

# Python : Types et opérations

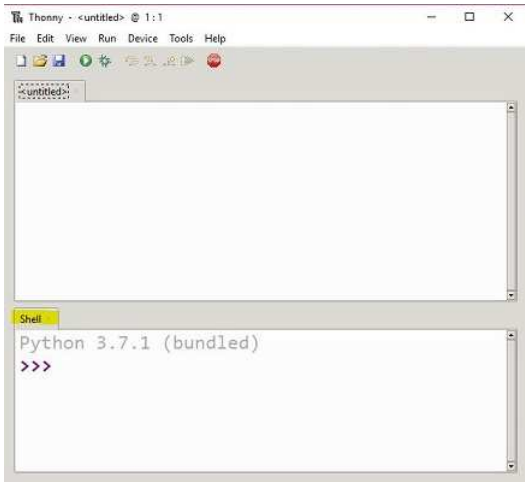
Les documents suivants sont construits de telle manière à ce que vous observiez des exemples simples commentés puis que vous réalisiez les exercices proposés au fur et à mesure.

Il est important de tester vous même les petits morceaux de code proposés dans les exemples avant de passer aux exercices

## I Thonny

Dans cette introduction, vous utiliserez Python en mode interactif via la zone **Shell** du logiciel [Thonny](#).

Cette zone permet de tester du code sans avoir à la sauvegarder au préalable.



Une invite de commande aussi appelée **prompt** et symbolisée par 3 chevrons '>>>' vous permet de tester directement du code qui sera immédiatement interprété après l'appui sur la touche `entrée`

## II Les types de base

L'instruction **type** permet d'obtenir le type d'un objet.  
Dans le shell, tester les types des différents objets suivants :

type(8)  
type(-5)  
type(8.0)  
type(-5.4)

type('Bonjour !')  
type(True)

Quelles sont les différentes valeurs possibles pour un booléen ? (voir ce [lien](#))

Compléter le tableau suivant avec les exemples précédemment faits :

Exemple ci-dessus	Type	Nom
		entiers (integer)
		flottant
		chaîne de caractères (string)
		booléen

Le type détermine les propriétés de la valeur (par exemple les opérations qu'elle peut subir) et matérielles (par exemple, la façon dont elle est représentée en mémoire et la place qu'elle occupe)

## III Les nombres entiers

Effectuer, via le shell, les calculs suivants et compléter au fur et à mesure le tableau ci-dessous :

60 - 7	10 % 3
2 * 3 + 4	11 % 3
(-2 + 3) * 7 - 1	12 % 3
6 / 3	13 % 3
1 / 3	14 % 3
2 ** 2	2 < 7
2 ** 3	2 < 2
2 ** 4	5 >= 5
10 // 3	2 ** 3 ** 2
11 // 3	3 / 1 / 3
12 // 3	
13 // 3	
14 // 3	

Opérateur	Résultat obtenu	Type de l'objet obtenu
a ** b		
a // b		
a % b		
a < b		

Peut-on appliquer tous les opérateurs à deux chaînes de caractères ?

## VI Changer le type d'un objet

Dans certains cas, il est nécessaire de changer le type d'un objet.

### *Convertir en un entier :*

Testez :

```
int('457') :  
int(3.145)  
int(-2.25)  
int('4.25')  
int('a')
```

### *Convertir en un flottant :*

```
float('135')  
float('13.25')  
float(45)  
float('a')
```

### *Convertir en une chaîne de caractères :*

```
string(3.1456)  
string(12)
```