
# Peer Review Assignment - Developers

Estimated time needed: **60** minutes

## Objectives

In this assignment you will:

- Use Watson APIs to create functions.
- Create a function that translates English to French.
- Create a function that translates English to German.
- Run coding standards check against the functions above.
- Write unit tests to test the above functions.
- Run unit tests and interpret the results.
- Package the above functions and tests as a standard python package.

# Important Notice

- Please keep in mind that sessions for this lab environment are not persisted. If you will not be completing the entire lab in one sitting, please save your code outisde of this environment, in GitHub or with other providers, so you can resume your work without loss.

## Provision Watson Translation Service

- Before you start, ensure you have provisioned an instance of the Watson Language Translator service and have API information available.

# Create Translation functions

## Create translator.py

- Create a new file called `translator.py`

## Task1: Write a function that translates English text to French in translator.py

- name the function as **english_to_french**
- Hint : You may refer to the API labs in the Python for Data Science, AI and Development course. Here are links to the Jupyter notebooks for the API labs in that course. API Lab 1 | API Lab 2
- You may refer to the IBM Watson Language Translation API documents here.
- Take a screenshot of your function and save it as a .jpg or .png with the filename `english_to_french`. You will be prompted to upload the screenshot in the Peer Assignement that follows.

# Write unit test for the translation function

## Create tests.py

- Create a new file called `tests.py`

## **Task 2:** Write the unit tests for English to French translator function in tests.py

- write at least 3 tests
- test for null input
- test for the translation of the world 'Hello
- Take a screenshot of your unit tests and save it as a .jpg or .png with the filename `english_to_french_unittests`.

# **Task 3:** Write a function that translates English text to German in translator.py

- name the function as **english_to_german**
- Take a screenshot of your function and save it as a .jpg or .png with the filename `english_to_german`.

# **Task 4:** Write the unit tests for English to German translator function in tests.py

- write at least 3 tests
- test for null input
- test for the translation of the world 'Hello
- Take a screenshot of your unit tests and save it as a .jpg or .png with the filename `english_to_german_unittests`.

# **Task 5:** Check your code against python coding standards

- Make sure your rating is at least 7.
- Take a screenshot of the output of the pylint analysis report showing your score and save it as a .jpg or .png with the filename `pylint_score`.

# **Task 6:** Run tests

- To run tests follow the steps below:
- Open a new terminal using the Terminal -> New Terminal option from the menu.
- At the terminal run the command **python3 tests.py**
- Take a screenshot of test results and save it as a .jpg or .png with the filename `unit_test_results`.

# **Task 7:** Package the above functions and tests as a standard python package.

## step 1 - create folder named machinetranslation

## step 2 - create `__init__.py` file

## step 3 - copy the file with functions

## step 4 - create a folder called tests under the newly created folder

## step 5 - copy the unit tests into the tests folder

- Take a screenshot of the folder structure of the package ( From the menu go to View -> Explorer to set the explorer view) an save it as a .jpg or .png with the filename `package_folder_structure`.

# Congratulations!

You have completed the tasks for this project. In the peer assignement that follows, you will be required to upload the screenshots you saved in this lab.

# Authors

Ramesh Sannareddy

## Other Contributors

Rav Ahuja

Azim Hirjani

Alison Woolford

# Change Log

| Date (YYYY-MM-DD) | Version | Changed By | Change Description |
|---|---|---|---|
| 2021-01-29 | 1.3 | Rav Ahuja | Additional instructions, formatting, navigation |
| 2021-01-29 | 1.2 | Alison Woolford | Added prompts for screenshots |
| 2021-01-29 | 1.1 | Azim Hirjani | Fixed typos and license |
| 2020-11-25 | 1.0 | Ramesh Sannareddy | Created initial version of the lab |