

## MINI MEMO PYTHON

### Affecter des valeurs à des variables

`age=15.3` .....Stocke le nombre décimal 15,3 dans `age`  
`age=age+1`  
.....Stocke la valeur de `age+1` dans `age` ⇒ Incréméte (= augmente) la valeur de `age` de 1  
`age = int(age)`.....Calcule la valeur entière de `age` et la stocke dans `age`  
`age="15"` .....Stocke le texte "15" dans `age`  
`age=int(age)`.....Convertit le texte "15" en nombre entier 15

### Gestion des entrées

`age=input("quel âge avez-vous ?")`  
.....Demande à l'utilisateur son âge et stocke le texte tapé **en tant que texte** dans `age`  
`age=int(input("quel âge avez-vous ?"))`  
.....Affiche la question, convertit le texte tapé en nombre entier, puis le stocke dans `age`  
`age=float(input("quel âge avez-vous ?"))`  
.....Affiche la question, convertit le texte tapé en nombre décimal, puis le stocke dans `age`

### Gestion des sorties

`age=15`  
`unite="ans"`  
`print("Votre age :",age,unite)`  
.....Affiche "Votre âge :", suivi du contenu de la variable `age`, suivi du contenu de `unite`

### Instruction conditionnelle if - else (si - sinon)

<code>if age&lt;0:</code>	Si l'âge est négatif,
<code>print("nombre négatif !")</code>	on affiche "Nombre négatif !"
<code>print("Impossible !")</code>	on affiche "Impossible !"
<code>else:</code>	Sinon
<code>print("Votre âge :",age,"ans")</code>	on affiche l'âge

Remarquer le ":" à la suite duquel on a une indentation (décalage de 4 espaces) au début des lignes suivantes qui contiennent les instructions à faire si la condition est vraie.  
La fin de l'indentation indique la fin du `if` (si) ou du `else` (sinon)

### Les opérateurs de comparaison

Les opérations suivantes renvoient **true** (vrai) ou **false** (faux)

`x==y` ..... `x` est-il égal à `y` ?  
`x!=y` ..... `x` est-il différent de `y` ?  
`x>y` `x<y` `x>=y` `x<=y` ..... `x > y` ?    `x < y` ?    `x ≥ y` ?    `x ≤ y` ?  
`3<x and x<=7` ..... a-t-on  $3 < x \leq 7$  ?  
`not(3>=x or x>7)` ..... est-ce que  $3 \geq x$  ou  $x > 7$  est faux ? ⇔ a-t-on  $3 < x \leq 7$  ?

⚠ Ne pas confondre la **commande** `x=y` qui stocke le contenu de `y` dans `x`  
et l'**opération** `x==y` qui renvoie `true` ou `false` quand `x` est égal ou non à `y`

## La répétition non bornée while (tant que)

```
age=-1
while age<0:
    age=float(input("Quel âge avez-vous?"))
print("Votre âge :",age,"ans")
```

On stocke -1 dans age  
Tant que age est négatif  
on demande l'âge  
On affiche age

Cette boucle while permet la répétition de la demande de l'âge tant que l'utilisateur donne une valeur négative

La fin de l'indentation marque la sortie de la boucle while

## La répétition bornée for (pour)

```
for i in [1,2,3,5,7,9,11]:
    print("le double de",i,"est",2*i)
print("j'ai fini !")
```

Pour chacune des valeurs de la liste, la variable i prend les valeurs successives de la liste, on affiche i et son double.

Une fois tous les nombres de la liste passés, on poursuit le programme.

### i peut être pris dans un intervalle

range(b) : le nombre prendra les valeurs dans [0 ; b[ (b exclu de l'intervalle)  
for i in range(5) : i prendra pour valeurs : 0 ; 1 ; 2 ; 3 et 4

range(a,b) : le nombre prendra les valeurs dans [a ; b[ (b exclu de l'intervalle)  
for i in range(2,5) : i prendra pour valeurs : 2 ; 3 et 4

range(a,b,c) : le nombre prendra les valeurs de [a ; b[ par pas de c  
for i in range(5,11,2) : i prendra les valeurs : 5 ; 7 ; 9

## Les opérations de base

addition, soustraction, multiplication	+ - *	2+5 → 7	2-5 → -3	2*5 → 10
puissance	**	5**3 → 5 <sup>3</sup> = 125		
division	/	7/2 → 3.5		
reste de division entière	%	7%3 → 1		
quotient de division entière	//	7//3 → 2		
Valeur absolue	abs( )	abs(2-5) → 3		

## Importations de fonctions particulières

Instructions from à placer en début de programme pour utiliser des fonctions particulières

from math import \* ..... importe toutes les fonctions du module math (sqrt, ...)

from random import \* ..... importe les fonctions aléatoires

randint(0, 3) ..... donne un entier entre 0 et 3 inclus

random() ..... donne une valeur décimale de l'intervalle [0 ; 1[

from turtle import \* ..... importe les fonctions de la tortue

pendown() penup() ..... baisse, lève le stylo

showturtle() hideturtle() ..... montre, cache la tortue

forward(20) left(90) right(90) ..avance de 20, tourne à gauche, à droite de 90°