

# Langage Python

## Cours 1/5 : bases du langage

Hubert Godfroy

`hubert.godfroy@univ-lorraine.fr`

22 octobre 2015

# Plan

Déroulement du cours

Intérêts de Python

Éléments de base

Prise en main de l'IDE

Le cours se déroulera selon ce plan

- ▶ 10 séances de 2h
- ▶ 5 séances de cours/TDs
- ▶ 4 séances de préparation du projet
- ▶ Une séance de présentation (14 janvier 2016)

Un projet est à rendre avant le 10 janvier 2016. Plus de détails vous seront donnés dans les prochaines semaines.

# Plan

Déroulement du cours

**Intérêts de Python**

Éléments de base

Prise en main de l'IDE

# Vue d'ensemble de Python

- ▶ Langage de script
- ⇒ Automatisation des tâches

# Vue d'ensemble de Python

- ▶ Langage de script
- ⇒ Automatisation des tâches
- ▶ Syntaxe minimale
- ▶ Très peu verbeux
- ⇒ Adapté au prototypage et à l'étude d'algorithmes

# Vue d'ensemble de Python

- ▶ Langage de script
- ⇒ Automatisation des tâches
- ▶ Syntaxe minimale
- ▶ Très peu verbeux
- ⇒ Adapté au prototypage et à l'étude d'algorithmes

## Mot d'ordre

*Minimiser le nombre d'erreurs d'exécution*

## Contexte d'utilisation

- ▶ Un langage interprété
- ▶ s'exécute dans une machine virtuelle

# Contexte d'utilisation

- ▶ Un langage interprété
- ▶ s'exécute dans une machine virtuelle
- ⇒ Code portable
- ⇒ Environnement interactif
- ↪ Exécution lente par rapport à des langages compilé (ex. C).

# Plan

Déroulement du cours

Intérêts de Python

**Éléments de base**

Prise en main de l'IDE

## Prise en main

- ▶ Utilisation des mêmes opérateurs ( +, -, \*, /, %, ==, ...)
- ▶ L'indentation du code est importante.

## Prise en main

- ▶ Utilisation des mêmes opérateurs ( +, -, \*, /, %, ==, ...)
- ▶ L'indentation du code est importante.

Démo!

# Variables

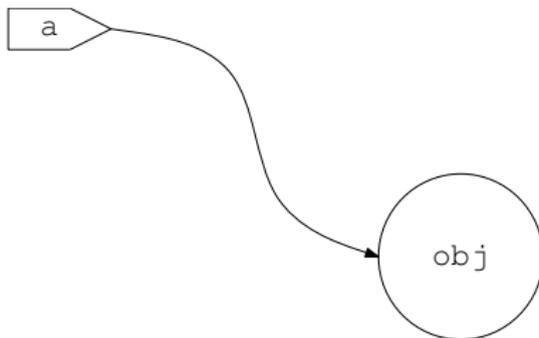
- ▶ Pas besoin de déclarer les variables.
- ▶ Les variables ne sont pas typée.
- ▶ Les valeurs étiquetée par les variables le sont !

# Explication plus détaillée

- ▶ Un **objet** est un **emplacement mémoire**.
- ▶ Une **variable** est une **étiquette** vers un objet Python.

## Exemple

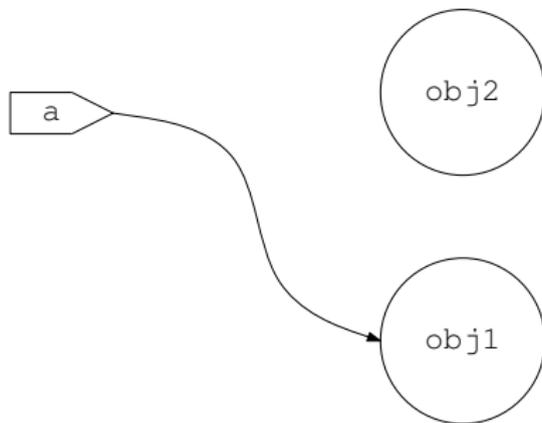
`a = obj`



# Réaffectation

```
a = obj1
```

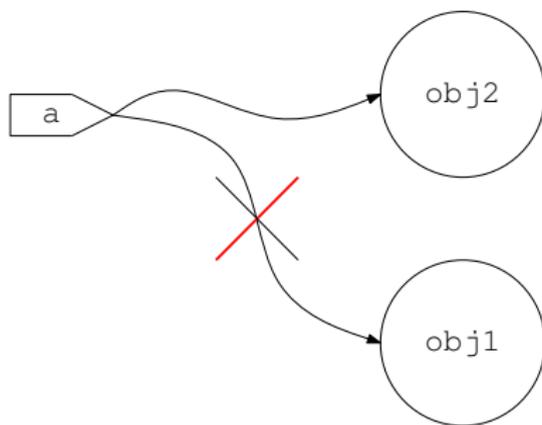
```
a = obj2
```



# Réaffectation

a = obj1

a = obj2



## Entrée/sorties

Se font avec `input` et `print` :

# Entrée/sorties

Se font avec `input` et `print` :

## Exemple de sortie

```
print ("Hello world")  
print ("La réponse est %d"% 42)  
print ("π = %f"% 3.14)
```

# Entrée/sorties

Se font avec `input` et `print` :

## Exemple de sortie

```
print ("Hello world")  
print ("La réponse est %d"% 42)  
print ("π = %f"% 3.14)
```

## Exemple d'entrée

```
réponse = input ("Question")
```

# Déclaration de fonctions

## Syntaxe

```
def nom_fonction (x1, ...):  
    ...
```

# Déclaration de fonctions

## Syntaxe

```
def nom_fonction (x1, ...):  
    ...
```

## Exemple

```
def irma (n) :  
    return (random.randint(1, n))
```

## Exemple minimal

Concision du style :

# Exemple minimal

Concision du style :

```
def f(n):  
    return n*2  
print f(5)
```

# Exemple minimal

Concision du style :

```
def f(n):  
    return n*2  
print f(5)
```

À comparer avec

```
public class Main {  
    public static int f (int anInt){  
        return 2*anInt;  
    }  
    public static void main(String[] args) {  
        System.out.println(f(3));  
    }  
}
```

# Plan

Déroulement du cours

Intérêts de Python

Éléments de base

Prise en main de l'IDE

- ▶ Environnement de développement intégré PyCharm
- ▶ Aide à la programmation : auto-complétion sémantique, réusinage, aide à la correction d'erreurs, ...

DEMO

Et maintenant, à vous de jouer !