Utilisation des f-strings en python

Objectif : Découvrir et utiliser les **f-strings** (formatted strings) pour afficher du texte contenant des variables de façon simple et lisible.

Table des matières

1. Les f-strings en Python (formatted string literals)....1
a. Qu'est-ce qu'un f-string ?....1
b. Exemples de base....1
c. Pourquoi utiliser les f-strings ?....1
d. Expressions dans les f-strings....2
e. Formatage de nombres....2
f. F-strings multilignes....2
g. Attention....2
h. À retenir....2
2. Exercices....3
a. Exercice 1 – Premiers pas avec les f-strings....3
b. Exercice 2 – Calculs dans les f-strings....3
c. Exercice 3 – Alignement et formatage....3
d. Défi – Crée ta propre phrase dynamique....3

1. Les f-strings en Python (formatted string literals)

a. Qu'est-ce qu'un f-string ?

Un **f-string** (ou "formatted string") est une manière **simple, rapide et lisible** d'insérer des **valeurs de variables** ou des **expressions** directement dans une chaîne de caractères.

🔧 Syntaxe :

```
f"texte {expression} suite du texte"
```

On place un **f** devant la chaîne, puis on écrit les variables ou expressions entre **accolades** `{}`.

b. Exemples de base

```
nom = "Alice"
age = 20

print(f"Bonjour {nom}, tu as {age} ans.")
```

✅ Affichage :

```
Bonjour Alice, tu as 20 ans.
```

c. Pourquoi utiliser les f-strings ?

Avant les f-strings, on faisait :

```
# Concaténation peu lisible :
print("Bonjour " + nom + ", tu as " + str(age) + " ans.")

# Méthode format() plus propre, mais plus lourde :
print("Bonjour {}, tu as {} ans.".format(nom, age))
```

Les f-strings rendent le code plus court, plus clair et plus puissant.

d. Expressions dans les f-strings

On peut **faire des calculs directement** dans les `{}` :

```
a = 7
b = 3
print(f"La somme de {a} et {b} est {a + b}.")
```

✅ Résultat :

```
La somme de 7 et 3 est 10.
```

e. Formatage de nombres

Les f-strings permettent aussi de **formater les nombres** :

Format	Signification	Exemple	Résultat
`8.2f`	8 caractères dont 2 décimales	`{pi:8.2f}`	3.14
`:>8`	aligné à droite sur 8 caractères	`{val:>8}`	`123`
`:<10`	aligné à gauche sur 10 caractères	`{nom:<10}`	`Alice`

Exemple combiné :

```
pi = 3.14159265
print(f"Valeur de pi : {pi:7.3f}")
```

✅ Résultat :

```
Valeur de pi :   3.142
```

f. F-strings multilignes

Tu peux aussi utiliser des f-strings dans des chaînes multilignes :

```
nom = "Sam"
age = 25

texte = f"""
Nom : {nom}
Âge : {age}
"""
print(texte)
```

g. Attention

- Le `f` doit être placé **juste avant les guillemets**.
- Les **accolades** ne doivent contenir que des **expressions valides** en Python (pas de déclaration, pas de `import`, etc.).

h. À retenir

- Les f-strings sont disponibles depuis **Python 3.6**.
- Ils rendent le code plus **lisible**, plus **concis**, et permettent d'**insérer facilement des expressions** dans les chaînes.
- Syntaxe : `f"texte {variable} autre texte {calcul}"`

2. Exercices

a. Exercice 1 – Premiers pas avec les f-strings

1. Complète le code suivant pour afficher correctement les informations de l'utilisateur :

```
prenom = "Lucas"
ville = "Lyon"
age = 20

# À compléter :
print(f"...")
```

✅ **Attendu :**

```
Bonjour Lucas, tu as 20 ans et tu habites à Lyon.
```

b. Exercice 2 – Calculs dans les f-strings

2. Observe et complète :

```
a = 12
b = 8

# À compléter :
print(f"La somme de {a} et {b} vaut ...")
print(f"Leur produit est ...")
```

✅ **Attendu :**

```
La somme de 12 et 8 vaut 20
Leur produit est 96
```

c. Exercice 3 – Alignement et formatage

3. Complète ce tableau d'affichage avec des f-strings et des largeurs fixes :

```
produit1 = "Clavier"
prix1 = 49.9
produit2 = "Souris"
prix2 = 19.5

print(f" ...à compléter ...") # 1ere col sur 10 car ; 2 ème col sur 8 car
print("-" * 22)
print(f" ...à compléter ...")
print(f" ...à compléter ...")
```

✅ **Attendu :**

```
Produit    |  Prix (€)
----------------------
Clavier    |    49.90
Souris     |    19.50
```

d. Défi – Crée ta propre phrase dynamique

4. Invente une phrase contenant **au moins deux variables**, **un calcul** et **une chaîne formatée avec un f-string**. Exemple :

```
nom = "Emma"
points = 14
total = 16
```

Afficher par exemple :

Emma a obtenu une note de **14/16**, ce qui correspond à **17.5**/20 ou encore **87.5** % de réussite.

En utilisant les variables pour afficher ce qui est en gras dans la phrase.