

Trucs et astuces

1. 1. Récupérer une entrée clavier dans un programme

Entrée clavier

En Python, la fonction `input` permet de faire une pause dans l'exécution du programme et attend que l'utilisateur fasse une entrée au clavier (terminée bien entendu par la touche `ENTER`).

Cette fonction renvoie cette entrée systématiquement sous le type `str`. Il faudra penser à la convertir en `int` ou `float` si la saisie attendue est numérique.

On peut, de façon facultative, préciser en argument une chaîne de caractère qui sera affichée au préalable (pratique pour poser une question).

Exemples

Ces exemples sont à tester dans un IDE.

Saisie d'un texte (sans argument)

Script Python

```
>>> r = input()
salut
>>> r
'salut'
>>> type(r)
<class 'str'>
>>>
```

Saisie d'un texte (avec argument)

Script Python

```
>>> r = input("Quel âge avez-vous? ")
Quel âge avez-vous? 15
>>> r
'15'
>>> type(r)
<class 'str'>
>>>
```

Conversion de la saisie en `int`

Script Python

```
>>> r = int(input("Quel âge avez-vous?"))
Quel âge avez-vous? 15
>>> r
15
>>> type(r)
<class 'int'>
>>>
```

2. 2. L'incrémentation d'une variable.

«*Incrémenter*» une variable signifie l'augmenter.

Imaginons une variable appelée `compteur`. Au démarrage de notre programme, elle est initialisée à la valeur 0.

Script Python

```
>>> compteur = 0
```

Considérons qu'à un moment du programme, cette variable doit être modifiée, par exemple en lui ajoutant 1.

En Python, cela s'écrit :

 **Script Python**

```
>>> compteur = compteur + 1
```

Observée avec des yeux de mathématicien, la précédente instruction est une horreur.



Vue avec des yeux d'informaticien, voilà comment est interprétée la commande

 **Script Python**

```
>>> compteur = compteur + 1
```

1. On évalue la partie droite de l'égalité, donc l'expression `compteur + 1`.
2. On va donc chercher le contenu de la variable `compteur`. Si celle-ci n'existe pas, un message d'erreur est renvoyé.
3. On additionne 1 au contenu de la variable `compteur`.
4. On écrase le contenu actuel de la variable `compteur` avec la valeur obtenue au 3.

À la fin de ces opérations, la variable `compteur` a bien augmenté de 1.

Cette procédure d'**incrément** est très très classique, il faut la maîtriser parfaitement !

Syntaxe classique et syntaxe Pythonesque

L'incrémentation d'une variable `compteur` s'écrira donc en Python :

Script Python

```
>>> compteur = compteur + 1
```

Mais il existe aussi une syntaxe particulière, un peu plus courte :

Script Python

```
>>> compteur += 1
```

Cette syntaxe peut se ranger dans la catégorie des **sucres syntaxiques** : c'est bien de la connaître, c'est amusant de s'en servir, mais son utilisation n'est en rien obligatoire et peut avoir un effet néfaste, celui d'oublier réellement ce qu'il se passe derrière.

3. 3. Utilisation du module `PIL`

Le module `PIL` permet la création et la manipulation d'images.

Voir [ici](#) .