

**DEV1 – DEV1L – Laboratoires Python****TD 04 – Alternatives**

Le but de ce TD est d'apprendre à écrire une instruction if de façon à n'exécuter un bloc de code que si une certaine condition est satisfaite.

**Table des matières**

<b>1</b>	<b>L'instruction if</b>	<b>2</b>
<b>2</b>	<b>Expressions booléennes</b>	<b>3</b>
<b>3</b>	<b>Exercices</b>	<b>4</b>
<b>4</b>	<b>Exercices plus avancés</b>	<b>4</b>
<b>5</b>	<b>En résumé ...</b>	<b>5</b>

## Rappel

Vous devez absolument avoir fait et compris les td précédents *avant* d'aborder celui-ci.

# 1 L'instruction if

Le morceau de code suivant illustre l'utilisation de l'instruction `if`, il affichera « Ce nombre est positif » si `x` est plus grand que 0, « Ce nombre est négatif » si `x` est plus petit que 0 et « Ce nombre est nul » si `x` vaut 0.

```
1 if x > 0:
2     print("Ce nombre est positif")
3 elif x < 0:
4     print("Ce nombre est négatif")
5 else:
6     print("Ce nombre est nul")
```

ifelif.py

En Python, le symbole `:` annonce le début d'un nouveau bloc de code

Un tel bloc de code doit *obligatoirement être indenté* : les lignes de code qui le composent doivent avoir au moins un espace au début. Les bonnes pratiques de programmation Python recommandent d'utiliser *exactement 4 espaces* (ni plus, ni moins).

Le code suivant sera par exemple refusé par Python :

```
1 x = int(input("Entrez un nombre: "))
2 if x >= 0:
3     print("Ce nombre est positif")
```

nonindente.py

## Exercice 1 L'âge - if

Écrivez un script Python qui demande à l'utilisateur son âge et affiche s'il est majeur (s'il a plus de 18 ans). S'il n'est pas majeur le programme n'affiche rien.

Exemple : Si l'utilisateur entre 19 le programme affiche « vous êtes majeur ».

## Exercice 2 Le maximum

Écrivez un script qui demande deux nombres à l'utilisateur et qui affiche le maximum à l'écran. Attention : on vous demande d'y arriver sans utiliser la fonction `max()` de Python.

## Exercice 3 Paire ou impair - if/else

Écrivez un script Python qui demande à l'utilisateur un nombre entier et affiche « ce nombre est pair » ou « ce nombre est impair » selon le cas.

Exemple : Si l'utilisateur entre -23 le programme affiche « ce nombre est impair ».

## Exercice 4 Les espaces - if/elif/else

Écrivez un script Python qui demande à l'utilisateur d'entrer du texte. Le script affiche ensuite à l'écran « Votre texte ne contient pas d'espace », « Votre texte contient un unique espace » ou « Votre texte contient plus d'un espace » selon les cas.

Exemple : Si l'utilisateur entre « Bonjour Charlot » le programme affiche « Votre texte contient un unique espace ».

Pour rappel, on a vu dans le TD01 comment compter le nombre de fois qu'un caractère apparaît dans une chaîne.

## 2 Expressions booléennes

Pour construire une condition (c'est-à-dire une expression dont la valeur s'évalue à `True` ou `False`), on utilisera les opérateurs de comparaison de nombres :

Signification	Symbole	Exemple
plus petit	<code>&lt;</code>	<code>age &lt; 18</code>
plus petit ou égal	<code>&lt;=</code>	<code>age &lt;= 10</code>
plus grand	<code>&gt;</code>	<code>age &gt; 18</code>
plus grand ou égal	<code>&gt;=</code>	<code>age &gt;= 21</code>
égal	<code>==</code>	<code>i == 4</code>
différent	<code>!=</code>	<code>nb != 42</code>

Et on combinera des conditions avec les *opérateurs booléens* :

- ▷ Le *ET* logique s'écrit `and`
- ▷ Le *OU* logique s'écrit `or`
- ▷ La négation s'écrit `not`.

Les opérateurs `in` et `not in` peuvent également être utilisés pour tester qu'une chaîne de caractère en contient une autre ou qu'une liste contient un objet donné.

### Exercice 5 Conditions

Lancez Python en mode interactif et affichez les valeurs des expressions suivantes. Pour chacune d'entre-elles, donner sa valeur et sa sémantique.

Expression	Valeur de l'expression	Sémantique de l'expression
<code>10 &lt; 20</code>		
<code>10 &gt; 20</code>		
<code>1 == 2</code>		
<code>21/2 != 10</code>		
<code>21//2 != 10</code>		
<code>'ABC' == 'ABC'</code>		
<code>'abc' == 'ABC'</code>		
<code>'1' == 1</code>		
<code>[1,2,3] == [1,2,3]</code>		
<code>[1,2,3] == [1,3,2]</code>		
<code>1 in [7,8,9,0]</code>		
<code>'A' not in ['ABC','def','G']</code>		
<code>'ABC' not in 'ABCDEFGH'</code>		
<code>'c' in 'ABCDEFGH'</code>		

Une condition peut parfaitement faire intervenir des variables ou des appels de fonctions, comme nous pouvons le voir dans l'exercice suivant :

### Exercice 6 Conditions (bis)

Lancez Python en mode interactif. Déclarez 4 variables `a`, `b`, `c` et `d` initialisées avec les valeurs `423`, `4`, `'salut'` et `[10,3,8,-1]` respectivement. Affichez les valeurs des expressions suivantes. Qu'affiche Python pour chacune d'entre-elles ?

Expression	Valeur de l'expression
<code>a%3 == 0</code>	
<code>a%2 == 0 and b &gt; 0</code>	
<code>a%b == 0</code>	
<code>a &lt; b</code>	
<code>a%b == 0 or b%a == 0</code>	
<code>max(d) != 30</code>	
<code>len(c) == 5</code>	
<code>c[0] == 's'</code>	
<code>c[1] == 'z'</code>	
<code>1 in d</code>	
<code>-b/4 in d</code>	
<code>28 not in d</code>	

### 3 Exercices

#### Exercice 7 Les voyelles

Écrire un script Python qui demande à l'utilisateur un mot et affiche si la première lettre est une voyelle ou non.

#### Exercice 8 Première == dernière ?

Écrire un script Python qui demande à l'utilisateur un mot et affiche si la première lettre est la même que la dernière ou non. Par exemple, si l'utilisateur entre « python », le programme affiche que la première et la dernière lettre ne sont pas égales.

(Aide : Le premier caractère de la chaîne `s` est `s[0]`, le dernier est `s[len(s)-1]`)

#### Exercice 9 Le type de triangle

Écrivez un script Python qui demande à l'utilisateur la longueur des 3 côtés d'un triangle et affiche s'il est : équilatéral (tous égaux), isocèle (2 égaux) ou quelconque.

Exemple : si l'utilisateur entre 2.5, 5 et 5 le programme affiche « le triangle est isocèle ».

### 4 Exercices plus avancés

Dans les exercices suivants, très peu d'aide vous est fournie (délibérément). Il va probablement que vous cherchiez un peu sur le web si vous ne savez pas comment faire. Cela fait partie du travail d'informaticien de chercher :)

#### Exercice 10 Égalité de fractions

Écrivez un script qui demande à l'utilisateur 4 nombres `a`, `b`, `c`, et `d`, et affiche `True` si les fractions  $\frac{a}{b}$  et  $\frac{c}{d}$  sont égales. Si  $b = 0$  ou  $d = 0$ , le script affiche un message d'erreur.

#### Exercice 11 Droites

Écrivez un script qui demande à l'utilisateur 3 nombres `a`, `b` et `c`, et affiche les coordonnées des points d'intersection de la droite d'équation  $ax + by + c = 0$  avec :

- ▷ l'axe X (si l'intersection n'existe pas, le script affiche «pas d'intersection avec l'axe X»),

- ▷ l'axe Y (si l'intersection n'existe pas, le script affiche «pas d'intersection avec l'axe Y»).

Si  $a = 0$  et  $b = 0$ , le script affiche un message d'erreur (vu que l'équation ne définit pas une droite dans ce cas).

#### Exercice 12 Racines de polynômes du second degré

Écrivez un script qui demande à l'utilisateur 3 nombres  $a$ ,  $b$  et  $c$ , et affiche

- ▷ les racines réelles du polynôme  $ax^2 + bx + c$  si elles existent et qu'il y en a plusieurs,
- ▷ la racine réelle du polynôme  $ax^2 + bx + c$  si elle existe et est unique,
- ▷ « Pas de racines réelles » si elles n'existent pas.

Attention, le coefficient  $a$  peut être nul.

#### Exercice 13 Dans un cercle

Écrivez un script qui demande à l'utilisateur 5 nombres  $x$ ,  $y$ ,  $xc$ ,  $yc$  et  $r$ , et affiche `True` si le point  $(x, y)$  se trouve à l'intérieur du cercle de centre  $(xc, yc)$  et de rayon  $r$ , `False` sinon.

#### Exercice 14 Triangle possible

Écrivez un script qui demande à l'utilisateur 3 nombres  $a$ ,  $b$  et  $c$ , et affiche `True` si on peut créer un triangle dont les côtés mesurent  $a$ ,  $b$  et  $c$ , `False` sinon.

## 5 En résumé ...

### Principaux points de matière du TD

Voici les principaux points abordés lors de ce TD. Vous devez absolument être à l'aise avec ceux-ci avant d'aborder la prochaine séance d'exercice.

1. Comprendre et utiliser l'instruction `if` de Python.
2. Ecrire et utiliser des expressions booléennes.
3. Mettre l'instruction `if` et les expressions booléennes à profit pour écrire des scripts Python résolvant un problème donné.