

Nom :	Prénom :	Classe :
-------	----------	----------

NSI 1re — Portée d'une variable : variable locale, variable globale

Objectifs :

- Savoir identifier une variable locale d'une variable globale.
- Connaître les limites d'une variable locale.

Introduction

En programmation, une **variable locale** n'est accessible que **dans la fonction ou le bloc d'instruction où elle est définie**, tandis qu'une **variable globale** est accessible **dans tout le programme**.

Variable locale

En Python, une variable déclarée **dans** une fonction est dite **locale** : elle existe dans cette fonction, et on ne peut pas l'utiliser en dehors.

```

1  # On définit une nouvelle fonction nommée "f"
2  # contenant une variable nommée "a", dont la valeur est quelconque
3  def f():
4      a = 8
5
6  # On exécute la fonction
7  f()
8
9  # On tente d'accéder à la variable "a"
10 # par exemple en affichant son contenu
11 print(a)
12
13 # Python déclenche une erreur :
14 # NameError: name 'a' is not defined

```

► Exercice 1 — À vous de jouer !

a) Dire de la variable a si elle est une variable locale ou globale, à travers les exemples ci-dessous :

<pre> a = 113 def tva(p): return p + p * 20 / 100 a = tva(a) </pre>	<pre> def soustraire(n1, n2): a = n1 - n2 return a print(soustraire(14, 67)) </pre>	<pre> c = 45 - 1789 a = c + 105 def f(r): return 3.14 * r ** 2 a = f(a) </pre>
Ici <input type="checkbox"/> a est une variable	Ici <input type="checkbox"/> a est une variable	Ici <input type="checkbox"/> a est une variable

b) Est-il possible d'accéder à une variable déclarée dans le bloc d'instruction d'une fonction ?

Variable globale

Une variable déclarée **en dehors** d'une fonction est dite **globale** et peut être lue partout (donc également depuis une fonction).

```
1 # On déclare une variable nommée "a", dont la valeur est quelconque.
2 # Cette variable est déclarée en dehors d'une fonction.
3 # On dit alors d'elle qu'elle a une portée « globale ».
4 a = 10
5
6 # On définit une nouvelle fonction nommée "f"
7 # qui tente d'accéder à la variable "a"
8 # par exemple en affichant son contenu
9 def f():
10     print(a)
11
12 # On exécute la fonction
13 f() # ... Ce qui affiche la valeur de "a" (10)
```

► Exercice 2

Depuis une fonction, est-il possible d'accéder à une variable globale ?

Une variable globale ne peut pas être modifiée depuis une fonction

En Python, une variable globale (déclarée en dehors d'une fonction) peut être **lue** partout. Mais lorsqu'on en **modifie** la valeur depuis une fonction, cette valeur ne sera changée que **dans** la fonction.

```
1 # On déclare une variable globale nommée "a"
2 a = 5
3
4 # On définit une fonction "f" qui donne une nouvelle
5 # valeur à une variable "a"
6 def f():
7     a = 10
8     print(a)
9
10 # On affiche la valeur de "a"
11 print(a) # Affiche 5 😊
12
13 # On exécute la fonction
14 f() # Affiche 10 😊
15
16 # On affiche la valeur de "a"
17 print(a) # Affiche 5 😞
```



Pour résumer, on pourrait dire que toutes les modifications effectuées sur une variable dans une fonction restent cloisonnées (enfermées) dans la fonction.

► Exercice 3

Comment faire pour modifier une variable globale, à partir de la valeur de sortie d'une fonction ?

Une variable déclarée dans une condition ou une boucle est-elle locale ou globale ?

Ce n'est pas le cas dans tous les langages de programmation mais, en Python, une variable définie dans une condition ou une boucle est globale : elle reste également définie après le bloc d'instruction de la condition ou de la boucle.

Exemples :

```
1  if True:
2      a = 6
3
4  print(a) # Affiche 6
```

```
1  for _ in range(1):
2      a = 12
3
4  print(a) # Affiche 12
```

```
1  z = 0
2  while z < 1:
3      a = 9
4      z += 1
5
6  print(a) # Affiche 9
```

► Exercice 4

Une variable déclarée dans une condition ou une boucle est-elle globale ou locale ?

Hors programme



Ce chapitre tente d'apporter quelques informations complémentaires qui ne sont pas dans le programme de NSI ... Et qui peuvent également induire en erreur ! À lire avec précaution.

Nous avons dit qu'une variable globale ne pouvait pas être modifiée depuis une fonction...

Mais il est possible de rendre cette affirmation fausse, en utilisant le mot clé `global` :

```
1 a = 5
2
3 def f():
4     global a
5     a = 10
6     print(a)
7
8 print(a) # Affiche 5
9 f()     # Affiche 10
10 print(a) # Affiche 10
```

Le programme ci-dessus est volontairement *petit*, de manière à rester *humainement lisible*.

Dès lors, on peut imaginer que dans un programme colossal de plusieurs milliers de lignes de code, le suivi de la valeur d'une variable globale modifiée depuis une ou plusieurs fonctions peut s'avérer vite compliqué.



Lorsqu'une fonction modifie la valeur d'une variable globale, on parle d'une fonction à « effet de bord » ou d'une fonction « à effet secondaire ».