

1 Compléments mineurs mais utiles sur le type `str`

La « fonction » `input` prend en entrée une chaîne de caractères et fournit en sortie une chaîne de caractère. Elle écrit à l'écran la chaîne qu'on lui donne en argument et attend que l'utilisateur tape une ligne (terminée par un retour chariot); le résultat de la « fonction » est alors la chaîne de caractère qu'a tapée l'utilisateur et est donc toujours de type `str`.

```
>>> def bonjour () :  
...     prenom = input("Quel est votre prénom ? : ")  
...     nom = input("Quel est votre nom de famille ? : ")  
...     print("Bonjour %s %s ! bonne journée." % (prenom,nom))  
...  
>>> bonjour()  
Quel est votre prénom ? : Albert  
Quel est votre nom de famille ? : Durand  
Bonjour Albert Durand ! bonne journée.
```

Il est cependant souvent utile de demander à l'utilisateur une valeur d'un autre type que `str`. Il faut alors gérer « à la main » la conversion de `str` vers le type que l'on souhaite.

Rappel : la fonction `int` convertit en entier (si possible) l'argument qu'on lui donne. Ainsi les nombre flottants (type `float`) sont tronqués par la fonction `int`. Cette fonction `int` marche aussi pour les chaînes de caractères *ne contenant que des chiffres* (au passage, `str` fait donc l'inverse).

```
>>> int(12.7)  
12  
>>> int("12")  
12  
>>> int("12.7")  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
ValueError: invalid literal for int() with base 10: '12.7'  
>>> str(126)  
'126'
```

Si l'on veut ignorer les caractères qui ne sont pas des chiffres, il faut le programmer soi-même :

```
>>> def intsouple (s) :  
...     r = 0  
...     for c in s :  
...         if c >= '0' and c <= '9' :  
...             r = 10 * r + int(c)  
...         #sinon on ignore le caractère  
...     return r  
...  
>>> intsouple ("to3to4glop0")  
340
```

et ainsi par exemple :

```
>>> def askint () :  
...     s = input("entrez un entier positif : ")  
...     n = intsouple(s)  
...     if str(n) != s :  
...         print("Vous tapez à coté des touches !")  
...     return n  
...  
...
```

```
>>> 2 * askint()
entrez un entier positif : 45
90
>>> 2 * askint()
entrez un entier positif : glop9u0
Vous tapez à coté des touches !
180
```

2 Divers exercices d'application

...