

Syntaxe		Rôle de la fonction	Exemples en Python
Algorithme	Python		
Abs (n)	abs (n)	Retourne la valeur absolue de n.	x = abs (-5) → x = 5
Aléa (Vi, Vf)	randint (Vi, Vf)	Retourne un entier aléatoire appartenant à l'intervalle [Vi, Vf].	x = randint (5,20) → 5 ≤ x ≤ 20
Arrondi (n)	round (n)	Retourne l'entier le plus proche de n.	x = round (3.7) → x = 4
Ent (n)	int (n) ou floor(n)	Retourne la partie entière de n.	x = int (7.2) → x = 7
RacineCarré (n)	sqrt (n)	Retourne la racine carrée de n (n doit être positif).	x = sqrt (2) → x = 1.41

N.B :

- L'utilisation des fonctions mathématiques nécessite l'importation de la bibliothèque math (from math import *).
- L'utilisation de la fonction randint nécessite l'importation de la bibliothèque random (from random import *).

Les fonctions sur le type caractère

Syntaxe		Rôle de la fonction	Exemples en Python
Algorithme	Python		
Ord (car)	ord (car)	Retourne le code ASCII du caractère car.	cod = ord ('A') → cod = 65
Chr (cod)	chr (cod)	Retourne le caractère dont le code ASCII est cod.	car = chr (98) → car = 'b'

Les fonctions sur le type chaînes de caractères

Syntaxe		Rôle de la fonction	Exemples en Python
Algorithme	Python		
Majus (ch)	ch.upper ()	Convertit la chaîne ch en majuscules.	ch = 'bonjour' ch = ch.upper () → ch = 'BONJOUR'
Long (ch)	len (ch)	Retourne le nombre de caractères de la chaîne ch.	x = len ('Informatique') → x = 12
Pos (ch1 , ch2)	ch2.find (ch1)	Retourne la première position de la chaîne ch1 dans la chaîne ch2.	ch = 'informatique' x = ch.find ('i') → x = 0
Convch (n)	str (n)	Convertit une valeur numérique n en une chaîne de caractères.	ch = str (17.2) → ch = '17.2'
Valeur (ch)	int (ch) float (ch)	Convertit une chaîne de caractères ch en une valeur numérique, si c'est possible.	x = int ('12') → x = 12 x = float ('14.3') → x = 14.3
Estnum (ch)	ch.isnumeric() ch.isdigit ()	Retourne Vrai si la chaîne ch est convertible en une valeur numérique, elle retourne Faux sinon.	ch = '154' → ch.isnumeric () donne True ch = '9?3' → ch.isnumeric () donne False
Sous_Chaine(ch , d , f)	ch [d : f]	Retourne une partie de la chaîne ch à partir de la position d jusqu'à la position f (f exclue).	ch = 'Bonjour' ch1 = ch [0:3] → ch1 = ch [:3] → ch1 = 'Bon' ch2 = ch [3:len(ch)] → ch2 = ch [3:] → ch2 = 'jour' ch3 = ch [2:5] → ch3 = 'njo'
Effacer(ch , d , f)	ch [: d] + ch [f :]	Efface des caractères de la chaîne ch à partir de la position d jusqu'à la position f (f exclue).	ch = 'Bonjour' ch = ch [:2] + ch[5:] → ch = 'Bour' ch = ch [:0] + ch[3:] → ch [3:] → ch = 'jour' ch = ch [:3] + ch[len(ch):] → ch [:3] → ch = 'Bon'