

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Python

La programmation procédurale

Olivier SNOECK

Qu'est-ce que le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Qu'est-ce que le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- Python est le choix préféré des développeurs, des ingénieurs, des "data scientists" et des bricoleurs : il y a beaucoup de bibliothèques...

Qu'est-ce que le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

- Python est le choix préféré des développeurs, des ingénieurs, des "data scientists" et des bricoleurs : il y a beaucoup de bibliothèques...
- Le langage Python a été créé en 1989 aux Pays-Bas. Le nom Python est un hommage à la série télévisée *Monty Python's Flying Circus*

Qu'est-ce que le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- Python est le choix préféré des développeurs, des ingénieurs, des "data scientists" et des bricoleurs : il y a beaucoup de bibliothèques...
- Le langage Python a été créé en 1989 aux Pays-Bas. Le nom Python est un hommage à la série télévisée *Monty Python's Flying Circus*
- Le langage Python est multi-plateformes : Windows, Mac OS X, Linux, Android, iOS. De la Raspberry Pi aux supercalculateurs...

Qu'est-ce que le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

- Python est le choix préféré des développeurs, des ingénieurs, des "data scientists" et des bricoleurs : il y a beaucoup de bibliothèques...
- Le langage Python a été créé en 1989 aux Pays-Bas. Le nom Python est un hommage à la série télévisée *Monty Python's Flying Circus*
- Le langage Python est multi-plateformes : Windows, Mac OS X, Linux, Android, iOS. De la Raspberry Pi aux supercalculateurs...
- Le langage Python est gratuit, avec 2 grandes versions : la version 2 (Python 2.7.2 : obsolète) et la version 3 (Python 3.12) : utilisez la version 3 sauf dans de très rares cas (utilisation de bibliothèques particulières)

Qu'est-ce que le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

- Python est le choix préféré des développeurs, des ingénieurs, des "data scientists" et des bricoleurs : il y a beaucoup de bibliothèques...
- Le langage Python a été créé en 1989 aux Pays-Bas. Le nom Python est un hommage à la série télévisée *Monty Python's Flying Circus*
- Le langage Python est multi-plateformes : Windows, Mac OS X, Linux, Android, iOS. De la Raspberry Pi aux supercalculateurs...
- Le langage Python est gratuit, avec 2 grandes versions : la version 2 (Python 2.7.2 : obsolète) et la version 3 (Python 3.12) : utilisez la version 3 sauf dans de très rares cas (utilisation de bibliothèques particulières)

Pourquoi le langage Python ?

Introduction

Python, c'est quoi ?

**Python, ses
avantages et
inconvénients**

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Pourquoi le langage Python ?

- Le langage Python est un langage haut niveau : il est détaché de la couche matériel. Les langages de bas niveau sont très proches de la structure du μ P, des périphériques (mémoires, bus,...) : Python est proche de la langue humaine (en anglais)

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

Pourquoi le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- Le langage Python est un langage haut niveau : il est détaché de la couche matériel. Les langages de bas niveau sont très proches de la structure du μ P, des périphériques (mémoires, bus,...) : Python est proche de la langue humaine (en anglais)
- C'est un langage interprété (et pas compilé : C \rightarrow .exe). Un script Python n'a pas besoin d'être compilé pour être exécuté, contrairement à des langages comme le C ou le C++.

Pourquoi le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- Le langage Python est un langage haut niveau : il est détaché de la couche matériel. Les langages de bas niveau sont très proches de la structure du μ P, des périphériques (mémoires, bus,...) : Python est proche de la langue humaine (en anglais)
- C'est un langage interprété (et pas compilé : C \rightarrow .exe). Un script Python n'a pas besoin d'être compilé pour être exécuté, contrairement à des langages comme le C ou le C++.
- Il est orienté objet. Avec Python, tout est un objet... (prochain cours)

Pourquoi le langage Python ?

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- Le langage Python est un langage haut niveau : il est détaché de la couche matériel. Les langages de bas niveau sont très proches de la structure du μP , des périphériques (mémoires, bus,...) : Python est proche de la langue humaine (en anglais)
- C'est un langage interprété (et pas compilé : C \rightarrow .exe). Un script Python n'a pas besoin d'être compilé pour être exécuté, contrairement à des langages comme le C ou le C++.
- Il est orienté objet. Avec Python, tout est un objet... (prochain cours)
- Python, plus simple à prendre en main, éloigné du matériel, utilise beaucoup de bibliothèques mais souvent l'exécution d'un script Python est plus longue qu'en programme en C (ou C++) et compilé en langage machine : le langage machine est directement lu par le μP

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un interpréteur lit le code source du programme comme s'il a été écrit par le programmeur, analyse le code et interprète les instructions à la volée : les unes derrière les autres.

Python est un interpréteur : on peut donc l'utiliser en mode interactif dans lequel, il va exécuter ligne par ligne ce qui a été saisi

Exemple :

```
>>> x=12
```

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un interpréteur lit le code source du programme comme s'il a été écrit par le programmeur, analyse le code et interprète les instructions à la volée : les unes derrière les autres.

Python est un interpréteur : on peut donc l'utiliser en mode interactif dans lequel, il va exécuter ligne par ligne ce qui a été saisi

Exemple :

```
>>> x=12
```

```
>>> print(x)
```

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un interpréteur lit le code source du programme comme s'il a été écrit par le programmeur, analyse le code et interprète les instructions à la volée : les unes derrière les autres.

Python est un interpréteur : on peut donc l'utiliser en mode interactif dans lequel, il va exécuter ligne par ligne ce qui a été saisi

Exemple :

```
>>> x=12
```

```
>>> print(x)
```

```
12
```

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un interpréteur lit le code source du programme comme s'il a été écrit par le programmeur, analyse le code et interprète les instructions à la volée : les unes derrière les autres.

Python est un interpréteur : on peut donc l'utiliser en mode interactif dans lequel, il va exécuter ligne par ligne ce qui a été saisi

Exemple :

```
>>> x=12
```

```
>>> print(x)
```

```
12
```

```
>>> y=x+7
```

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un interpréteur lit le code source du programme comme s'il a été écrit par le programmeur, analyse le code et interprète les instructions à la volée : les unes derrière les autres.

Python est un interpréteur : on peut donc l'utiliser en mode interactif dans lequel, il va exécuter ligne par ligne ce qui a été saisi

Exemple :

```
>>> x=12
```

```
>>> print(x)
```

```
12
```

```
>>> y=x+7
```

```
>>> print(y)
```

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un interpréteur lit le code source du programme comme s'il a été écrit par le programmeur, analyse le code et interprète les instructions à la volée : les unes derrière les autres.

Python est un interpréteur : on peut donc l'utiliser en mode interactif dans lequel, il va exécuter ligne par ligne ce qui a été saisi

Exemple :

```
>>> x=12
```

```
>>> print(x)
```

```
12
```

```
>>> y=x+7
```

```
>>> print(y)
```

```
19
```

Python : interpréteurs (suite)

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Python : interpréteurs (suite)

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Le prompt (`>>>`) est la manière dont l'interpréteur Python pose la question : que veux-tu que je fasse ?

On peut lui demander d'afficher `print(x)` ou `print("x")` ou `print("Bonjour")`

Pour mettre fin à une conversation Python, soit on ferme la fenêtre, ou mieux, on fait `>>> quit()`

Python est un langage de haut niveau (par rapport au processeur) conçu afin d'être simple à lire et à écrire pour les humains et simple à lire et à traiter pour une machine. Les autres langages de haut niveau sont Java, C++, PHP, Ruby, Basic, JavaScript,...

Un langage de haut niveau n'est pas compris par le CPU (Central Processing Unit) qui lui, ne comprend que le langage machine constitué de 0 et de 1

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

**Python :
interpréteurs et
compilateurs**

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

- un compilateur doit stocker l'ensemble du programme dans un fichier et ensuite, un processus de traduction du code source en langage machine (avec des 0 et des 1) est effectué. C'est le rôle du compilateur de traduire le programme en langage machine
- L'interpréteur Python est écrit dans un langage de bas niveau : le langage C

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Il existe plusieurs solutions pour coder en Python telles que :

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Il existe plusieurs solutions pour coder en Python telles que :

- Directement dans le shell

```
olivier@oliviers:~$ python3
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> ma_variable=5
>>> print("Bonjour les amis")
Bonjour les amis
>>> print(ma_variable)
5
>>> exit()
```

- On y voit la création de la variable `ma_variable`
- Cette variable prend 5 (son type est donc entier (integer))
- On affiche un texte en utilisant les guillemets
- On affiche le contenu de la variable `ma_variable`
- On sort du shell `exit()`

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Il existe plusieurs solutions pour coder en Python telles que :

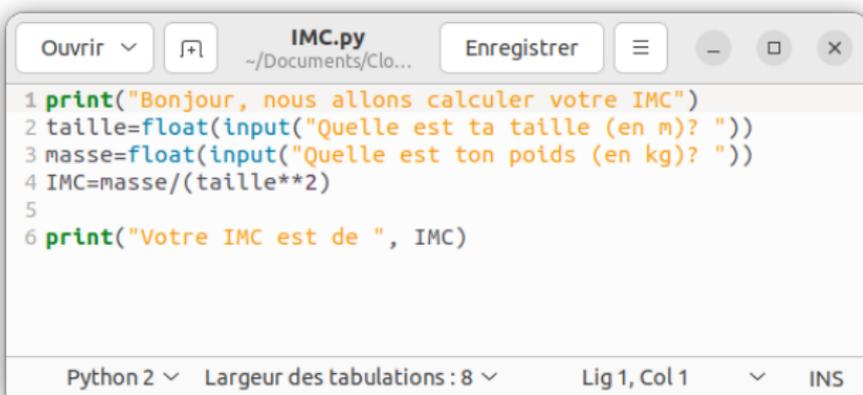
- Directement dans le shell

```
olivier@oliviers:~$ python3
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> ma_variable=5
>>> print("Bonjour les amis")
Bonjour les amis
>>> print(ma_variable)
5
>>> exit()
```

- On y voit la création de la variable `ma_variable`
- Cette variable prend 5 (son type est donc entier (integer))
- On affiche un texte en utilisant les guillemets
- On affiche le contenu de la variable `ma_variable`
- On sort du shell `exit()`
- **Le shell est pratique pour tester les lignes les une après les autres mais en aucun cas, ce ne sera un programme**

Il existe plusieurs solutions pour coder en Python telles que :

- Dans un éditeur de texte (Notepad++ sous windows, Gedit sous linux)



The screenshot shows a text editor window titled "IMC.py" with the following Python code:

```
1 print("Bonjour, nous allons calculer votre IMC")
2 taille=float(input("Quelle est ta taille (en m)? "))
3 masse=float(input("Quelle est ton poids (en kg)? "))
4 IMC=masse/(taille**2)
5
6 print("Votre IMC est de ", IMC)
```

The code is color-coded: strings are in orange, keywords like `print` and `float` are in green, and variables and operators are in blue. The editor interface includes a menu bar with "Ouvrir", "Enregistrer", and window control buttons. The status bar at the bottom shows "Python 2", "Largeur des tabulations : 8", "Lig 1, Col 1", and "INS".

On remarque la colorisation syntaxique

On voit comment écrire sur le moniteur

On voit comment poser une question à l'utilisateur avec son changement de type (ici, float)

On voit la puissance de 2 avec l'utilisation de 3 variables

On voit l'affichage d'une chaîne de caractère et du contenu d'une variable

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

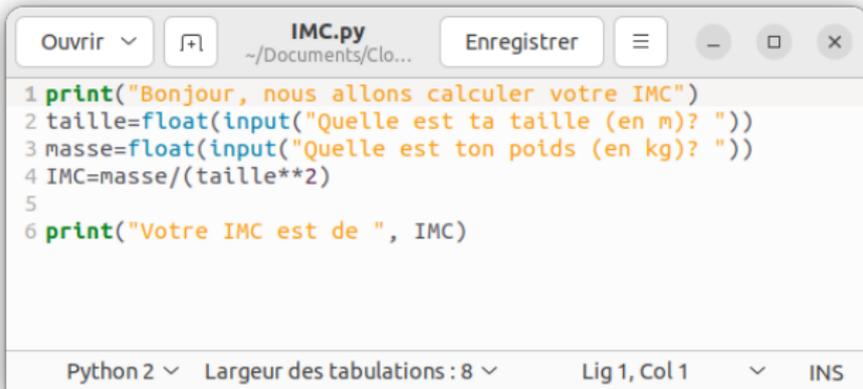
La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

Il existe plusieurs solutions pour coder en Python telles que :

- Dans un éditeur de texte (Notepad++ sous windows, Gedit sous linux)



```
1 print("Bonjour, nous allons calculer votre IMC")
2 taille=float(input("Quelle est ta taille (en m)? "))
3 masse=float(input("Quelle est ton poids (en kg)? "))
4 IMC=masse/(taille**2)
5
6 print("Votre IMC est de ", IMC)
```

Python 2 | Largeur des tabulations : 8 | Lig 1, Col 1 | INS

Sous **Linux**, l'exécution du programme IMC.py se fait par la commande "python3 IMC.py" dans le terminal (ctrl + alt + T)
Sous **Windows**, il faut utiliser un logiciel (IDLE par exemple)

Il existe aussi Atom, [Visual Studio Code](#), Sublime Text, emacs, vim, geany... sous Windows ou Linux : attention aux paramètres....

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

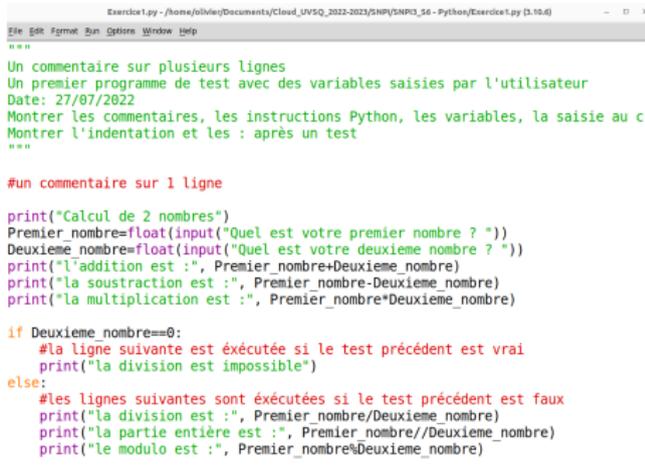
La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Il existe plusieurs solutions pour coder en Python telles que :

- Dans un Environnement de Développement Intégré (EDI ou IDE) comme **IDLE Python** (IDLE : Integrated Development and Learning Environment.)



```
Exercice1.py - /home/olivier/Documents/CLOUD_UV5Q_2022-2023/SNRP/SNRP3_56 - Python/Exercice1.py (3.10.6)
File Edit Format Run Options Window Help
"""
Un commentaire sur plusieurs lignes
Un premier programme de test avec des variables saisies par l'utilisateur
Date: 27/07/2022
Montrer les commentaires, les instructions Python, les variables, la saisie au c
Montrer l'indentation et les : après un test
"""

#un commentaire sur 1 ligne

print("Calcul de 2 nombres")
Premier_nombre=float(input("Quel est votre premier nombre ? "))
Deuxieme_nombre=float(input("Quel est votre deuxieme nombre ? "))
print("L'addition est :", Premier_nombre+Deuxieme_nombre)
print("la soustraction est :", Premier_nombre-Deuxieme_nombre)
print("la multiplication est :", Premier_nombre*Deuxieme_nombre)

if Deuxieme_nombre==0:
    #la ligne suivante est exécutée si le test précédent est vrai
    print("la division est impossible")
else:
    #les lignes suivantes sont exécutées si le test précédent est faux
    print("la division est :", Premier_nombre/Deuxieme_nombre)
    print("la partie entière est :", Premier_nombre//Deuxieme_nombre)
    print("le modulo est :", Premier_nombre%Deuxieme_nombre)
```

Introduction

Python, c'est quoi ?

Python, ses avantages et inconvénients

Python : interpréteurs et compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons possibles

Structures conditionnelles : if - elif - else

Structures conditionnelles : switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des erreurs et des exceptions

Exercice 6

Exercice 7

Il existe plusieurs solutions pour coder en Python telles que :

- Dans un éditeur et interpréteur en ligne : replit.com puis [create](https://create.replit.com) puis choisissez **Python** (ou autre)

The screenshot shows a web browser window with the URL <https://replit.com/@oscar/IMCmain.py>. The IDE interface includes a file explorer on the left showing a file named 'main.py' and a 'Tools' panel with various utility icons. The main editor area contains the following Python code:

```
1 print("Bonjour, nous allons calculer votre IMC")
2 taille=float(input("Quelle est ta taille (en m) :"))
3 masse=float(input("Quelle est ton poids (en kg) :"))
4 IMC=masse/(taille**2)
5
6 print("Votre IMC est de :", IMC)
```

The terminal window on the right shows the output of the program:

```
Bonjour, nous allons calculer votre IMC
Quelle est ta taille (en m) :
```

Exercice : Calcul de l'IMC

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvenients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Programmez et testez le calcul de l'IMC (Indice de Masse Corporel) suivant :

```
print("Bonjour, nous allons calculer votre IMC")
taille=float(input("Quelle est ta taille (en m)? "))
masse=float(input("Quelle est ton poids (en kg)? "))
IMC=masse/(taille**2)
```

```
print("Votre IMC est de ", IMC)
```

Notez que Python comprend la saisie de l'utilisateur comme sous la forme d'une chaîne de caractères.

Le "float" permet de demander à python de changer la chaîne de caractères saisie par l'utilisateur en nombre à virgule (float)

La variable **IMC** est donc obligatoirement un float (c'est une division entre nombres flottants)

Les comparaisons possibles

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE
Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

`==` égal à
`!=` différent de
`>` strictement supérieur à
`>=` supérieur ou égal à
`<` strictement inférieur à
`<=` inférieur ou égal à

Les structures de contrôle : if - elif - else

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

La structure **if** est une structure conditionnelle en informatique permettant de prendre en compte des **conditions**.

Si la **condition** est **vraie**, alors la suite de la structure comprise après les : (et indentée) sera exécutée.

Si la **condition n'est pas vraie**, alors la structure comprise après les : (et indentée) ne sera pas exécutée. La structure **elif** (ou **else**) qui suit la structure **if** le sera.

Les structures **elif** (et **else**) sont des structures facultatives qui, lorsqu'elles sont présentes, sont exécutées si et seulement si la condition est fausse.

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

```
1  if test1 == True:
2      # action1 à exécuter si le test1 est vrai
3      # ...
4      # action7 à exécuter si le test1 est vrai
5  elif test2 == False:
6      '''
7      action8 à exécuter si le test2 est faux
8      ...
9      action15 à exécuter si le test2 est faux
10     '''
11  #...
12  elif test3:
13      # ...
14  else:
15      # action75 à exécuter si aucun test est vérifié
16      # ...
17      # action77 à exécuter si aucun test est vérifié
```

Note : Les structures *elif* (et *else*) sont des structures facultatives..

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

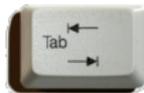
Exercice 6

Exercice 7

```
1 x,y=8,4
2 if x == y:
3     print ("x et y sont égaux")
4 elif x > y:
5     print ("x est plus grand que y")
6 else:
7     print ("x est plus petit que y")
```

L'indentation est vitale pour que le test fonctionne

L'indentation est faite automatiquement derrière les `:` en appuyant sur la touche "enter"



L'indentation se fait également avec la touche de tabulation à gauche du clavier

Les structures de contrôle : switch-case

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

La structure **switch-case** est une structure conditionnelle en informatique permettant de faire des **conditions multiples**. Elle permet d'éviter d'imbriquer des structures if-else dans certaines situations..

La structure **switch-case** est très employée lorsqu'il y a plusieurs valeurs possibles pour une variable. Dans une structure **switch-case**, switch porte la variable qui est soumise à la condition. La structure case porte la condition. Chaque case est séparé par le mot-clé break. Malheureusement, la structure **switch-case** n'existe pas en langage Python mais, de puis la version Python 3.10, il existe la structure **match-case**

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

```
1 x,y=8,4
2 soustraction= x-y
3 match soustraction:
4     case 0:
5         print ("x et y sont égaux")
6         break
7     case -1:
8         print ("x est plus petit que y")
9         break
10    case -2:
11        print ("x est plus petit que y")
12        break
13    case -3:
14        print ("x est plus petit que y")
15        break
16    case -4:
17        print ("x est plus petit que y")
18        break
19    case 1:
20        print ("x est plus grand que y")
21        break
22    case 2:
23        print ("x est plus grand que y")
24        break
25    case 3:
26        print ("x est plus grand que y")
27        break
28    case 4:
29        print ("x est plus grand que y")
30        break
31
```

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Demandez 2 nombres (des flottants) à l'utilisateur de votre programme et renvoyez lui les résultats de l'addition entre ces 2 nombres (2 variables), de la soustraction, de la multiplication et de la division (/) mais aussi la partie entière de la division (//) et son reste (le modulo %).

- Essayez d'obtenir une fenêtre de résultats sans fautes et agréable
- Attention à ne jamais diviser par 0

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Demandez 2 phrases (ou 2 mots) à l'utilisateur et tentez l'addition de ces 2 variables. Qu'observez vous ?

C'est ce qu'on appelle la concaténation

- Essayez d'obtenir une fenêtre de résultats sans fautes et agréable à lire

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Testez ce programme :

```
1 a= "les"  
2 b=" "  
3 c="frites"  
4 print("Salut, j'aime ",a+b+c)  
5
```

La fonction print permet d'afficher des chaînes de caractères et des variables

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Réalisez un programme qui demande le prénom et l'âge de l'utilisateur et qui lui renvoie une phrase du type :

Bonjour, vous êtes et vous avez ... ans
L'âge peut aussi être affiché en binaire

Modifiez ce programme afin de donner l'année de naissance de la personne (attention, il faut connaître le mois actuel grâce à l'horloge de l'ordinateur par exemple). Il y a quelques recherches à faire...

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Réalisez un programme qui calcul la moyenne des notes d'un étudiant.

Les notes ne peuvent pas être négative (si elles sont mal saisies, on affiche un message d'erreur) et les notes ne peuvent pas être au dessus de 20 (si elles sont mal saisies, on affiche un message d'erreur).

Le programme boucle en continu et affiche donc la moyenne à chaque saisie de l'utilisateur.

Pour arrêter le programme, l'utilisateur devra appuyer sur la touche P

3 solutions à obtenir :

- 1 sans utiliser de liste : avec des boucles
- 2 en utilisant une liste
- 3 en utilisant la librairie numpy

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénientsPython :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures
de contrôleLes comparaisons
possiblesStructures
conditionnelles : if -
elif - elseStructures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Même si une instruction ou une expression est syntaxiquement correcte, elle peut générer une erreur lors de son exécution. Les erreurs détectées durant l'exécution sont appelées des **exceptions** et ne sont pas toujours fatales. Dans notre cas, gérer l'exception de la saisie d'une note négative ou d'un caractère autre qu'un chiffre peut être intéressant.

```
1 try:
2     x = int(input("Saisissez la note obtenue: "))
3     if x < 0 or x > 20:
4         print("La note doit être entre 0 et 20.")
5         continue
6         break
7 except ValueError:
8     print("Oops! That was no valid number. Try again...")
9
```

Les exceptions sont listées sur ce lien

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Réalisez un programme qui calcul la moyenne des notes d'un étudiant.

Les notes ne peuvent pas être négative (si elles sont mal saisies, on affiche un message d'erreur) et les notes ne peuvent pas être au dessus de 20 (si elles sont mal saisies, on affiche un message d'erreur).

Le programme boucle en continu et affiche donc la moyenne à chaque saisie de l'utilisateur.

Pour arrêter le programme, l'utilisateur devra appuyer sur la touche P. Dans cet exercice 6, utilisez la gestion des erreurs....

Introduction

Python, c'est quoi ?

Python, ses
avantages et
inconvénients

Python :
interpréteurs et
compilateurs

Python, et ses IDE

Exercice : IMC

Les structures de contrôle

Les comparaisons
possibles

Structures
conditionnelles : if -
elif - else

Structures
conditionnelles :
switch-case

Des exercices

Exercice 1

Exercice 2

Exercice 3

Exercice 4

Exercice 5

La gestion des
erreurs et des
exceptions

Exercice 6

Exercice 7

Je vous propose d'utiliser le programme de calcul d'IMC déjà effectuée et d'afficher la classification de l'IMC selon ce lien

Exemple : Vous avez un IMC de $18\text{kg}/\text{m}^2$ qui vous indique que votre poids est normal