

TP N°2 – Input et IF

Lors du 1^{er} TP, nous avons utilisé Python en mode interactif. Ce mode est pratique pour découvrir Python mais rapidement on ressent le besoin de sauvegarder les lignes de code (les instructions) qu'on a saisies dans l'interpréteur de manière à ne pas avoir à tout retaper d'une fois sur l'autre. Sur le site repl.it, nous allons utiliser le cadre de gauche pour écrire notre programme. Pour exécuter votre programme, il vous suffira de cliquer sur "Run".

Note: à partir de maintenant vous allez écrire des programmes dans lesquels vous demanderez à l'utilisateur de rentrer des valeurs ou des chaînes de caractères. Pour cela vous allez inviter l'utilisateur à rentrer des caractères au clavier en utilisant la fonction **input**. (cf. annexe).

1. Instruction input

Dans certains cas le programme demande à l'utilisateur de répondre à des questions, à des choix ... L'utilisateur doit donc rentrer des valeurs ou des chaînes de caractères lors de l'exécution du programme. Cette interaction utilisateur/programme se fait au moyen de l'instruction **input** (voir en annexe).

Exercice1 - input

Faire un programme permettant à un utilisateur de saisir deux chaînes de caractères utilisant et affichant à l'écran :

- la première chaîne et sa longueur,
- la seconde chaîne et sa longueur,
- le nombre de fois où la seconde chaîne apparaît dans la première.

Exercice 2 - input & int()

Faire un programme permettant de calculer l'aire d'un rectangle étant données sa largeur et sa longueur, ces deux dernières valeurs étant entrées par l'utilisateur sur la ligne de commande en utilisant ici l'instruction **input**.

Exercice 3 – input et révisions sur les chaînes de caractères

Ecrire un programme permettant à un utilisateur de saisir une séquence nucléotidique et d'afficher le pourcentage en A, en T, en G et en C, ainsi que le pourcentage de caractères qui ne font pas partie de l'alphabet nucléotidique (qui correspondent soit à des erreurs de saisie, soit à des indéterminations après séquençage). On fera en sorte que le programme fonctionne, quelle que soit la casse utilisée pour saisir a, t, g, ou c (**la casse** correspond à la façon dont les mots sont écrits : majuscule ou minuscule).

2. Instruction if et blocs d'instructions

L'instruction conditionnelle est l'un des deux fondements de la programmation. En python cette instruction se note **if**.

Note : A partir de maintenant, on se servira du mode interactif exclusivement pour tester des instructions qu'on ne maîtrise pas. Comme on va dès à présent écrire des programmes de plus en plus longs et complexes, il est utile de commenter son code, ne serait-ce que pour savoir, sans lire les instructions, ce qu'il est censé faire. On utilisera pour cela le **caractère #** qui indique à l'interpréteur Python d'ignorer (de ne pas considérer) tout ce qui le suit (jusqu'au saut de ligne).

Exercice 4 - if :

Ecrire un programme permettant à un utilisateur de saisir deux chaînes caractères et de savoir si la seconde est contenue dans la première (on fera deux versions de ce programme : l'une avec **count**, l'autre avec **in**).

Exercice 5 - if : ... else:

Ecrire un programme permettant de savoir si un nombre entier saisi par l'utilisateur est pair ou impair.

Aide : % est l'opérateur modulo (le reste de la division entière).

Exercice 6 - if : ... elif : ... else:

Ecrire un programme permettant à un utilisateur de saisir un nombre et indiquant si ce nombre est positif, négatif ou égal à zéro.

Exercice 7 - Extraction d'une partie d'une chaîne

7.1. Ecrire un programme permettant à un utilisateur de saisir une chaîne ainsi que d'extraire de cette chaîne les n premiers caractères (encore une fois la valeur de n doit être saisie par l'utilisateur),

7.2. Ecrire un programme permettant à un utilisateur de saisir une chaîne ainsi que d'extraire de cette chaîne les n derniers caractères de la chaîne saisie (via la saisie de n par l'utilisateur)

7.3. En utilisant les exercices précédents écrire un programme **donnant le choix à l'utilisateur** de traiter un des deux cas précédents. On utilisera dans ce cas une instruction du type "if: ... elif: ... else:"

Annexe

La fonction **input()** provoque une interruption dans le programme courant. L'utilisateur est invité à entrer des caractères au clavier et à terminer avec <Enter>. Lorsque cette touche est enfoncée, l'exécution du programme se poursuit, et la fonction fournit en retour une chaîne de caractères correspondant à ce que l'utilisateur a entré. Cette chaîne peut alors être assignée à une variable quelconque, convertie, etc.

On peut invoquer la fonction **input()** en laissant les parenthèses vides. On peut aussi y placer en argument un message explicatif destiné à l'utilisateur. Exemple :

```
prenom = input("Entrez votre prénom : ")
print("Bonjour,", prenom)
```

La fonction **input()** renvoie toujours une chaîne de caractères. Si vous souhaitez que l'utilisateur entre une valeur numérique, vous devrez donc convertir la valeur entrée (qui sera donc de toute façon de type string) en une valeur numérique du type qui vous convient, par l'intermédiaire des fonctions intégrées **int()** (si vous attendez un entier) ou **float()** (si vous attendez un réel). Exemple :

```
print("Entrez un nombre entier : ")
ch = input()
n = int(ch)           # conversion de la chaîne en un nombre entier
```

```
>>> a = input("Entrez une valeur numérique : ")
Entrez une donnée numérique : 40.25
>>> type(a)
<class 'str'>
>>> b = float(a)      # conversion de la chaîne en un nombre réel
>>> type(b)
<class 'float'>
```