

Demander des valeurs à l'utilisateur

La fonction `input()` est l'un des outils les plus basiques mais essentiels que vous rencontrerez en apprenant à programmer en Python. Elle vous permet d'interagir avec l'utilisateur en récupérant des données depuis la console. Dans cet article, nous allons explorer en détail la fonction `input()`, comment l'utiliser efficacement, et comment convertir les données entrées en utilisant `int()` et `float()`.

3GMS



Qu'est-ce que la Fonction `input()` ?

En Python, `input()` est une fonction intégrée qui permet à un programme de **recevoir des données entrées par l'utilisateur depuis le clavier**. Elle attend que l'utilisateur saisisse une ligne de texte, la valide en tapant la **touche "entrée"** et retourne cette ligne sous forme de **chaîne de caractères (string)**. Voici un exemple simple :

```
1 | nom_utilisateur = input("Entrez votre nom : ")
2 | print("Bonjour, " + nom_utilisateur + "!")
```

Dans cet exemple, `input("Entrez votre nom : ")` affiche le message "Entrez votre nom :" à l'utilisateur et **attend** que l'utilisateur entre son nom. La valeur entrée est ensuite stockée dans la variable `nom_utilisateur` sous forme de **chaîne de caractères** et affichée dans un message de salutation.

★ La fonction `input` retourne une chaîne de caractères GB **string**. Si vous voulez utiliser cette valeur pour effectuer des calculs, il faut d'abord la convertir.

Conversion avec `int()` et `float()`

Lorsque vous utilisez `input()` pour obtenir des données numériques de l'utilisateur, celles-ci sont **toujours stockées sous forme de chaînes de caractères (strings)**. Si vous avez besoin de travailler avec ces données en tant que nombres, vous devez les **convertir** en utilisant les fonctions `int()` ou `float()`.

- `int()` est utilisé pour convertir une chaîne en un nombre entier (int).
- `float()` est utilisé pour convertir une chaîne en un nombre décimal (float).

Voici des exemples de conversion avec `int()` et `float()` :

```
1 | age_texte = input("Entrez votre âge : ")
2 | age = int(age_texte) # Convertir la chaîne en un entier
```

```
3 |  
4 | prix_texte = input("Entrez le prix : ")  
5 | prix = float(prix_texte) # Convertir la chaîne en un nombre décimal (float)
```

L'utilisation de `int()` et `float()` permet de traiter les données entrées par l'utilisateur comme des nombres, ce qui est souvent nécessaire pour effectuer des calculs ou manipuler des valeurs numériques.

Si l'utilisateur entre des nombres invalides, cela provoquera le plantage de votre programme. Pour l'instant on peut s'en contenter. Partons du principe pour ce cours que l'on reçoit des données valides. Sachez qu'il existe des moyens d'éviter ces problèmes.

Conclusion

La fonction `input()` permet d'interagir avec l'utilisateur et obtenir des données d'entrée dans vos programmes Python. Lorsque vous travaillez avec des données numériques, n'oubliez pas de les convertir en utilisant `int()` ou `float()` pour les manipuler correctement. Avec `input()` et la conversion appropriée, vous pouvez créer des applications interactives et conviviales qui répondent aux besoins de vos utilisateurs.

Quelques exercices

- 1. Salutation personnalisée:** Écris un programme qui demande à l'utilisateur son prénom et affiche : `Bonjour, [prénom] !`
- 2. Présentation complète:** Demande à l'utilisateur son prénom et son nom, puis affiche une phrase complète : `Tu t'appelles [prénom] [nom].`
- 3. Réponse humoristique:** Demande à l'utilisateur son animal préféré et réponds : `[animal] ? C'est un choix intéressant !`
- 4. Couleur préférée:** Demande la couleur préférée de l'utilisateur et affiche une phrase : `[couleur] est une belle couleur !`
- 5. Deviner une profession:** Demande à l'utilisateur ce qu'il aimerait faire comme métier et réponds : `Devenir [métier] ? Super idée !`

- 1. Double d'un nombre:** Demande un nombre à l'utilisateur et affiche son double.
- 2. Addition simple:** Demande deux nombres à l'utilisateur et affiche leur somme.
- 3. Calcul de périmètre:** Demande la longueur d'un côté d'un carré et calcule son périmètre.
- 4. Multiplication:** Demande deux nombres à l'utilisateur et affiche leur produit.
- 5. Moyenne de trois nombres:** Demande trois nombres et calcule leur moyenne.
- 6. Temps en secondes:** Demande un temps en minutes et affiche le nombre de secondes.
- 7. Conversion km → miles:** Demande une distance en kilomètres et affiche la conversion en miles : `1 km ≈ 0.621371 miles`

8. Conversion Celsius → Fahrenheit: Demande une température en Celsius et affiche la conversion en Fahrenheit: $F = C \times 9/5 + 32$

Vidéo à la une à regarder sur Youtube:  <http://youtu.be/UlSmDST7zfk>