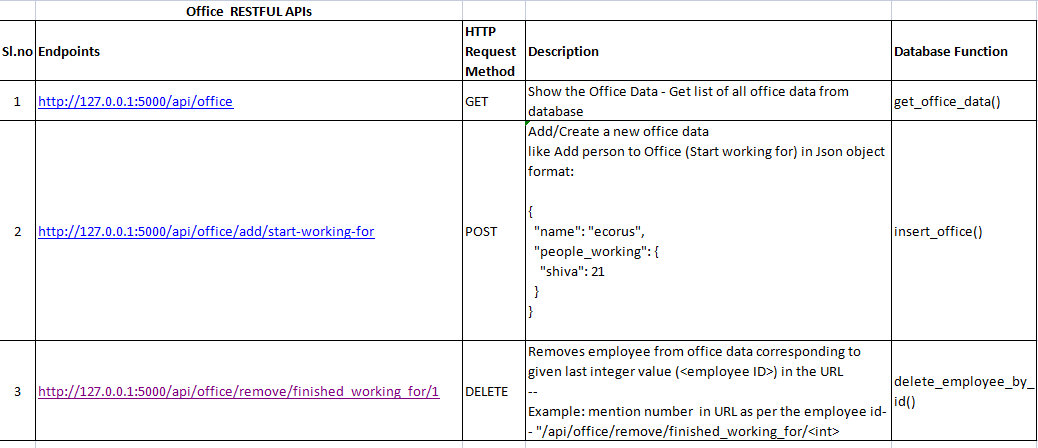
Added a component to improve requests handling, used template with HTML pages to show the data in the front end, used postman to test APIs.



RESTFUL API End points screenshots

**Note:** Also attached Excel sheet , please refer





**Questions:**

1. - What has been the more difficult part?

Was aware about RestAPI, need to research and learn about using Flask RESTFUL API,endpoints, and SQLTE3 Database connections.

1. - What part of the system could be improved?

Frontend UI feature with forms could be included for easy interaction with the Application, can enhance code with better Exception handling refactor database implementation,REstful api authentication, including functional tests to test the database,api, app.

1. - How would you scale it, to be able to handle 1K calls per sec? and to handle 1M?

By using load balancer with multiple application servers. By this we can distribute/route the requests across multiple resources/server. Based on network traffic/demand we can add/ remove servers. (This also helps in resilient to failure)

1. - How would you automate the testing?

Using python Pytest module, we can prepare testcases to test App, database, API (using request library), UI etc.It can either be run as an individual tool for testing or it can be attached to Python web development frameworks like Flask, enhancing their unit test capabilities and reliability.By this way we can automate the running of tests and finally obtain test coverage report. Also we have to note that there are many other tools in the market for testing.

1. - How would you implement a continuous development system (pipelines) for this particular case?

We can use Jenkins to design CI/CD pipeline.

Jenkins is a well-known open-source continuous integration and continuous development automation tool

We need following setup to implement end to end CI/CD pipleline

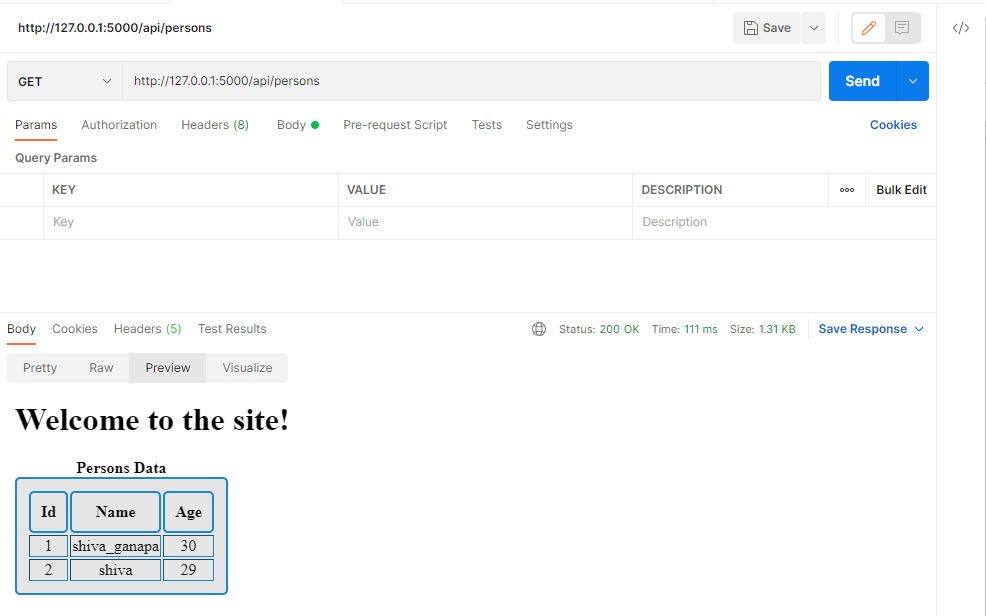
* Python setup : to create code for assignments, unit tests
* API creation using flask
* setting up git repository for the project
* Docker image creation for the API then run it as container
* Jenkins server setup
* Creating CI pipeline in Jenkins and Running it
* Testing the end to end flow

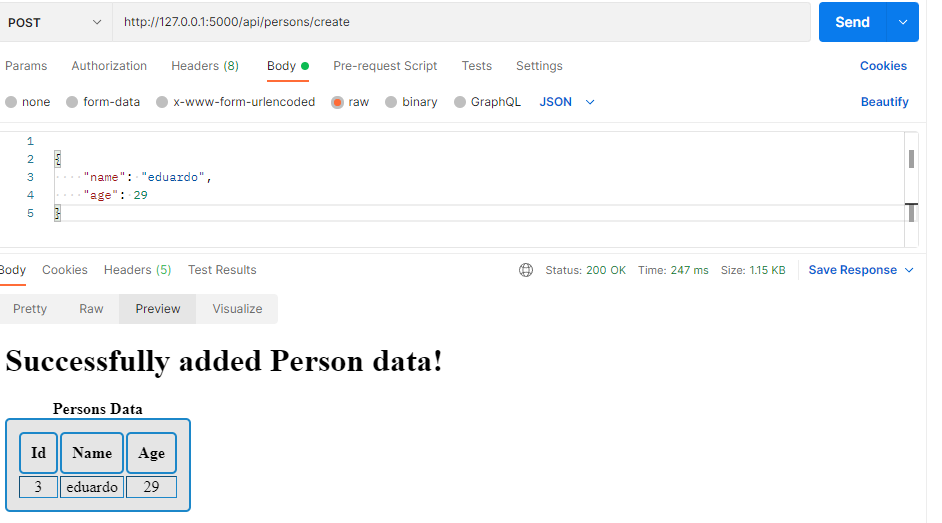
Please note that there are other tools like Azure, Travis CI etc.

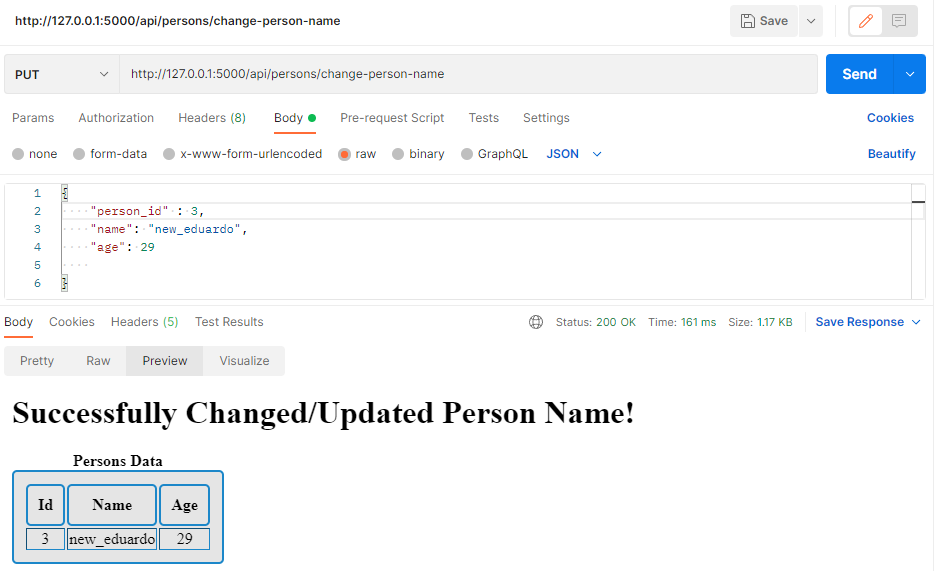
**OUTPUT Screenshots of Assignment 2** -

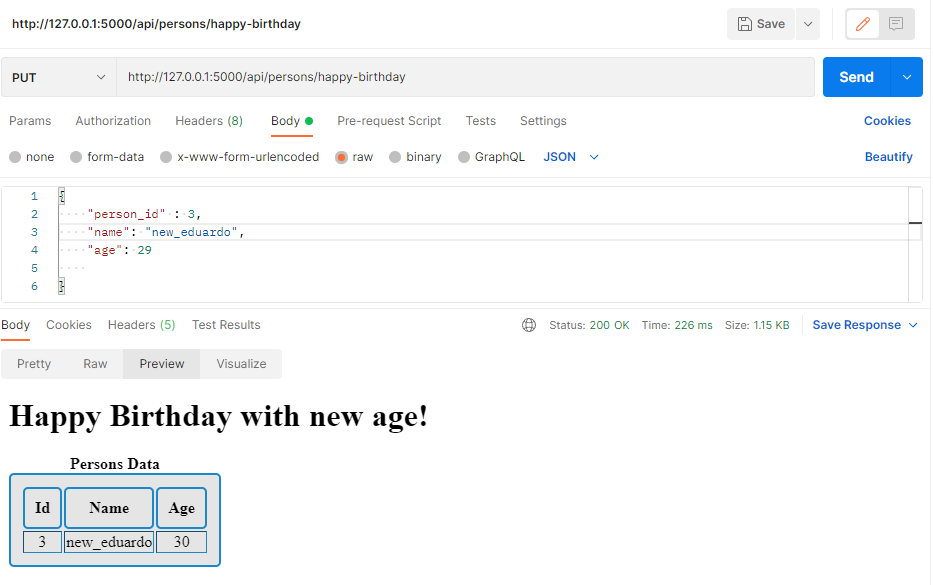
Falsk Restful API Functional Testing using Postman:

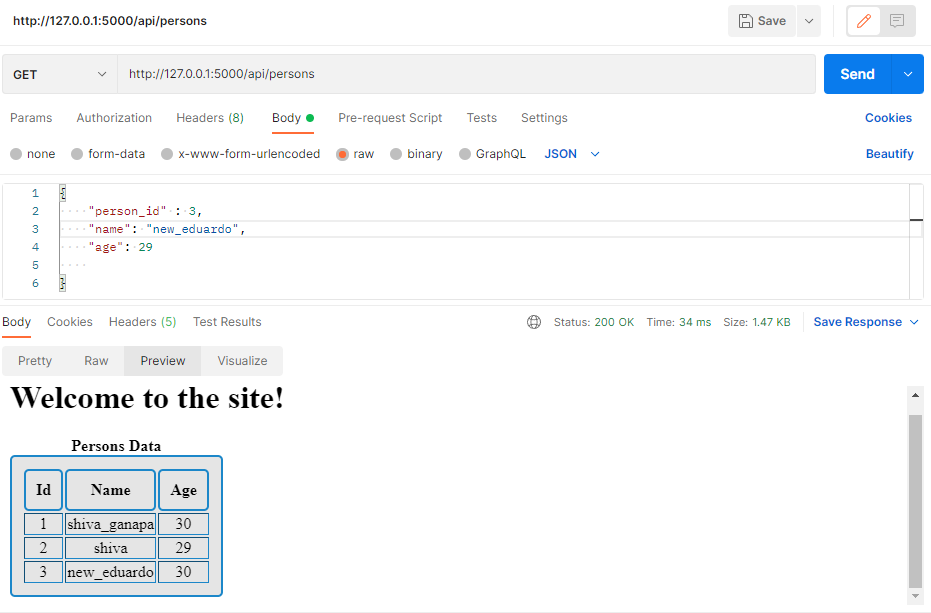
**Person APIs:**



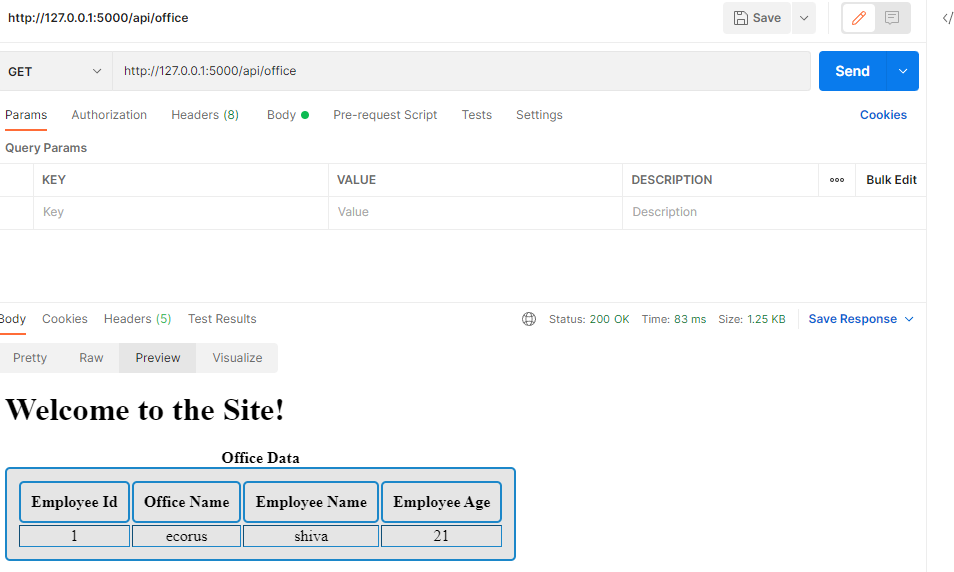


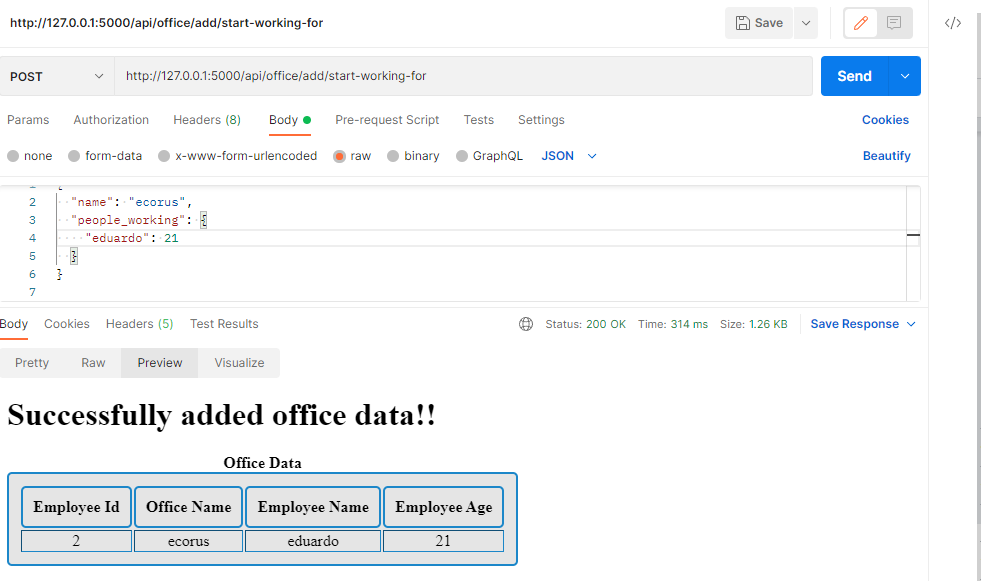


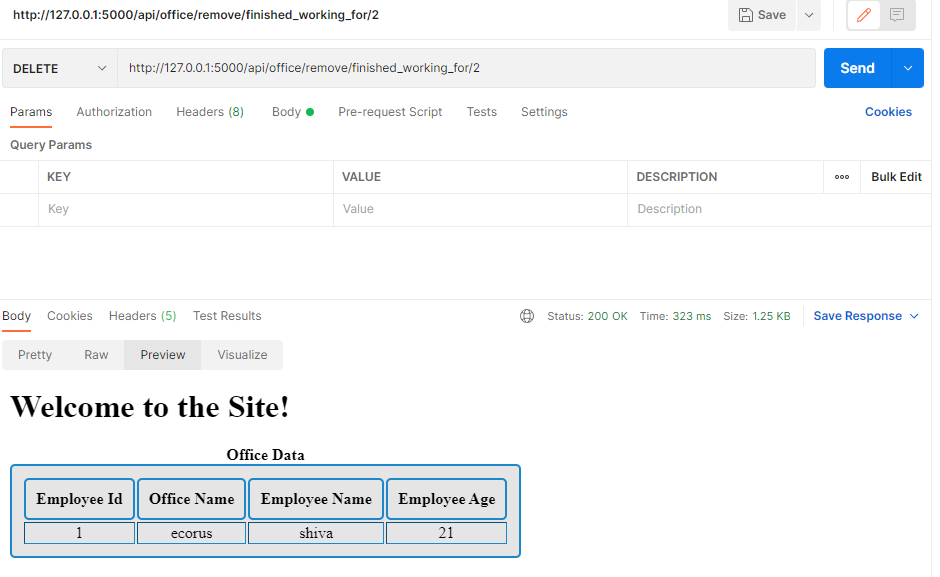




**OFFICE APIS:**

****

****

****