

Semaine 3

Initiation à l'algorithmique et programmation

Revekka Kyriakoglou

Plan du cours

1 String manipulation

2 Iteration

Strings

String (Chaîne de caractères) est un type d'objet.

- caractères : lettres, caractères spéciaux, space, numéros
- guillemets ou guillemets simples
- concatenation avec +

Opérations

Il y a diverses opérations sur les chaînes :

- concatenation,
- le découpage (slicing),
- la mise en forme,
- ...

Longueur (length)



Pour trouver la longueur d'une chaîne, vous pouvez utiliser la fonction `len()`.

```
texte = "Python_is_amazing"  
longueur = len(texte) # length of a string  
print(longueur) #>>> 17
```

Découpage



On peut extraire une partie d'une chaîne à l'aide du **découpage**. Le découpage utilise le format

`chaîne[début:fin]`

où début est l'indice du premier caractère souhaité et fin est l'indice du caractère juste après le dernier caractère souhaité.

```
texte = "Python_is_amazing"  
sous_chaine = texte[7:10] # Decoupage  
print(sous_chaine)  
print(len(sous_chaine))
```

Mise en forme



On peut formater des chaînes de caractères à l'aide des **f-strings** (chaînes formatées) ou de la méthode **.format()**

```
name1 = "Anna"  
age1 = 31  
message = f"Je m'appelle {name1} et j'ai {age1} ans."  
print(message)  
#>>> "Je m'appelle Anna et j'ai 31 ans."
```

```
name2 = "John"  
age2 = 25  
message = "Je suis {} et j'ai {} ans".format(name2, age2)  
print(message)  
#>>> "Je suis John et j'ai 25 ans."
```

Méthodes



Python propose de nombreuses méthodes de chaînes intégrées pour les opérations courantes sur les chaînes.

- `.upper()`
- `.lower()`
- `.strip()`

```
text = "___Programmation_Python___"  
stripped_text = texte.strip()  
uppercase_text = texte.upper()  
print(stripped_text)  
print(uppercase_text)
```


Recherche



On peut vérifier si une sous-chaîne existe dans une chaîne à l'aide de l'opérateur `in`.

```
texte = "Python_est_un_langage_puissant"  
if "Python" in texte:  
    print("Python_est_dans_le_texte.")
```



Qu'attendez-vous de ce code ?

Inverser une chaîne



On peut inverser une chaîne en Python en utilisant plusieurs méthodes :

- Utilisation de la Slicing (Découpage) avec une étape négative.
- Utilisation d'une boucle (loop).

```
chaîne = input("Entrez une chaîne : ")
chaîne_inversee = chaîne[::-1]
print("Chaîne inversée :", chaîne_inversee)
```



Qu'attendez-vous de ce code ?

Multiplication (Star operator)



Multiplication entre une chaîne de caractères et un nombre entier.
En python, cela signifie : répéter la chaîne (str) autant le nombre de fois.

```
hi = 'hello_there'  
name = 'Anna'  
greeting = hi + '_' + name  
print(greeting)  
repetition = hi + ('_' + name)*3  
print(repetition)
```

print()



Nous avons déjà vu qu'il est possible d'imprimer plusieurs choses à l'aide du symbole `+`.

Nous pouvons également utiliser la **virgule** (`,`) à la place de `+`.



Compilez le code suivant pour voir quelle est la différence (s'il y en a une) entre les deux.

```
x = 'anna'  
print(x, x, x)  
print(x + x + x)
```

Exercise

? Exercise

Créer un programme qui demande à l'utilisateur un mot (mot) et un nombre (nbr) et qui imprime ensuite le mot, nbr fois.