

Présentation Python



NOTIONS TRANSVERSALES DE PROGRAMMATION

Au collège (cycle 4), les élèves ont découvert et pratiqué les éléments fondamentaux d'algorithmique et de programmation. Le programme de seconde de mathématiques approfondit l'apprentissage de la programmation. Une coordination avec le cours de mathématiques est donc nécessaire pour déterminer à quel moment des éléments de programmation peuvent être utilisés en sciences numériques et technologie.

Contenus	Capacités attendues
Affectations, variables Séquences Instructions conditionnelles Boucles bornées et non bornées Définitions et appels de fonctions	Écrire et développer des programmes pour répondre à des problèmes et modéliser des phénomènes physiques, économiques et sociaux.
Exemples d'activités	
Illustrer ces notions par des activités liées aux différents thèmes du programme.	

- INTRODUCTION

- INTRODUCTION

- ENVIRONNEMENT PYTHON

- INTRODUCTION
- ENVIRONNEMENT PYTHON
- LA CONSOLE

- INTRODUCTION
- ENVIRONNEMENT PYTHON
- LA CONSOLE
- LES SCRIPTS

- INTRODUCTION
- ENVIRONNEMENT PYTHON
- LA CONSOLE
- LES SCRIPTS
- LES MODULES

INTRODUCTION

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

Il est sous licence libre GPL et gratuit et fonctionne sur toutes les plateformes (Windows, Linux, OSX . . .).

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

Il est sous licence libre GPL et gratuit et fonctionne sur toutes les plateformes (Windows, Linux, OSX . . .).

Python a été conçu pour être un langage lisible.

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

Il est sous licence libre GPL et gratuit et fonctionne sur toutes les plateformes (Windows, Linux, OSX . . .).

Python a été conçu pour être un langage lisible.

Les commentaires sont indiqués par le caractère croisillon (#).

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

Il est sous licence libre GPL et gratuit et fonctionne sur toutes les plateformes (Windows, Linux, OSX . . .).

Python a été conçu pour être un langage lisible.

Les commentaires sont indiqués par le caractère croisillon (#).

Les blocs sont identifiés par l'indentation.

INTRODUCTION

Python est un langage interprété créé par Guido Van Rossum au début des années 90.

Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

Il est sous licence libre GPL et gratuit et fonctionne sur toutes les plateformes (Windows, Linux, OSX . . .).

Python a été conçu pour être un langage lisible.

Les commentaires sont indiqués par le caractère croisillon (#).

Les blocs sont identifiés par l'indentation.

Python possède un certain nombre de mots-clés, des objets de différents types (int, float, bool, tuple, list, string . . .).

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.
Le site officiel est : <http://www.python.org>.

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.
Le site officiel est : <http://www.python.org>.
L'éditeur associé est IDLE.

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.
Le site officiel est : <http://www.python.org>.
L'éditeur associé est IDLE.
On peut utiliser : Pyzo,

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder,

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython, ...

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython,

- ▶ Sur tablette android, on peut utiliser QPython3

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython,

- ▶ Sur tablette android, on peut utiliser QPython3
- ▶ Sur tablette iOS, on peut utiliser Pythoni3
- ▶ On peut utiliser un navigateur et utiliser un site web qui propose ce service. Quatre sites assez complets :

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython,

- ▶ Sur tablette android, on peut utiliser QPython3
- ▶ Sur tablette iOS, on peut utiliser Pythoni3
- ▶ On peut utiliser un navigateur et utiliser un site web qui propose ce service. Quatre sites assez complets :
 - ★ <https://jupyter.org/try/>

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython, ...

- ▶ Sur tablette android, on peut utiliser QPython3
- ▶ Sur tablette iOS, on peut utiliser Python3
- ▶ On peut utiliser un navigateur et utiliser un site web qui propose ce service. Quatre sites assez complets :
 - * <https://jupyter.org/try/>
 - * <https://trinket.io/python3>

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython, ...

- ▶ Sur tablette android, on peut utiliser QPython3
- ▶ Sur tablette iOS, on peut utiliser Pythoni3
- ▶ On peut utiliser un navigateur et utiliser un site web qui propose ce service. Quatre sites assez complets :
 - * <https://jupyter.org/try/>
 - * <https://trinket.io/python3>
 - * <https://repl.it/languages/python3>

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython, ...

- ▶ Sur tablette android, on peut utiliser QPython3
- ▶ Sur tablette iOS, on peut utiliser Pythoni3
- ▶ On peut utiliser un navigateur et utiliser un site web qui propose ce service. Quatre sites assez complets :
 - ★ <https://jupyter.org/try/>
 - ★ <https://trinket.io/python3>
 - ★ <https://repl.it/languages/python3>
 - ★ <http://pythonfiddle.com/>

ENVIRONNEMENT PYTHON 3

- ▶ On peut utiliser un éditeur installé sur le serveur du lycée ou sur son ordinateur personnel.

Le site officiel est : <http://www.python.org>.

L'éditeur associé est IDLE.

On peut utiliser : Pyzo, Sypder, EduPython, ...

- ▶ Sur tablette android, on peut utiliser QPython3
- ▶ Sur tablette iOS, on peut utiliser Pythoni3
- ▶ On peut utiliser un navigateur et utiliser un site web qui propose ce service. Quatre sites assez complets :
 - ★ <https://jupyter.org/try/>
 - ★ <https://trinket.io/python3>
 - ★ <https://repl.it/languages/python3>
 - ★ <http://pythonfiddle.com/>

Dans un environnement Python, on dispose toujours d'une fenêtre appelée « console » ou « Shell » et un éditeur de scripts Python. Avec Pyzo, on a :

```
premierProgram.py (D:\stageSNT\Python\premierProgram.py) - Interactive Editor for Python
File Edit View Settings Shell Run Tools Help
Shells
Python
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) on Windows (32 bits).
This is the Pyzo interpreter with integrated event loop for ASYNCIO.
Type 'help' for help, type '?' for a list of *magic* commands.
Running script: "D:\stageSNT\Python\premierProgram.py"

>>> 2*3+4
10

>>> print("Bonjour")
Bonjour

>>> Bonjour()
'bonjour'
```

```
premierProgram.py
1 def Bonjour():
2     #ceci est un commentaire : très utile pour savoir ce qu'on a fait !
3     #cette fonction renvoie la chaine 'bonjour'
4
5     return('bonjour')
6
```

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16
- ▶ définir et modifier des variables

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16
- ▶ définir et modifier des variables

Une variable est un espace mémoire dans lequel il est possible de stocker une valeur (une donnée).

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16
- ▶ définir et modifier des variables
Une variable est un espace mémoire dans lequel il est possible de stocker une valeur (une donnée).
- ▶ évaluer des variables

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16
- ▶ définir et modifier des variables
Une variable est un espace mémoire dans lequel il est possible de stocker une valeur (une donnée).
- ▶ évaluer des variables
- ▶ exécuter des instructions.

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16
- ▶ définir et modifier des variables
Une variable est un espace mémoire dans lequel il est possible de stocker une valeur (une donnée).
- ▶ évaluer des variables
- ▶ exécuter des instructions.

On exécute une instruction à la fois (ou un bloc à la fois) et un affichage se produit s'il n'y a pas d'erreur de syntaxe.

LA CONSOLE

Lorsqu'on a la main, on a généralement un curseur qui clignote devant « >>> » ou [1] :

On peut utiliser la console pour :

- ▶ faire des calculs : $1+5*3$ renvoie 16
- ▶ définir et modifier des variables
Une variable est un espace mémoire dans lequel il est possible de stocker une valeur (une donnée).
- ▶ évaluer des variables
- ▶ exécuter des instructions.

On exécute une instruction à la fois (ou un bloc à la fois) et un affichage se produit s'il n'y a pas d'erreur de syntaxe.

Attention : Python est sensible à la casse ; a et A ne désigne pas la même variable.

L'ÉDITEUR DE SCRIPTS

On peut écrire les instructions dans un fichier texte dont l'extension est `.py`, dans une fenêtre appelée l'éditeur de script.

L'ÉDITEUR DE SCRIPTS

On peut écrire les instructions dans un fichier texte dont l'extension est `.py`, dans une fenêtre appelée l'éditeur de script. Ensuite, on demande à l'interpréteur d'exécuter toutes les instructions, les unes après les autres.

L'ÉDITEUR DE SCRIPTS

On peut écrire les instructions dans un fichier texte dont l'extension est `.py`, dans une fenêtre appelée l'éditeur de script. Ensuite, on demande à l'interpréteur d'exécuter toutes les instructions, les unes après les autres. L'avantage est qu'on peut sauvegarder les instructions dans le fichier et y écrire plusieurs « programmes » ou « fonctions ».

L'ÉDITEUR DE SCRIPTS

On peut écrire les instructions dans un fichier texte dont l'extension est `.py`, dans une fenêtre appelée l'éditeur de script. Ensuite, on demande à l'interpréteur d'exécuter toutes les instructions, les unes après les autres. L'avantage est qu'on peut sauvegarder les instructions dans le fichier et y écrire plusieurs « programmes » ou « fonctions ».

```
1 # coding : utf-8
2
3 def Bonjour():
4     return('Bonjour')
5
6 def carre(x):
7     y=x**2 # le carré de x
8     return(y)
9
10 def moyenne(a,b):
11     m=(a+b)/2
12     return(m)
13
```

L'ÉDITEUR DE SCRIPTS

On peut écrire les instructions dans un fichier texte dont l'extension est .py, dans une fenêtre appelée l'éditeur de script. Ensuite, on demande à l'interpréteur d'exécuter toutes les instructions, les unes après les autres. L'avantage est qu'on peut sauvegarder les instructions dans le fichier et y écrire plusieurs « programmes » ou « fonctions ».

```
1 # coding : utf-8
2
3 def Bonjour():
4     return('Bonjour')
5
6 def carre(x):
7     y=x**2 # le carré de x
8     return(y)
9
10 def moyenne(a,b):
11     m=(a+b)/2
12     return(m)
13
```

Par exemple pour obtenir la moyenne de 8 et 10, il suffit de taper dans la console : `moyenne(8, 10)` ou `print(moyenne(8,10))` puis retour clavier.

LES MODULES.

On a parfois besoin d'utiliser des fonctions de Python qui ne sont pas chargées par défaut. Ces fonctions sont stockées dans des programmes Python appelées modules ou bibliothèques.

LES MODULES.

On a parfois besoin d'utiliser des fonctions de Python qui ne sont pas chargées par défaut. Ces fonctions sont stockées dans des programmes Python appelées modules ou bibliothèques.

Quelques exemples de module :

LES MODULES.

On a parfois besoin d'utiliser des fonctions de Python qui ne sont pas chargées par défaut. Ces fonctions sont stockées dans des programmes Python appelées modules ou bibliothèques.

Quelques exemples de module :

- ▶ le module `math` qui contient les fonctions mathématiques usuelles
- ▶ le module `PIL` pour le traitement des photographies numériques.

LES MODULES.

On a parfois besoin d'utiliser des fonctions de Python qui ne sont pas chargées par défaut. Ces fonctions sont stockées dans des programmes Python appelées modules ou bibliothèques.

Quelques exemples de module :

- ▶ le module `math` qui contient les fonctions mathématiques usuelles
- ▶ le module `PIL` pour le traitement des photographies numériques.
- ▶ le module `folium` pour la géolocalisation.

LES MODULES.

On a parfois besoin d'utiliser des fonctions de Python qui ne sont pas chargées par défaut. Ces fonctions sont stockées dans des programmes Python appelées modules ou bibliothèques.

Quelques exemples de module :

- ▶ le module `math` qui contient les fonctions mathématiques usuelles
- ▶ le module `PIL` pour le traitement des photographies numériques.
- ▶ le module `folium` pour la géolocalisation.
- ▶ le module `csv` pour le traitement des données.

LES MODULES.

On a parfois besoin d'utiliser des fonctions de Python qui ne sont pas chargées par défaut. Ces fonctions sont stockées dans des programmes Python appelées modules ou bibliothèques.

Quelques exemples de module :

- ▶ le module `math` qui contient les fonctions mathématiques usuelles
- ▶ le module `PIL` pour le traitement des photographies numériques.
- ▶ le module `folium` pour la géolocalisation.
- ▶ le module `csv` pour le traitement des données.
- ▶ le module `microbit` pour programmer une carte du même nom pour l'informatique embarquée.

LES MODULES

Pour utiliser une fonction d'un module on peut procéder de plusieurs façons :

LES MODULES

Pour utiliser une fonction d'un module on peut procéder de plusieurs façons :

- ▶ On utilise l'instruction `import ...` et pour utiliser une fonction du module, on doit la préfixer par le nom du module. Cette façon de faire est un peu compliquée mais elle a l'avantage d'indiquer le module utilisé.

```
>>> import math  
  
>>> math.sqrt(2)  
1.4142135623730951
```

LES MODULES

- ▶ On importe uniquement la fonction dont on a besoin avec l'instruction `from ... import ...`.

```
>>> from math import sqrt
>>> sqrt(2)
1.4142135623730951
```

LES MODULES

- ▶ On importe uniquement la fonction dont on a besoin avec l'instruction `from ... import ...`.

```
>>> from math import sqrt
>>> sqrt(2)
1.4142135623730951
```

- ▶ On peut importer toutes les fonctions du module avec l'instruction `from ... import*`

```
>>> from math import*
>>> sqrt(2)
1.4142135623730951
```