UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

La programmation objet avec Python

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Principe général

- La programmation objet est un paradigme dans lequel :
  - L'idée est de définir des briques logicielles appelées **objets**,
  - Un objet représente un concept, une idée ou toute entité du monde physique,
  - Les objets ont une structure interne (**implémentation**) et un comportement (**interface**).
- Le but de cette méthode :
  - permettre une meilleure modélisation des problèmes, et donc une meilleure programmation,
  - obtenir des briques logiciel facilement réutilisables en dissociant l'implémentation de l'interface (**encapsulation**).

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Langage

- On peut programmer orienté objet dans n'importe quel langage.
- Mais certains ont un support natif de la POO : Smalltalk, Objective-C, Java, C++, Go, PHP, Python, Self, JavaScript, …
- Ils ne fonctionnent pas tous de manières différentes :
  - soit avec des classes qu'on instancie (C++, Java, Python) ;
  - soit avec des prototypes qu'on clone (Self, JavaScript) ;
  - ou alors avec d'autres choses moins clairement nommées (Go).

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

La classe

- Une classe est la déclaration d'un type d'objet.
- Les données internes de la classe appelées **attributs**.
- Les traitements sur ces données et/ou d'autres données sont appelées **méthodes**.
- On parle d'une **instance** d'une classe lors de la création d'un objet qui a cette classe pour type.

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Objet

- Un objet est l'instance d'un type :
  - *12* est une instance de ***int***
  - *`bonjour`* est une instance ***str***
- Ne pas confondre créer une classe et instancier une classe :
  - Créer une classe = définir une nouvelle classe, écrire ses attributs et méthodes.
  - Instancier une classe = créer un exemplaire d'un objet. Par exemple *i=1*.

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Créer une classe

- On déclare une classe avec le mot-clé class suivi du nom de cette classe :

```
class nom_classe :
    attributs et méthodes
# suite du programme
```

- Les attributs sont des données qui vont être partagées par l'ensemble des instances de la classes

```
class Point :
    x = 0
    y = 0
# suite du programme
```

- Si x est modifié par une instance de Point, alors elle modifiées pour toutes les autres instances !

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Fonctions - Méthodes

- Les méthodes et les fonctions sont fondamentalement de même nature.
- On les appelle méthodes quand elles sont implémentées dans une classe.
- et fonctions dans le reste du code.
- En python les fonctions sont généralement, comme en mathématiques, préfixes : fonction(argument).
- Et les méthodes sont généralement postfixes : objet.méthode()

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Constructeur

- Lorsqu'une méthode est appelée sur une instance, elle reçoit cette instance en argument.
  - Dans certains langages, le passage de cet argument est entièrement implicite, et son nom est alors le plus souvent le mot-clef *this*.
  - En Python, c'est le premier argument de la méthode et on le nomme **self**, par convention.
- Il existe une méthode spéciale qu'on appelle le constructeur de la classe et qui est chargée de l'**initialisation** des attributs.
  - Parfois c'est la méthode qui a le même nom que la classe.
  - En Python, c'est la méthode qui s'appelle **__init__**.
  - C'est la méthode qui est appelée quand on **instancie** la classe.

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Constructeur

- Le constructeur est une méthode spéciale qui permet d'instancier un objet.

```
class Point :
    def __init__(self, x, y):
        print("création d'un objet point")
        self.x = x
        self.y = y
# suite du programme
A = Point (2, 1)
print (A.x)
```

- x et y sont des attributs d'instance de la classe Point. Elles reçoivent ici les valeurs passées en paramètres du constructeur.

- Pour offrir plusieurs constructeurs, on donne une valeur par défaut aux variables d'instances

- On interagit avec les attributs d'un objet à l'aide du .

UNIVERSITÉ PARIS 8 VINCENNES-SAINT-DENIS

Encapsulation : principe

- Une des forces de la programmation orientée objet est l'**encapsulation**.
- Il s'agit de cacher les détails de l'implémentation d'une classe à ses utilisateurs, et de leur offrir une interface stable.
- Ainsi :
  - l'implémentation de la classe peut changer sans impact sur l'utilisateur,
  - les objets sont seuls responsables du maintien de leur cohérence interne !