

INF3005 – Programmation web avancée

Introduction à Python

Jacques Berger

# Objectifs

Introduire le langage de programmation Python

# Prérequis

# Programmation

# Python

Langage de programmation orienté-objet

Versions courantes : 2.7 & 3.6

Version utilisée dans ce cours : 3.6

# Python 3.6

Pourquoi Python 3?

Python 2 ne sera plus supporté le 12 avril 2020

# Python 3

Python 3 introduit plusieurs nouveaux concepts et de nouveaux éléments syntaxiques

De plus, les programmes fait avec Python 2 ne sont pas compatibles avec Python 3

# Python

Python sera lancé à la ligne de commande

Il faut :

- Installer Python

- Possiblement configurer le PATH du shell

Pour connaître la version installée :

```
python3 -V
```

# Hello World!

Un premier programme :

```
print("Hello World!")
```



# Opérateurs

Les opérateurs habituels fonctionnent :

+ - \* / < > = == <= >= += -= != %

Opérateurs booléens :

and or not

# strings

Le délimiteur de string est le ' ou le "

```
print("Chaîne avec des caractères accentués")  
print('Chaîne avec des caractères accentués')  
print("J'aime les fruits.")  
print('Ces élus sont "compétents", parfois!')
```

Voir exemples

# Typage

Python met en oeuvre le typage dynamique

Pas d'instruction de déclaration

Pas de type lors de la déclaration

Le type est déterminé à l'exécution en fonction du contenu de la variable

# Variable

Les variables sont en snake\_case

```
annee_courante = 2017  
annee_naissance = 1967  
age = annee_courante - annee_naissance
```

# Fichier

## Lecture d'un fichier

```
text_file = open("input/input1")  
print(text_file.read())  
text_file.close()
```

## Voir les fonctions readline et write

Pour écrire dans un fichier, ajouter le mode d'ouverture w lors de l'appel à open

# Bloc

Les blocs d'exécution sont définis avec l'indentation

Le code qui va dans un if doit être indenté avec le if en question

On indente de 4 espaces

# Condition

## if / elif / else

```
if user_number < mystery_number:  
    print("Plus haut!")  
elif user_number > mystery_number:  
    print("Plus bas!")  
else:  
    found = True
```

# while

## La boucle while

```
tries = 10
while tries > 0 and found == False:
    ...
    tries = tries - 1
```



# Liste

Une liste peut contenir des éléments de différents types

```
une_liste = ["Pascal", "SQL", "PHP", "Python", "Ruby", "Groovy"]
```

# Liste

On accède aux éléments avec l'opérateur [ ] (en spécifiant la position de l'élément)

La fonction len permet de calculer la taille de la liste

# for

## La boucle for permet de parcourir une liste

```
for each in une_liste:  
    print(each)
```

```
for i in range(0, 10):  
    print(i)
```

# Dictionnaire

Ensemble de clés/valeurs  
Peut être appelé map

```
objet = {"nom": "Berger", "prenom": "Jacques", "age": 87}
```

On accède aux valeurs avec l'opérateur [ ]

# Dictionnaire

La fonction `items` retourne une liste sur laquelle on peut boucler

```
for cle, valeur in objet.items():  
    print(cle, valeur)
```

# Dictionnaire

On supprime dans un dictionnaire avec le mot-clé  
del

# Fonction

On déclare des fonctions avec le mot-clé `def`, on retourne une valeur avec `return`

```
def print_error_message(message):  
    today = get_today()  
    print("Erreur", today, message)  
  
def get_today():  
    return datetime.date.today()
```

# Fonction

Les noms de fonction sont en snake\_case



# Classe

## Définir une classe :

```
class Person(object):  
    def __init__(self, firstname, lastname):  
        self.firstname = firstname  
        self.lastname = lastname  
  
    def get_complete_name(self):  
        return "%s %s" % (self.firstname, self.lastname)  
  
    def set_age(self, age):  
        self.age = age
```

# Classe

Toutes les méthodes reçoivent le paramètre `self` (le premier paramètre) correspondant à l'instance

On place les variables d'instance dans la variable `self`

La méthode `__init__` est le constructeur

# Classe

## Instancier une classe

```
teacher = Person("Jacques", "Berger")
```

# Classe

Les noms de classe sont en PascalCase

# Plus loin...

Learn Python the Hard Way, Third Edition  
Zed A. Shaw  
Addison-Wesley, 2014