

Gestion des entrée et des sorties – lorsqu'on a besoin d'interagir avec le programme.

```
age=input("quel est votre âge ?")
#demande à l'utilisateur son âge et le stocke dans la mémoire age#
age=int(age)
#transforme la réponse de l'utilisateur en nombre entier#
print(age)
#affiche l'âge de l'utilisateur#
```



Instruction conditionnelle, l'équivalent de

```
#instruction conditionnelle ; si l'âge est inférieur à 0, on renvoie un message#
#sinon, on affiche l'âge indiquée par l'utilisateur#
#remarquer les ":" qui indique l'utilisation des instructions spécifiques "if" et "else"#
#remarquer l'indentation c'est à dire le décalage de 4 espaces#
#la fin de l'indentation indique la sortie des instructions conditionnelles "if" ou "else"#
if age<0:
    print("impossible cela ne peut pas être un nombre négatif")
else :
    print(age)
```



La répétition non bornée WHILE ( ou TANT QUE), l'équivalent de

```
age=-1
#cette boucle permet la répétition de la demande de l'âge tant que l'utilisateur donne un âge négatif#
#remarquer les deux points qui finissent l'instruction spécifique while#
#remarquer l'indentation c'est à dire le décalage de 4 espaces #
#la fin de l'indentation indique la sortie de la boucle while#
while age <0 :
    age=input("quel est votre âge ?")
    age=float(age)
    #ici on transforme l'âge en nombre décimal et non entier#
print(age)
```



La répétition bornée FOR ( ou POUR), l'équivalent de

```
#répéter 3 fois le fait d'avancer de côté et de tourner à 120°#
#remarquer les ":" qui indique l'utilisation de l'instruction spécifique for#
#remarquer l'indentation c'est à dire le décalage de 4 espaces #
#la fin de l'indentation indique la sortie de la boucle for#
for i in range(3):
    forward(cote)
    left(120)
```