ACIT2515 - In class evaluation

This assignment must be completed in class. Submit the following files to D2L - **DO NOT ZIP OR COMPRESS YOUR FILES** !

- `person.py`
- `car.py`

Exercise 1: `Person` class (4 marks)

Create the class `Person`:

- it has a constructor taking two arguments:
  - the name of the person: stored in the `name` attribute of the instance
  - the age of the person: stored in the `age` attribute of the instance
  -
- the constructor must check that:
  - the name provided is a string containing at least three letters
  - the age of the person is a strictly positive integer (strings are not accepted)
  - if any of these conditions is not met, the constructor must raise an `AttributeError`
- it has a method `get_name`, which returns a string with the following elements separated by spaces:
  - the name of the person in **uppercase**
  - a slash `/`
  - the age of the person
  - example: `TIM GUICHERD / 20`

You can use the `test_person.py` file to check your code.

Submission and granding

- 1 mark for each test that passes

Exercise 2: the `Bike` class (6 marks)

Create the `Bike` class. It represents a shared bike that can be rented by the hour.

The bike has the following attributes:

- `rider` (a string) - set to `None` by default
- `distance` (an integer) - set to `0` by default
- these attributes are set in the constructor, not received as arguments!

The bike has the following methods:

- `start_rental`: takes a string (the name of the driver renting the car)
    - starts a rental period
    - it sets the `rider` attribute to the name provided
    - there can only be one person riding the bike at a time. If a rental is already active (`rider` is set) and `start_rental` is called: raise a `RuntimeError` exception.
- `bike`: takes an integer argument (the distance biked)
    - it adds the distance provided to the `distance` attribute
    - the distance driven must be a positive number - otherwise raise an `AttributeError` exception
- `end_rental`: returns the total distance driven during the rental period
    - sets the `rider` and `distance` back to their defaults (`None` and `0`)

You can use the `test_bike.py` file to check your code.

Submission and granding

- 1 mark for each test that passes