

Manipulation des chaînes de caractères

Plan de la séance

- 1h de Cours Magistral interactif
- 1h de Travaux Pratiques (TP)
- TDs supplémentaires + Mini-projet + Évaluation (facultative)

I. Introduction aux chaînes de caractères

- Une chaîne est une **séquence de caractères** encadrée par des guillemets simples ' ou doubles ''.
- Les chaînes sont **immuables** (non modifiables directement).

```
chaine1 = "Python"  
chaine2 = 'Programmation'
```

II. Opérations de base

1. Concaténation

```
prenom = "Alice"  
salutation = "Bonjour " + prenom  
print(salutation)  # Bonjour Alice
```

2. Répétition

```
etoiles = "*" * 10  
print(etoiles)  # **********
```

3. Accès par indice

```
mot = "ordinateur"  
print(mot[0])  # o  
print(mot[-1])  # r
```

4. Slicing (tranches)

```
mot = "informatique"  
print(mot[0:5])  # infor  
print(mot[:2])  # ifratq  
print(mot[::-1])  # euqitamrofni
```

III. Méthodes utiles sur les chaînes

Méthode	Description
len(chaine)	Longueur de la chaîne
chaine.lower()	Mise en minuscules
chaine.upper()	Mise en majuscules
chaine.strip()	Suppression des espaces
chaine.replace(a,b)	Remplacement
chaine.find("mot")	Position du mot ou -1
"mot" in chaine	Présence du mot (True/False)
chaine.split()	Transformation en liste de mots

IV. Exemples illustratifs

Recherche de motif

```
texte = "Il fait beau aujourd'hui"
print("beau" in texte)      # True
print(texte.find("beau"))   # 7
```

Remplacement

```
texte = "Python est facile"
print(texte.replace("facile", "puissant")) # Python est puissant
```

Travaux Pratiques

► Exercice 1 : Compter les voyelles

Écrire un programme qui compte les voyelles d'une chaîne.

```
def compter_voyelles(chaine):
    voyelles = "aeiouyAEIOUY"
    compteur = 0
    for c in chaine:
        if c in voyelles:
            compteur += 1
    return compteur

texte = input("Entrez un texte : ")
print("Nombre de voyelles :", compter_voyelles(texte))
```

► Exercice 2 : Inverser une chaîne

```
def inverser(chaine):  
    return chaine[::-1]  
  
texte = input("Entrez une chaîne : ")  
print("Chaîne inversée :", inverser(texte))
```

► Exercice 3 : Supprimer les espaces

```
def supprimer_espaces(texte):  
    return texte.replace(" ", "")  
  
print(supprimer_espaces("du texte avec des espaces"))
```

► Exercice 4 : Compter les mots

```
def compter_mots(texte):  
    return len(texte.split())  
  
print(compter_mots("Ceci est une phrase simple"))
```

TD Supplémentaires

TD 1 : Mot le plus long

```
def mot_plus_long(phrase):  
    mots = phrase.split()  
    return max(mots, key=len)  
  
texte = input("Entrez une phrase : ")  
print("Mot le plus long :", mot_plus_long(texte))
```

TD 2 : Fréquence des lettres

```
def frequence_lettres(texte):  
    texte = texte.lower()  
    for lettre in "abcdefghijklmnopqrstuvwxyz":  
        if lettre in texte:  
            print(f"{lettre} : {texte.count(lettre)}")  
  
frequence_lettres("Hello World!")
```

Mini-projet : Analyseur de texte

Objectif : à partir d'un texte entré par l'utilisateur, le programme doit :

- Afficher le nombre de caractères
- Afficher le nombre de mots
- Afficher le mot le plus long
- Afficher le nombre de voyelles

```
def analyse_texte(texte):
    voyelles = "aeiouyAEIOUY"
    nb_voyelles = sum(1 for c in texte if c in voyelles)
    nb_caracteres = len(texte)
    mots = texte.split()
    nb_mots = len(mots)
    mot_long = max(mots, key=len) if mots else ""

    print(f"Nombre de caractères : {nb_caracteres}")
    print(f"Nombre de mots : {nb_mots}")
    print(f"Mot le plus long : {mot_long}")
    print(f"Nombre de voyelles : {nb_voyelles}")

texte = input("Entrez un texte : ")
analyse_texte(texte)
```

Mini-évaluation

QCM

1. `len("Python")` renvoie :
 - 5
 - 6
 - 7
2. `"Py" in "Python"` renvoie :
 - False
 - True
 - Erreur
3. `"abc".upper()` donne :
 - "ABC"
 - "abc"
 - "Abc"

Exercice

Écrire une fonction `retourne_chaine(s)` qui retourne la chaîne inversée sans les espaces.

```
def retourne_chaine(s):
    s = s.replace(" ", "")
    return s[::-1]
```

```
print(retourne_chaine("Bonjour tout le monde")) # ednomeltuorujnoB
```

Pour aller plus loin

- S'entraîner sur [w3schools – Strings Python](#)
- Créer un filtre basique pour détecter et masquer des mots interdits dans une phrase (modérateur de chat)