Les structures conditionnelles

Nous allons commencer par un exemple. Notre but est de demander l'âge de l'utilisateur et de déterminer si il/elle a plus de 16 ans par exemple.

```
  age = int(input("Quel est votre âge ? ")) # Souvenez-vous, il faut
convertir en un entier
  if age > 16: # Si l'âge est strictement supérieur à 16 (ans)
      print("Vous avez plus de 16 ans :)")
```

Voyons ce qui s'est passé. Nous avons commencé par demander son âge au visiteur tout en nous assurant de récupérer un entier. Nous avons ensuite effectué un test : si la variable age est supérieur à 16, on affiche un message. Dans le cas contraire rien ne se passe. Nous verrons les différents tests que l'on peut effectuer dans la suite de ce chapitre, mais pour l'instant concentrons-nous sur la structure.

Vous avez sans doute remarqué la présence d'espaces devant le print. On appelle indentation le fait de décaler une ou plusieurs lignes de code à l'aide d'espaces, généralement 4 comme recommandé par Python, ou d'une tabulation (touche Tab à gauche de la touche A). Cette indentation est requise pour le bon fonctionnement de la condition. En effet, Python a besoin de savoir ce qui doit être exécuté uniquement si la condition est vérifiée et ce qui sera toujours exécuté. Les lignes indentées après le if forment les instructions qui seront uniquement exécutées si la condition est vérifiée. Les lignes alignées avec le if seront, elles, toujours exécutées.



A partir de maintenant, vous êtes susceptibles de rencontrer une **IndentationError**. Ce type d'erreur indique que vous avez un problème d'indentation. Il peut s'agir d'un if qui n'est suivi d'aucun bloc indenté ou d'un nombre incohérent d'espace utilisé tout au long de votre programme : si vous utilisez 4 espaces pour indenter la première fois, n'en mettez pas 5 au if suivant, sinon Python va râler. Si vous utilisez Tab , vous avez probablement oublié ou mis une tabulation de trop

Maintenant, ajoutons quelques éléments :

```
  age = int(input("Quel est votre âge ? "))
  if age > 16: # Si l'âge est strictement supérieur à 16 (ans)
      print("Vous avez plus de 16 ans :)")
  elif age < 0: # Si l'âge est strictement inférieur à 0
      print("Tu te moquerais pas de moi ?")
 else: # Dans tous les autres cas
     print("Tu es encore un peu jeune")
 print("Au revoir")
```

Détaillons les trois mots clé introduits :

-— **if**, c'est-à-dire « si », marque le début de la structure conditionnelle. Il est suivi d'une condition. Il n'y a qu'un seul if dans une même structure conditionnelle. C'est également la première condition à être vérifiée.

—- **elif**, qui correspond à « sinon si », est aussi suivi d'une condition. Il peut y avoir plusieurs elif dans une même structure de condition. La condition ne sera testée que si aucune des conditions précédentes se trouvant dans la même structure n'est vérifiée. De plus, les elif seront testés dans l'ordre dans lequel ils se trouvent dans votre fichier.

—- **else** correspond à « sinon » ou « dans tous les autres cas ». Il n'est pas suivi d'une condition. Il n'y a forcément qu'un seul else par structure. En effet, son rôle est d'être nécessairement exécuté si aucune des conditions au-dessus de lui n'a été vérifiée.

Les instructions elif et else peuvent ou non faire partie de votre structure.

Détaillons maintenant notre exemple. Une fois l'âge récupéré, plusieurs scénarios sont possibles :

—- si age est strictement plus grand que 16, print("Vous avez plus de 16 ans :)") est exécuté.

—- sinon si age est strictement plus petit que 0, print("Tu ne te moquerais pas de moi ?") est exécuté.

—- dans tous les autres cas, ce qui correspond ici à un âge entre 0 et 16, print("Tu es encore un peu jeune") est exécuté.

—- Dans tous les cas, c'est-à-dire peut importe l'âge, print("Au revoir") est exécuté.

Dès qu'une condition est vérifiée, le bloc d'instructions associé est exécuté et le programme reprend son exécution à la sortie de la condition, c'est-à-dire dans notre exemple à print("Au revoir"). Ainsi, les conditions suivantes ne seront pas testées et directement ignorées.

Vous pouvez, à la suite de la première condition, en ajouter une deuxième. Celle-ci commencera par un if et sera indépendante de la précédente. Comme toujours, n'hésitez pas à faire des tests.

From:
https://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
https://chanterie37.fr/fablab37110/doku.php?id=debuter_en_python:if&rev=1726348861

Last update: **2024/09/14 23:21**