

COURS 3 : Programmer des instructions conditionnelles – 1 heure

Dans un algorithme, on peut être amené à exécuter une ou plusieurs instructions uniquement si une certaine condition est vérifiée, c'est ce que l'on appelle des instructions conditionnelles.

Si la condition n'est pas vérifiée, on peut exécuter un autre bloc d'instructions, soit rien faire et continuer le reste du programme.

Notation pour la condition if

>= : supérieur ou égal

<= : inférieur ou égal

== : égal

Programme 14 :

```
1 x=float(input("Saisir un nombre : "))
2 if x>=0:
3     print ("nombre est positif.")
4 else :
5     print ("le nombre est strictement négatif.")
6     print ("son opposé est : ")
7     print (-x)
```

Étude du programme :

Ligne 1 : la variable 'x' sera de type numérique de classe décimale et cette variable posera une question (input)

Ligne 2 : **si (=if)** x est supérieur ou égal à 0

Ligne 3 : **alors** on affiche « le nombre est positif » si la condition de la ligne 2 est vérifié

Ligne 4 : **sinon (=else)**

Ligne 5 : on affiche « le nombre est négatif »


Ligne 6 : on affiche « son opposé est : »

Ligne 7 : on affiche l'inverse du nombre choisit

On dit que print est **indenté** dans la fonction else


QUESTIONS :

- 1- Exécuter le programme 14 en saisissant d'abord -2.4, puis 10.5 comme valeur de x.
- 2- Expliquer les deux affichages obtenus.

Powered by  trinket
Saisir un nombre : -2.4
le nombre est strictement négatif.
son opposé est :
2.4

La condition if n'est pas vérifiée alors la ligne 3 n'est pas activée.

Le programme poursuit la lecture des lignes 4, 5, 6 et 7.

Powered by  trinket
Saisir un nombre : 10.5
nombre est positif.

La condition if est vérifiée ($x \geq 0$) alors les lignes 4, 5, 6 et 7 ne seront pas lues.

Le programme s'arrête à la ligne 3


Programme 15 :

Modifier la dernière ligne du programme 14

```
1 x=float(input("Saisir un nombre : "))
2 if x>=0:
3     print ("nombre est positif.")
4 else :
5     print ("le nombre est strictement négatif.")
6     print ("son opposé est : ")
7 print(-x)
```

QUESTIONS :

- 1- Exécuter le programme en saisissant 10 pour valeur de x.
- 2- Expliquer le résultat obtenu.

Powered by  trinket
Saisir un nombre : 10
nombre est positif.
-10.0

Le programme exécute la ligne 1

10 vérifie la condition SI (ligne 2) et affiche la ligne 3

La condition if étant vérifiée les lignes 4, 5 et 6 ne sont pas lu par le programme.

Le programme passe donc à la ligne suivante

La ligne 7 affiche le contraire de x (cette ligne est non indentée)

Réalisation d'un jeu :

Le programme lance un dé à 6 faces de façon aléatoire où l'on gagne si le résultat est 6.

- Si la valeur de la variable x est 6, alors on affiche « gagné ! »
- Sinon on affiche « Perdu... ».

Dans les deux cas, le programme se poursuit en affichant « A bientôt ».

En vous aidant du programme 15 et de l'aide ci-dessous, écrire le programme 16.

Aide :

Un tirage aléatoire demande l'appel d'un module nommé **random**.

Le programme doit démarrer par la ligne : **import random**

La variable x sera donc aléatoire et comprise entre les valeurs de a à b : **x=random.randint(a,b)**



The screenshot shows a code editor interface with a toolbar at the top containing icons for a menu, a key, Python logo, play, dropdown, share, and another dropdown. Below the toolbar is a tab labeled 'main.py'. The code editor displays the following Python code:

```
1
2
3
4
5
6
7
1 import random
2 x=random.randint(1,6)
3 if x==6:
4     print("gagné !")
5 else :
6     print("perdu..")
7 print ("A bientôt !")
```

Exécuter le programme et indiquer au bout de combien de tirage le programme gagne.

Remarque :

La condition peut être de la forme **SI** condition 1 **ou** condition 2

En langage Python : **if** condition 1 **or** condition 2 :

La condition peut être de la forme **SI** condition 1 **et** condition 2

En langage Python : **if** condition 1 **and** condition 2 :

Programme 17 :

Objectif : Écrire un programme, comprenant deux variables aléatoires avec deux conditions qui permettent d'afficher gagné.

QUESTION :

- 1- Proposez un programme qui affichera gagné lorsque deux variables comprises entre 0 et 10 vérifieront les conditions suivantes :

X=1 ou y=6

Sinon le programme affichera « recommencer »



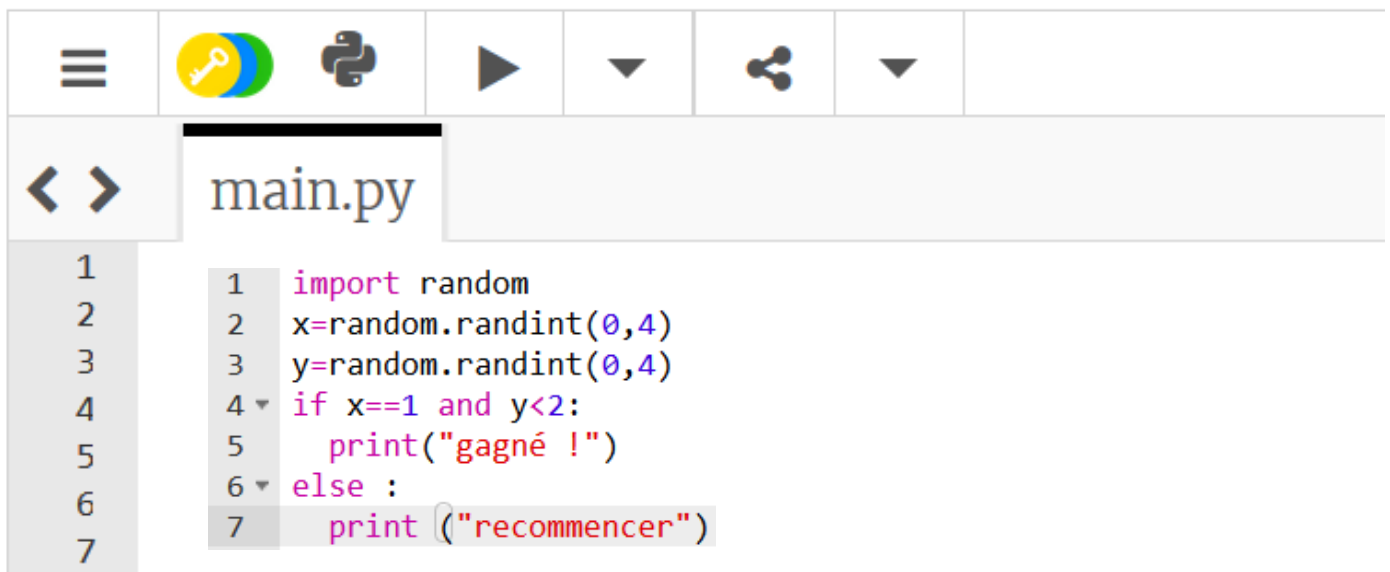
The image shows a code editor interface. At the top, there is a toolbar with icons for a menu, a key, a Python logo, a play button, a dropdown arrow, a share icon, and another dropdown arrow. Below the toolbar, the file name 'main.py' is displayed. The code is written in Python and is as follows:

```
1 import random
2 x=random.randint(0,10)
3 y=random.randint(0,10)
4 if x==1 or y==6:
5     print("gagné !")
6 else :
7     print ("recommencer")
```

Programme 18 :

Modifier le programme 17 qui vérifiera les conditions suivantes :

$X=1$ et $y<2$ pour des valeurs comprises entre 0 et 4



The screenshot shows a code editor with a toolbar at the top containing icons for a menu, a key, a Python logo, a play button, a dropdown arrow, a share icon, and another dropdown arrow. Below the toolbar, the file name 'main.py' is displayed. The code is as follows:

```
1 import random
2 x=random.randint(0,4)
3 y=random.randint(0,4)
4 if x==1 and y<2:
5     print("gagné !")
6 else :
7     print ("recommencer")
```

Programme 19 :

Modifier le programme 18 qui vérifiera les conditions suivantes :

$X\geq 1$ et $y\neq 2$



The screenshot shows a code editor with a toolbar at the top containing icons for a menu, a key, a Python logo, a play button, a dropdown arrow, a share icon, and another dropdown arrow. Below the toolbar, the file name 'main.py' is displayed. The code is as follows:

```
1 import random
2 x=random.randint(0,4)
3 y=random.randint(0,4)
4 if x>=1 and y!=2:
5     print("gagné !")
6 else :
7     print ("recommencer")
```