

➤ Import

*from random import ** Il n'y a pas beaucoup de fonctions, elles seront toutes disponibles directement sans préfixe.

Attention : Dans un programme, pour plus de sûreté, il est recommandé de n'importer que les fonctions que l'on utilise réellement (**Exemple** : *from random import randint, choice*)

➤ Des nombres aléatoires ou pseudo-aléatoires ?

Il n'y a pas de méthode pour générer des nombres réellement aléatoires sur un ordinateur : Un ordinateur ne sait pas lancer des dés !

Un programme informatique est écrit à partir d'un algorithme. Or, le propre d'un algorithme, est de se comporter toujours exactement de la même façon. Difficile donc de créer de l'aléatoire...

Les générateurs de nombres pseudo-aléatoires utilisent généralement des suites récurrentes dont les calculs sont suffisamment compliqués pour que les résultats soient prévisibles mais qui respectent certaines lois statistiques. Le problème est qu'avec un même terme initial, tout le monde aura la même suite de nombres pseudo-aléatoires ! C'est ce qui arrive quand on utilise cette fonction la première fois en classe avec des élèves ayant tous la même calculatrice neuve ! Python, utilise comme terme initial un nombre dépendant du temps depuis lequel l'ordinateur est allumé...

➤ Quelques fonctions du module *random*

random() : Crée un nombre à virgule flottante (*float*) aléatoire appartenant à [0.0 ; 1.0]

randrange(n) : Crée un nombre entier (*int*) aléatoire appartenant à [0 ; n - 1] (n doit être un entier)

randint(a, b) : Crée un nombre entier (*int*) aléatoire appartenant à [a ; b] (a, b doivent être des entiers)

choice(objet) : Choisit un élément aléatoire de l'objet cité (chaîne de caractères, liste, tuple)

Exemples :

choice("aeiouy") (choisit une voyelle au hasard)

choice(["lundi", "mardi", "mercredi", "jeudi", "vendredi", "samedi", "dimanche"])

sample(objet, n) : Crée une liste de taille n d'éléments aléatoires de l'objet cité. (chaîne, liste, tuple)

Exemples :

sample("aeiouy", 2) (créé une liste de deux voyelles aléatoires différentes)

sample(["lundi", "mardi", "mercredi", "jeudi", "vendredi", "samedi", "dimanche", 3])

Remarque :

Si un élément est présent plusieurs fois dans l'objet, il peut alors être choisit autant de fois qu'il s'y trouve !

shuffle(liste) : Remplace la liste nommée par une liste avec les mêmes éléments dans un ordre aléatoire.

seed(objet) : Initialise le générateur de nombres pseudo-aléatoires à partir de la valeur de l'objet cité.

Il existe d'autres fonctions présentes dans ce module, elles ont pour la plupart une action sur le mode de distribution des nombres aléatoires générés.