

Les identifiants comme les noms de variable sont un très bon support pour concrétiser un programme : ils permettent de placer directement dans les instructions des mots au choix du programmeur, avec lesquels il peut décrire ce qu'il souhaite faire. Nous présentons ci-dessous des techniques à prendre en main, qui seront très appréciées par les membres du forum, et qui vous aideront vous-même, à relire, analyser et corriger vos programmes.

Choisir un bon nom

Lorsqu'on a besoin d'une nouvelle variable, on cherchera toujours à **trouver un nom pour cette variable qui décrive le mieux possible ce qu'elle représente**. Il ne faut pas hésiter à mettre à la suite plusieurs mots pour construire un nom précis. Nous vous conseillons de nommer vos variables par une suite de mots ou abréviations accolés les uns aux autres, en mettant en majuscule la première lettre de chaque mot sauf le premier. Voici quelques exemples :

```
longueurFeuille  
nbPiecesJaunes  
maxHauteurPiquets
```

Ce style de nommage est très répandu : on l'appelle couramment **CamelCase**, ou « casse ChatMot » en français, en référence aux bosses du chameau.

On évitera en particulier l'utilisation de noms réduits à une simple lettre, comme *i* ou *a*. Les lignes suivantes montrent un exemple de très mauvais code :

```
int a = 1000;  
int b = 50;  
int c = a + b;
```

En effet, on comprend beaucoup mieux ce qui se passe lorsque des noms plus parlants sont employés :

```
int prixFour = 1000;  
int prixBatteur = 50;  
int prixTotal = prixFour + prixBatteur;
```

Le nom doit placer la variable dans son contexte afin d'être efficace. Ci-dessus, lors d'achats culinaires, nous calculons le coût total à partir du prix de nos deux produits.

Nommer une valeur

Nous avons vu au départ qu'une variable permettait d'éviter de recopier une valeur. On peut également créer une variable, simplement dans le but de décrire une valeur ; c'est notamment pratique pour diviser un calcul compliqué en plusieurs parties identifiables.

Imaginons par exemple un marchand de fruits et légumes, qui vend les produits suivants :

Nom	Quantité (kg)	Prix (centimes/kg)
Pommes	32	90
Pêches	12	96
Aubergines	23	120
Carottes	17	102
Patates	22	85

On souhaite calculer la valeur totale de ses produits. On pourrait écrire le programme comme ceci :

```
Afficher 32 * 90 + 12 * 96 + 23 * 120 + 17 * 102 + 22 * 85
```

Toutefois, si le marchand décide de changer un prix ou nous apprend qu'il s'est trompé dans une quantité, il n'est pas forcément facile de déterminer où effectuer la correction. Mais si on introduit des variables :

```
prixPommes <- 32 * 90  
prixPêches <- 12 * 96  
prixAubergines <- 23 * 120  
prixCarottes <- 17 * 102  
prixPatates <- 22 * 85  
Afficher prixPommes + prixPêches + prixAubergines + prixCarottes + prixPatates
```

il devient bien plus aisé de se repérer dans le programme.