Testing Assignment 1

Team members:

- Mohammad Ashraf Atef - 20176024
- Marwan Harb – 20176028

Github repo's link:

- https://github.com/MohammadAshrafTolba/CountryRestAPI

API description:

The API consists of multiple modules.

1- ***Server:*** The main server that runs the api, also routes different links to different classes. In our case it routes our requests to the appropriate functions in the ***ResponseHandler*** class.

2- ***country_info:*** Contains the ***CountryInfo*** class that calls the public api and retrieve the required info from it.

3- ***response_handler:*** Contains the ***ResponseHandler*** class, where it handler requests by retrieving information from the public API using the ***CountryInfo*** class. Also it raises specialized error messages when needed.

4- ***offline_server:*** Defines a proof of concept offline server that can be use only to retrieve specific info pre stored on the machine. It routes the incoming requests to a class called ***OfflineRequestHandler***.

5- ***offline_request_handler***: Defines the ***OfflineRequestHandler*** class to parse and return offline info using the ***OfflineCountryInfo*** class, retrieving info from pre stored json files (as an example in the project ***offline_country_info.json*** containing **Egypt's** info).

6- ***unit_testing*** and ***integration_testing:*** Classes contain various testing functions, mainly built upon built in package called **unittest**.

Note: Caching requirement is being handled through the request_cache library

<u>Testing:</u>

-For assignment 2, most of these test cases are implemented with mocking specific objects (specially in the integration testing classes)

-The test cases covers nearly 90% of the code (can be shown through the coverage report generated continuously through each build,

run: *coverage report –m****)***

<u>Unit test cases:</u>

1- ***test_get_any_info***: Tests if the api returns any info if passed a valid argument. Testing get_all_info(country_name) function in the ResponseHandler class

2- ***test_invalid_keys***: Tests if the api is passed invalid argument, it will raise the right error message (which is invalid_info error message). Testing the get_info(country_name, keys) function in the ResponseHandler class.

3- ***test_get_true_info***: Tests if the api actually returns true info not just any data. Testing the get_info(country_name, keys) function in the ResponseHandler class.

4- ***test_empty_keys***: Tests if the api is passed emty argument, it will raise the appropriate error message. (which is invalid_info error message).Testing the get_info(country_name, keys) function in the ResponseHandler class.

5- ***test _public_api_error***: Tests if the api will raise the appropriate error (which is public_api_error) if the api is passed none/invalid countries.

6- ***test_get_valid_data***: Same as the ***test_get_true_info*** but with multiple keys.

7- test_filter_with_valid_keys: Testing the ***filter(keys, info)*** function in the ***ResponseHandler*** class, by passing valid keys.

8- test_filter_with_unvalid_keys: Testing the ***filter(keys, info)*** function in the ***ResponseHandler*** class, by passing valid keys. It should raise the invalid_info error

9- ***test_get_invalid_Country:*** Testing the ***get_info(country_name)*** function in the ***CountryInfo*** class, testing if passed invalid country name it should return **none**.

10- ***test_get_from_cache:*** Testing the caching mechanism through testing the **get_info(country_name)** function in the **CountryInfo** class, asserting the **"from cache"** flag to be true in the returned response.

Integration test cases:

For the sake of isolating the working testable part, we used the app.test_client() to mock our app

Here the test cases mainly targets our local link rather than functions.

1- ***test_valid_request***: Tests if the api returns true results if passed appropriate request.
eg: */country/name/egypt/keys/capital+name*

2- ***test_public_api_error***: Tests if the api raises the appropriate error when public api fails. Like when passed invalid country.
eg: */country/name/invalid_country/keys/name*

3- ***test_invalid_request_wrong_info***: Tests if the api raises the appropriate error when public api fails. Like when passed invalid keys.
eg: */country/name/egypt/keys/invalid_key*

Note: We applied exactly the above integration tests for the offline_server, those tests are in the **offline_server_integration_testing** class