

Vous avez remarqué qu'en C++, pour utiliser une nouvelle variable, la première fois qu'on lui affecte une valeur, il est nécessaire d'indiquer son type. Par exemple :

```
int reponse = 42;
```

Il est en fait aussi possible de séparer les deux : déclarer la variable sans valeur initiale, puis lui affecter une valeur dans la suite du programme.

```
int reponse;  
reponse = 42;
```

Nous vous recommandons cependant, dans la mesure du possible, de déclarer une variable en même temps que vous lui donnez sa première valeur, car cela organise mieux le programme : il est alors plus facile à comprendre et moins propice aux erreurs.

Remarquez par ailleurs qu'avant d'avoir affecté quelque chose à une variable, on ne sait pas ce qui se trouve dedans ! Il s'agit de la valeur qui se trouvait précédemment à l'emplacement mémoire dans lequel se trouve la variable. Ainsi, il est généralement illogique de consulter la valeur d'une variable avant de l'avoir initialisée, car cela ne va pas aider notre programme à bien se dérouler. Par conséquent, en C++, si on utilise une variable qui n'a pas été initialisée, on obtient un avertissement pour nous prévenir que l'on s'est sans doute trompé :

```
int reponse;  
cout << reponse << endl;
```

↳ *warning: 'reponse' is used uninitialized in this function*