

# PROGRAMMATION - CHAÎNES DE CARACTÈRES

## 9.1 Introduction

Les chaînes de caractères (type `str`) en Python sont des séquences de caractères qui peuvent être utilisées pour stocker des textes. Les chaînes de caractères sont définies entre apostrophes (`'`), guillemets simples (`"`), ou guillemets triples (`"""`).

Listing 9.1 – Chaînes de caractères

```
1 str1 = "Ceci me permet d'écrire l'apostrophe."  
2 str2 = 'Ceci me permet de "placer" le guillemet.'  
3 str3 = """Ceci me permet d'écrire  
4 sur plusieurs lignes """
```

## 9.2 Boite à outils

### 9.2.1 Opérateurs

L'opérateur `+` permet de concaténer des chaînes de caractères. L'opérateur `*` permet de répéter une chaîne de caractères plusieurs fois.

Listing 9.2 – Opérateurs

```
1 chaine1 = "Bonjour "  
2 chaine2 = "toi !"  
3 chaine3 = chaine1 + chaine2  
4 print(chaine3)                # affiche "Bonjour toi !"  
5  
6 chaine4 = "Coucou "  
7 chaine5 = chaine4 * 3  
8 print(chaine5)                # affiche "Coucou Coucou Coucou "
```

### 9.2.2 Fonction `len()`

En Python, la `len()` fonction intégrée peut être utilisée pour déterminer la longueur d'un objet. Il peut être utilisé pour calculer la longueur de chaînes, de listes, d'ensembles et d'autres objets dénombrables.

Listing 9.3 – Fonction len()

---

```

1 longueur = len("Hello")
2 print("La longueur :", longueur)                                # affiche 5

```

---

### 9.2.3 in

La syntaxe `in` est utilisée pour déterminer si une lettre ou une sous-chaîne existe dans une chaîne.

Elle renvoie `True` si une correspondance est trouvée, sinon `False` est renvoyée.

Listing 9.4 – in

---

```

1 jeu = "Popular Nintendo Game: Mario Kart"
2
3 if "l" in jeu:
4     print("l est dans la chaîne jeu.")
5 else:
6     print("l n'est pas dans la chaîne jeu.")

```

---

### 9.2.4 Indexation et découpage des chaînes

Un seul caractère peut être accédé avec la notation entre crochets `[index]`, ou une sous-chaîne peut être accédée en utilisant le découpage `[start:end]`.

L'indexation avec des nombres négatifs compte à partir de la fin de la chaîne.

Listing 9.5 – indexation

---

```

1 mot = 'orange'
2 #      012345
3
4 print(mot[0])           # => 'o'
5 print(mot[1])           # => 'r'
6 print(mot[4:6])         # => 'ge'
7 print(mot[:4])          # => 'oran'
8 print(mot[-1])          # => 'e'

```

---

### 9.2.5 Itérer la chaîne

Pour parcourir une chaîne en Python, la notation `for ... in` est utilisée.

Listing 9.6 – iteration

---

```

1 mot = "hello"
2 for c in mot:
3     print(c)

```

---

affiche chaque lettre du mot `"hello"` les unes après les autres.

### 9.2.6 Autres fonctions

`.lower()` renvoie une chaîne avec tous les caractères majuscules convertis en minuscules.

`.upper()` renvoie la chaîne avec tous les caractères minuscules convertis en majuscules.

Listing 9.7 – lower / upper

---

```

1 salutation = "Bienvenue chez Chili's"
2 print(salutation.lower())    # affiche bienvenue chez chili's

```

---

`.isalpha()` renvoie `True` si tous les caractères de la chaîne sont alphabétiques et qu'elle contient au moins un caractère, sinon `False`.

`.isdigit()` renvoie `True` si tous les caractères de la chaîne sont des chiffres et qu'elle contient au moins un caractère, sinon `False`.

Listing 9.8 – `isalpha` / `isdigit`


---

```

1 texte = "LeGrandParc"
2 if texte.isalpha():
3     print("Contient que des lettres")
4 else:
5     print("Contient d'autres caractères")
6 # affiche Contient que des lettres

```

---

### 9.3 Exercices

- 1 Écrire un programme qui demande à l'utilisateur-trice de saisir une phrase et qui affiche la longueur de cette phrase.
- 2 Écrire un programme qui demande à l'utilisateur-trice de saisir deux mots et qui dit si les deux mots sont les mêmes.
- 3
  - a) Écrire un programme qui demande à l'utilisateur-trice de saisir une phrase et qui affiche les 10 premiers caractères de cette phrase.
  - b) Modifier le code pour demander également à l'utilisateur-trice le nombre de caractères à afficher.
- 4 Afficher le menu d'un programme jusqu'à ce que l'utilisateur-trice saisisse "q" pour quitter.
 

```

-- Menu --
1. Option 1
2. Option 2
q. Quitter
Choisissez une option :

```
- 5
  - a) Écrire un programme qui demande à l'utilisateur-trice de saisir une phrase et dit si elle contient la lettre "a".
  - b) Écrire un programme qui demande à l'utilisateur-trice de saisir une phrase et qui compte le nombre de lettres "a".
  - c) Écrire un programme qui demande à l'utilisateur-trice de saisir une phrase et qui compte le nombre de voyelles.
- 6 Écrire un programme qui demande à l'utilisateur-trice de saisir une phrase et qui affiche cette phrase sans les espaces.
- 7 Écrire un programme qui demande à l'utilisateur-trice de saisir un mot et qui affiche ce mot à l'envers.