
I3 - Écrire un script

Un **script** est une suite d'instruction dont l'exécution permet de réaliser un objectif.

En général, un script est composé de trois parties :

- ★ Dans la première partie, on commence par demander à l'utilisateur d'entrer des informations qui seront nécessaires à l'exécution du script. Pour cela, on utilise la fonction **input**. Il est alors important de penser à convertir **tout de suite** les données numériques en entiers (**int**) ou en flottants (**float**) :

```
texte = input("Entrez du texte : ")
nombre_entier = int(input("Entrez un nombre entier : "))
nombre_reel = float(input("Entrez un nombre réel : "))
```

- ★ Dans la deuxième partie, on effectue des calculs grâce aux données entrées par l'utilisateur : c'est la partie algorithmique qui résout le problème posé. On n'hésitera pas à introduire d'autres variables ou des fonctions afin de limiter **la redondance du code** (les répétitions).
- ★ Dans la dernière partie, on va afficher le ou les résultats de nos calculs en utilisant si possible des phrases dynamiques (voir fiche **Afficher une phrase dynamique**).

Remarque :

- ★ Un script s'écrit **dans l'éditeur** mais, lors de son exécution, l'utilisateur répondra aux questions et verra les résultats du script dans le shell.
- ★ Pensez à bien répondre aux questions quand vous exécutez votre script ! Si vous avez l'impression que « Python ne répond plus » c'est qu'il attend probablement de nombreuses réponses que vous n'avez pas donné car vous avez validé plusieurs fois le script !

Exemple

```
poids = float(input("Entrez votre poids en kg : "))
taille = float(input("Entrez votre taille en mètres : "))
imc = poids / (taille ** 2)
print(f"Votre IMC est de {imc}.")
```

Erreurs fréquentes

► Oublier de convertir les données numériques

Par exemple, le code

```
1 poids = input("Entrez votre poids en kg : ")
2 taille = input("Entrez votre taille en mètres : ")
3 imc = poids / (taille ** 2)
4 print(f"Votre IMC est de {imc}.")
```

produit une erreur à la ligne numéro 3 alors que l'erreur est plutôt aux lignes 1 et 2.

► Ne pas enregistrer le résultat d'un input

Par exemple

```
input("Entrez votre nom : ")
```

est complètement inutile car le nom n'est pas enregistré!

- **S'étonner du fait que Python ne répond plus alors que ça fait 20 fois qu'on tape CTRL + ENTREE!**

Si on exécute une instruction contenant un appel à la fonction `input`, Python va attendre que l'on réponde à la question! Tant qu'on ne le fait pas, il ne répondra plus. Pire : si on continue d'exécuter le même code, les questions posées par les instructions `input` sont mises en attente et, si on répond à la première, la suivante nous sera ensuite posée.