

Les variables

I C'est quoi une variable ?

Une variable est une zone de la mémoire RAM formée d'un certain nombre d'octets et dans laquelle est codée en binaire une information telle qu'un nombre ou un morceau de texte. Il existe 3 types de variables :

- Les variables de type **string** (ou chaîne de caractères comme : Mon chat est noir)
- Les variables de type **integer** (ou entier relatif comme 2021)
- Les variables de type **Float** (ou nombre réel à virgule flottante, autrement dit un décimal comme 3,14)

II Définir et nommer une variable

Une variable porte un nom choisi par l'utilisateur pour être évocateur. Ainsi, pour définir une variable contenant son âge, l'utilisateur peut entrer l'instruction :

Mon_age=18

Dans ce cas, une zone comprenant un certain nombre d'octets (par exemple 4 octets, soit 32 bits) est réservée dans la mémoire RAM et une adresse pour y accéder est créée. Le nom de la variable Mon_age est donc simplement le nom de cette zone, comme une maison peut avoir un nom (sam'suffy par exemple). A noter que la variable créée est de type **integer**.

Dès que l'instruction est exécutée, le nombre entier 18 est converti en binaire ($18 = 1 \times 16 + 0 \times 8 + 0 \times 4 + 1 \times 2 + 0 \times 1$ s'écrit alors en base 2 : 1 0 0 1 0) et la zone de 4 octet reçoit l'enregistrement suivant, dans lequel nous avons séparé les 4 octets :

00000000	00000000	00000000	00010010
----------	----------	----------	----------

Attention !

Si le programmeur ne souhaite pas faire de calcul avec son âge, il n'a pas intérêt à le déclarer dans une variable de type **integer** mais plutôt une variable de type **string** qui, va prendre bien moins de place en mémoire, comme nous le montrons ici, avec l'instruction :

Mon_age="18"

C'est la présence des guillemets qui définit la variable comme un type **string**. Dans ce cas cette variable se retrouve en mémoire sous forme d'une chaîne de 2 caractères , le 1 et le 8, ce qui lui fait occuper seulement 2 octets (à comparer aux 4 du type entier)

On veillera donc à n'employer des variables numériques que si on fait des calculs avec, sinon, il vaut mieux les déclarer en type **string**.

III Afficher le contenu d'une variable à l'écran :

Si on s'est contenté de l'instruction précédente, son exécution avec le bouton RUN dans l'éditeur en ligne de Python, ne fait rien apparaître à l'écran. En effet l'instruction n' a fait qu'écrire dans la mémoire de l'ordinateur. Pour le rendre visible à l'exécution, il faut ajouter la commande **print** qui affiche le contenu de la variable à l'écran et le programme devient :

```
Mon_age="18"
```

```
print(Mon_age)
```