Handin 2

Due 20 Sep by 18:00 **Points** 100 **Submitting** an external tool
Available after 13 Sep at 18:00

The handins in the course will consist of two parts: 1) a set of short exercises to practice a particular part of the curriculum (e.g. loops), and 2) a project part. In the project part we will consider a specific Data Science case, each week working on the same (or similar) data, and gradually building up a complete analysis of the data.

Note that the tools required to solve the handins will be generally covered in the lectures in the week where the assignment is given. Sometimes, you might have to wait until the Friday lecture before you have all the tools to complete the assignment.

Part 1

This week, we'll look into tuples, dictionaries and functions.

1. Create a file called `handin2.py`. Inside this file, create a variable called `beatles_container1`, and assign to this variable a list of tuples, where each tuple consists of the the name and the instrument played by one of the Beatles:

Paul McCartney	bass guitar
John Lennon	rhythm guitar
George Harrison	lead guitar
Ringo Starr	drums

   The list should thus have 4 members, and each entry should be a tuple with two strings: name and instrument.

2. We want to be able to look up the instruments by the name of the band member. Inside the same `handin2.py` file, create a variable called `beatles_container2`, and choose an appropriate data type, such that we can write:

   ```
   print(beatles_container2['Ringo Starr'])
   ```

   to get the output `drums`. When writing this code, you should use the values from your `beatles_container1` - i.e. you should not type in the values again.
3. Now, let's put the functionality above into a function. Again, inside `handin2.py`, create a function called `beatle_lookup` and copy&paste your code from above into the body of the function. The function should have an argument called `name`, and then return the

instrument of the band member with that name.

If the name passed to the function is not one of the 4 Beatles, the function should return a string with an error message that provides the user with information on how the function can be called - this message should mention the four names for which the function works. For instance:

```
ERROR. Name 'Mick Jagger' not found. Available names: ['Paul McCartney', 'John Lennon', 'George Har
rison', 'Ringo Starr']
```

You should do this without typing in the four Beatles names again. Note: we'll see later in the course how to deal properly with error message in Python.

After the function definition, test your function by *calling* it first on `"John Lennon"` and then on `"Mick Jagger"`.

Part 2: Project

Last week, we opened the temperature anomaly data file, read it in, and then printed lines within a particular year range. This week, we'll put the data into a container variable, and wrap the code in a function, so we can make the selection for particular years more flexible.

1. Create a file called `handin2_project.py`. Write similar code as last week to iterate over the lines in the file and strip away the newlines (i.e. feel free to copy&paste), but now take each line and add it to a list called `data_list`. Only ignore comment lines and empty lines (i.e. include the lines for all years).
2. Now, we'll put this functionality inside a function. Create a function called `read_data` that takes a single argument: a string argument called `filename`. The function should read in the data from the specified filename and return a list of strings (in the same format as we had in `data_list` in the previous exercise). Feel free to copy&paste from the previous question.

   Below the function definition, *call* the function on the `Land_and_Ocean_summary.txt` file, and save the result in a variable called `data_list_returned`.

When you are done, click on the "Load Handin2 in a new window" button below, which will take you to the CodeGrade server. Here, please submit the `handin2.py`, `handin2_project.py` files. CodeGrade will then automatically check the code for you, and upgrade your grade for the assignment within Absalon. You can submit as many times as you want.

This tool needs to be loaded in a new browser window

Load Handin 2 in a new window