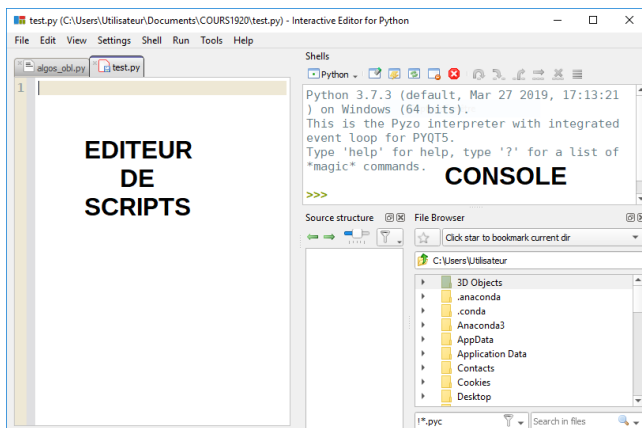


## Premiers pas en Python

Pour écrire et exécuter du code python, nous allons utiliser l'éditeur Pyzo.



- L'éditeur de scripts permet d'écrire les fonctions.
- La console permet de faire des calculs et d'appeler les fonctions.

### Attention!

Pour que les fonctions écrites dans l'éditeur de scripts puissent être appelées dans la console, il faut presser la touche 'F5' du clavier.

Pour tous les exercices, on pourra utiliser le tableau de "conversion" suivant:

Français	Python
$a$ est-il égal à 5 ?	<code>a == 5</code>
$a$ est-il différent de 5 ?	<code>a != 5</code>
si ... alors : ...	<code>if ... :</code> ...
si ... alors : ... Sinon : ...	<code>if ... :</code> ... <code>else :</code> ...
tant que ... faire : ...	<code>while ... :</code> ...
pour $i$ allant de 1 à 5 faire : ...	<code>for i in range(1, 6) :</code> ...
définir $f(x)$ : ... retourner ...	<code>def f(x) :</code> ... <code>return ...</code>

**Exercice 1** Une usine propose des tarifs dégressifs lorsqu'on commande un grand nombre de T-Shirt identiques. - Si entre 1 et 20 T-Shirts sont commandés, chaque T-Shirt coûte 10 euros. - Si plus de 20 T-Shirts sont commandés, chaque T-Shirt coûte 8 euros.

1. (a) Quel est le prix total de 6 T-Shirts? (b) Quel est le prix total de 25 T-Shirts?
2. Compléter la fonction `total` ci-dessous. Elle prend en entrée le nombre  $n$  de T-Shirts commandés et retourne le prix total à payer.

```
def total(n):
    if ...:
        prix = 10 * n
    else:
        prix = ...
    return prix
```

3. Programmer et exécuter la fonction `total`. Vérifier les résultats de la question 1.

**Exercice 2** Un magasin propose un tarif dégressif pour les photocopies.

Les 20 premières photocopies sont facturée 10 centimes chacune puis les suivantes sont facturées 8 centimes chacune.

1. Calculer le prix de 12 photocopies puis de 38 photocopies.
2. Ecrire une fonction `pr i x` qui calcule et renvoie le prix pour  $n$  photocopies ( $n$  est un paramètre)
3. En déduire le prix de 150 photocopies.

**Exercice 3** Un cinéma propose les tarifs suivants:

- Le tarif A: chaque séance de cinéma coûte 7€
- Le tarif B: avec un abonnement de 35€ (pour 1 an), chaque séance coûte 5€

1. (a) Quel est le prix de 6 séances au tarif A? (b) Quel est le prix de 9 séances au tarif B?
2. Compléter la fonction python `tar i fA` ci-contre que retourne le prix de  $n$  séances au tarif A:

```
def tarifA(n):
    prix = ...
    return ...
```

3. Ecrire de même une fonction python `tar i fB` que retourne le prix de  $n$  séances au tarif B.
4. Si on compte aller 25 fois au cinéma dans l'année, l'abonnement est-il économique?
5. On souhaite déterminer le nombre de séances à partir duquel l'abonnement est avantageux.

(a) Compléter la fonction `nbM i n` permettant de répondre au problème posé:

```
def nbMin():
    nb = 1
    while ... :
        nb = ...
    return nb
```

(b) Programmer cette fonction et répondre au problème posé.

**Exercice 4** On souhaite écrire une fonction `somme` qui prend en paramètre un entier naturel  $n$  et retourne la valeur de la somme  $1 + 2 + 3 + \dots + n$ .

1. Justifier que `somme(5)` renvoie 15.
2. Compléter la fonction `somme` suivante:

```
def somme(n):
    s = 0
    for i in range(... , ...):
```

```
s = ...
return s
```

3. Utiliser des appels à la fonction `somme` pour calculer les sommes suivantes :

- (a) Somme des entiers entre 1 et 50 (inclus).                      (b) Somme des entiers entre 20 et 100 (inclus).

**Exercice 5** Ecrire une fonction `produit` qui prend en paramètre deux entiers  $a$  et  $b$  et qui retourne le produit des entiers compris entre  $a$  (inclus) et  $b$  (inclus).

Par exemple `produit(3, 5)` retourne  $3 \times 4 \times 5 = 60$ .

**Exercice 6** En Python, on importe la fonction racine carrée (`sqrt`) avec la syntaxe `from math import sqrt` en début de script. De même, on peut importer la fonction cosinus (`cos`) ou d'autres fonctions.

1. On considère un triangle rectangle tel que les côtés de l'angle droit mesurent respectivement  $a$  cm et  $b$  cm (avec  $a > 0$  et  $b > 0$ ). Exprimer la longueur de l'hypoténuse en fonction de  $a$  et  $b$ .
2. En déduire une fonction `pyth` qui prend en paramètre les longueurs  $a$  et  $b$  et renvoie une valeur approchée de la longueur de l'hypoténuse de ce triangle.
3. Ecrire une fonction `dist(xA, yA, xB, yB)` qui calcule et retourne la distance entre deux points  $A(x_A; y_A)$  et  $B(x_B; y_B)$  d'un repère orthonormé, c'est-à-dire la longueur du segment  $[AB]$ .

**Exercice 7** L'indice de masse corporelle (IMC) d'une personne est donné par  $IMC = \frac{\text{masse}}{\text{taille}^2}$  où la masse est en kg et la taille en m.

1. On va écrire une fonction `imc` qui calcule l'IMC
  - (a) Quels sont les paramètres de cette fonction?
  - (b) Ecrire cette fonction `imc` en python.
  - (c) Que vaut l'IMC d'une personne de 60kg et 1,60m?
2. Un IMC inférieur à 18 correspond à la maigreur. Un IMC compris entre 18 et 30 correspond à une corpulence normale. Un IMC supérieur à 30 correspond au surpoids.
  - (a) Ecrire une fonction `resultat` qui calcule l'IMC d'une personne correspondant aux données fournies et renvoie "maigreur", "normal" ou "surpoids" selon le cas. Pour cela, faire appel à la fonction `imc` écrite à la question précédente.
  - (b) Tester la fonction `resultat` avec une personne de 70kg et 1,75m.

## Application: nombres, divisibilité, nombres premiers

### Exercice 8

1. Dans la console, entrer `1/3` et valider.  
Qu'observe-t-on? Python travaille-t-il avec des nombres rationnels?
2. Dans la console, entrer `0.1+0.1+0.1` et valider.  
Qu'observe-t-on? Peut-on effectuer des comparaisons de nombres décimaux de manière sûre en Python?

**Exercice 9**

1. Dans la console, entrer  $14 // 3$  Quel est le résultat?  
Quel est le rôle de l'opérateur  $//$ ? (faire éventuellement d'autres tests avec des entiers positifs).
2. Dans la console, entrer  $14 \% 3$  Quel est le résultat?  
Quel est le rôle de l'opérateur  $\%$ ? (faire éventuellement d'autres tests avec des entiers positifs).
3. Un restaurant livre ses sushi dans des boîtes de 12. Ecrire une fonction `nb_boîtes` qui prend en entrée un nombre  $n$  de sushi à livrer à un client et retourne le nombre boîtes nécessaires.

**Exercice 10**

La fonction ci-dessous prend en entrée un entier naturel  $n$  supérieur ou égal à 2.

```
def premier(n):
    i = 2
    while i < n:
        if n%i == 0:
            return False
        i = i+1
    return True
```

1. (a) Quel est le rôle de cette fonction?  
(b) Programmer cette fonction et l'exécuter pour  $n=23$ ,  $n=504$ .
2. Le mathématicien suisse Leonhard EULER (1707 – 1783) a proposé la méthode suivante pour obtenir des nombres premiers : choisir un entier naturel  $n$  puis calculer  $n^2 - n + 41$ .  
(a) Ecrire et programmer une fonction `euler` prenant en entrée un entier naturel  $n$  et retournant  $n^2 - n + 41$ .  
(b) Qu'obtient-on avec la fonction d'Euler pour  $n = 2$ ? Est-ce un nombre premier?  
(c) Ecrire une fonction `test_euler` qui teste si les valeurs `euler(i)` pour  $i$  compris entre 0 et 100 sont toutes des nombres premiers. Si ce n'est pas le cas, renvoyer la première valeur de  $i$  telle que `euler(i)` ne soit pas un nombre premier.

Aide: l'opérateur `not` permet de tester la négation d'une condition.

3. On souhaite améliorer la fonction `premier` ci-dessous afin qu'elle effectue moins d'opérations. Pour cela:
  - On ne teste la divisibilité de  $n$  que par 2 et les nombres impairs
  - On s'arrête dès que  $i$  a dépassé le premier entier supérieur à  $\sqrt{n}$
 (a) Ecrire une fonction `premier_opt` sur le modèle de `premier` et tenant compte de ces modifications.  
Aide: on peut obtenir l'entier inférieur à un nombre positif  $k$  avec l'instruction `int(k)`.  
(b) Comparer le temps de calcul avec de grands nombres premiers comme 5355443 ou 42302243.  
(c) Justifier mathématiquement ces améliorations.