Programmation Python Orientée Objet

🔧 Fix Me! : en construction

Concepts utilisés

- Un objet : c'est… n'importe quoi, qui peut être codé. En Python, tout est objet !
- Une classe est une description générique d'un type d'objet, incluant les données et les méthodes qui le caractérisent
    - Convention : on écrit les noms de classe en “CamelCase”
- Une instance d'une classe, c'est un objet individuel bien précis, avec son identifiant, et toutes les propriétés liées à la classe à laquelle il appartient.
    - Convention : les noms des instances commencent par une lettre en bas de casse (minuscule)
- Méthode : les objets peuvent évoluer par la réalisation de méthodes, qui sont des fonctions associées à des objets, pouvant accéder à leurs données. Comme toute fonction, les méthodes peuvent passer des arguments et renvoyer des valeurs
- Encapsulation : https://fr.wikipedia.org/wiki/Encapsulation_(programmation)
- Héritage : …
- Abstraction
- Polymorphisme

intro-OO-01.py3

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Introduction to object-oriented programming in Python

Led with number and status attributes
"""
class Led:
    def __init__(self, number, status):
        self.number = number
        self.status = status
    def on(self):
        self.status = 'on'
    def off(self):
        self.status = 'off'

led1 = Led(1, 'on')
print(led1)
print(type(led1))
print(type(Led))
print(led1.number)
```

```
print(led1.status)
```

Références

- https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet
- UML :
  - https://fr.wikipedia.org/wiki/UML_(informatique)
  - http://argouml.tigris.org/

From:
https://dvillers.umons.ac.be/wiki/ - **Didier Villers, UMONS - wiki**

Permanent link:
https://dvillers.umons.ac.be/wiki/teaching:progappchim:ppoo?rev=1467292536

Last update: **2016/06/30 15:15**