

Informatique & Programmation

#4 Programmation python

par David Albert

Table des matières

01 Conditions

if. else. elif.

02 Boucles

for. while.

03 Les fonctions

Définition de fonctions. Paramètres et valeur de retour.

04 Packages

Réutiliser le code existant.

01

Conditions

Conditions

1. if / else

```
if (cond):  
    # code si vrai  
else:  
    # code si faux
```

2. if / else if / else

```
if (cond):  
    # code si vrai  
elif (cond2):  
    # code si vrai  
else:  
    # code si faux
```

Exemple

```
temperature = input("Entrez une valeur : ")  
if (temperature >= 100)  
{  
    print("L'eau bout !")  
}
```

i

La notation **cond** dans les exemples ci-contre représente une expression quelconque qui renvoie un booléen.

Exemples :

- `if True:`
- `if is_winner:`
- `if (age < 30):`
- `if (age < 20 and name == "Mathéo"):`
- `if "John" in names:`

02

Boucles

Boucles

1. for

```
for i in range(10):  
    # ...  
    # code ici  
    # ...
```

2. while

```
while (cond):  
    # ...  
    # code ici  
    # ...
```

Exemple

```
secret = "azezesx"  
motDePasse = input("Entrez le mot de passe : ")  
while (motDePasse != secret or agePersonne <= 3)  
{  
    print("Accès refusé")  
    motDePasse = input("Réessayez : ")  
}
```

03

Fonctions

Déclarer une fonction

Une fonction est définie grâce :

- au mot-clé `def` en python
- à un identifiant
- à des paramètres (optionnels)
- à un type de retour et une valeur de retour

Exemple 1: Calcul de l'aire d'un rectangle

```
def aire_rectangle(longueur: int, largeur: int) -> int:  
    return longueur * largeur
```

Exemple 2: Affichage des données d'une classe

```
def display_user_data(user: User) -> None:  
    print("--- User Data ---")  
    print("name=", user.name)  
    print("age=", user.age)
```

i

Une fonction qui ne retourne aucune valeur est généralement appelée **procédure**.

04

Packages & modules

Réutiliser du code

En plus d'être simple et intuitif, le langage de programmation Python possède l'avantage d'avoir une très large communauté.

Ainsi, pas besoin de tout réimplémenter. Si vous avez besoin d'une fonctionnalité, quelqu'un l'aura probablement déjà implémenté et partagé avant vous.

Un **module** est une collection de fonctions et méthodes qui peuvent être réutilisés dans une autre partie du code.

Un **package** est un ensemble de modules munis d'une documentation et conçus pour des besoins spécifiques.

Exemples de package python : numpy, pandas, plotly, Django, Flask, PyTorch, Scikit Learn, ...

Importer des fonctions & méthodes

Importer tout un module

```
# Importer la library math  
import math  
  
# Retourne la racine carré de 9  
print(math.sqrt(9))
```

Importer une fonction spécifique d'un module

```
# Importer une fonction de la library math  
from math import sqrt  
  
# Retourne la racine carré de 9  
print(sqrt(9))
```