

Bonnes pratiques

Le langage Python a le défaut d'être peu bavard sur l'origine des erreurs. Le même message d'erreur peut apparaître pour des erreurs très différentes (cf `Python_Erreurs.pdf`).

Un conseil: Faites-vous un catalogue des erreurs que vous rencontrez. Cela facilitera le débogage des programmes.

1- Ne pas utiliser un mot réservé Python pour nom de variable (cf `Python_Mots_Reserves.pdf`)

2- Ne pas utiliser un nom de fonction pour nom de variable.

3- Ne jamais utiliser le même nom de variable pour des types différents. C'est une source d'erreur.

4- Attention aux erreurs typographiques. Eviter lors de l'écriture d'un programme d'utiliser dans un premier temps des variables explicites. Comme il n'y a pas de déclaration de variables en python, une erreur typographique définit une nouvelle variable. Il y a peu de chance qu'à l'exécution celle-ci soit détectée. Il est préférable d'utiliser un nom court suivi d'un underscore pour la mise au point du programme ou de la fonction. Ensuite il faut utiliser le FindAndReplace de l'IDE pour remplacer le nom par un nom explicite toujours suivi d'un underscore.

5- Bien vérifier que deux fonctions n'ont pas le même nom. Python ne le signale pas. Pour l'exécution, il prend systématiquement la dernière fonction.

6- Dans un programme Python, les fonctions doivent être écrites avant leur appel.

7- Eviter les variables globales.

8- Utiliser un maximum de fonctions. Cela facilitera le repérage des erreurs.

9- Quand des bibliothèques contiennent des fonctions aux noms identiques, il est préférable d'utiliser ***import*** nom_bibliothèque ***as*** nom_variable au lieu de ***from*** nom_bibliothèque ***import*** *. Cela évitera les conflits entre les bibliothèques.