mes remarques :

Livret Python 1NSI

arb 2023-2024

Les variables numériques

Syntaxe	Description
x = 1.4	Affecte la valeur 1.4 à la variable x.
x = x + 1	Incrémente la variable x de 1.
a + b	Addition de a et b.
a*b	Multiplication de a et b.
a/b	Division de a par b.
a**b	Élévation de a à la puissance b.
sqrt(a)	Racine carrée de a. ⚠Il faut importer sqrt depuis la bibliothèque math : from math import sqrt.
floor(a)	Partie entière de a ⚠Il faut importer floor depuis la bibliothèque math : from math import floor.
a//b	Quotient de la division euclidienne de a par b.
a%b	Reste de la division euclidienne de a par b.
int(x)	Reconnaît ou convertit le nombre ou la chaîne de caractères x en un entier. ⚠int(2.76) convertit la valeur 2.76 en donnant la troncature : l'entier 2.
float(x)	Reconnaît ou convertit le nombre ou la chaîne de caractères x en nombre flottant. ⚠float("235.76") convertit la chaîne de caractères "235.76" en nombre flottant : 235.76
str(x)	Reconnaît ou convertit le nombre x en la chaîne de caractères . ⚠str(176) convertit la valeur 176 en "176" , une chaîne constituée des caractères 1, 7 et 6.

Les bibliothèques

from [bibliotheque] import [fonction] — Depuis la bibliothèque, on importe la fonction.

Bibliothèque	Fonction	Description
Math	*	Importe toutes les fonctions de la bibliothèque math qui contient toutes les fonctions mathématiques utilisées au lycée.
	sqrt	Importe la fonction racine carrée.
	pi	Importe la valeur de pi (3.141592653589793).
	sin, cos, tan	Importe les trois fonctions trigonométriques principales.
random	*	Importe toutes les fonctions de la bibliothèque random qui contient toutes les fonctions relatives aux statistiques et aux probabilités utilisées au lycée.
	randint	randint(a,b) retourne aléatoirement un nombre entier entre a et b (bornes comprises).
	random	random() retourne aléatoirement un nombre décimal compris entre 0 et 1 exclus.
	choice	choice(tab) permet de tirer au sort un des éléments de la liste tab.
	shuffle	shuffle(tab) mélange aléatoirement les éléments de la liste tab.
	sample	sample(tab,k) permet de tirer au sort k éléments de la liste tab.
	seed	seed(123) permet de "bloquer" les fonctions aléatoires pour qu'elles renvoient toujours la même valeur.

Les commentaires

Les commentaires, qui ne seront pas interprétés par l'ordinateur doivent être précédés su symbole "#".

Les chaînes de caractères

chaine1 + chaine2	Concaténation des chaînes chaine1 et chaine2
n*chaine	Répétition de 3 fois la chaîne chaine.
chaine1 < chaine2	Teste si chaine1 est avant dans l'ordre alphabétique que chaine2.
len(chaine)	Retourne le nombre de caractères de la chaîne chaine.
chaine[n]	Extrait le caractère placée à la position n de la chaîne chaine.
\n	Dans une chaîne de caractère permet un retour à la ligne. Par exemple "une première ligne \n et une seconde" .

Les interaction programme/utilisateur

print(variable)	Affiche le contenu de la variable dans la console.
print("message")	Écrit le texte entre les guillemets dans la console.
variable = input("message")	Demande une chaîne de caractères à l'utilisateur qui sera stockée dans la variable. ⚠Pour demander un nombre entier il faut utiliser variable = int(input("message")) et un décimal variable = float(input("message")).

L'instruction conditionnelle if

c = False	Initialise le booléen c.
a and b	True si a et b sont à True.
a or b	True si a ou b (ou les deux) sont à True.
not(a)	True si a est à False et réciproquement.
if booléen : instruction(s)	Si le booléen est True, exécute la (ou les) instruction(s) indentées.

if booléen :
  instruction(s) 1
else :
  instruction(s) 2
	Si le booléen est True, exécute la (ou les instruction(s) 1, sinon exécute la (ou les) instructions 2.

if booléen 1 :
  instruction(s) 1
elif booléen 2 :
  instruction(s) 2
else :
  instruction(s) 3
	Si le booléen 1 est True, exécute la (ou les instruction(s) 1, sinon , si le booléen 2 est True exécute la (ou les) instructions 2, sinon exécute la (ou les) instruction(s) 3.

a==b	Teste si a est égal à b. ⚠Remarquez la présence des deux symboles "égal".
a!=b	Teste si a est différent de b.
a<b (resp. a >b)	Teste si a est strictement inférieur à b (resp. si a strictement supérieur à b).
a<=b (resp. a >=b)	Teste si a est inférieur ou égal à b (resp.si a supérieur ou égal à b).

La boucle bornée for

for variable in range(n) : instruction(s)	Exécute en boucle la (ou les) instructions pour une variable allant de 0 à n-1.
for variable in range(n, m) : instruction(s)	Exécute en boucle la (ou les) instructions pour une variable allant de n à m-1.
for variable in range(n, m, k) : instruction(s)	Exécute en boucle la (ou les) instructions pour une variable allant de n à m-1 avec un pas de k.
for caractere in chaine : instruction(s)	Exécute en boucle la (ou les) instructions pour chaque caractère de la chaîne de caractères chaine.

La boucle non bornée while

while booleen : instruction(s)	Exécute en boucle la (ou les) instructions tant que le booleen est True.

Les fonctions

def nomDeLaFonction(arg1, arg2) :
  instruction(s)
  return resultat
	Une fonction est un programme qui porte un nom et qui peut utiliser un, plusieurs ou aucun paramètres (arg1, arg2...).
	⚠L'instruction return n'est pas obligatoire.

LYCÉE LA BOURDONNAIS
Livret Python 1NSI

Les dictionnaires

dico = {"one" :1, "two" :2}	création d'un dictionnaire
dict(liste)	Convertir une liste de listes à deux éléments en dictionnaire
dict(one=1, two=2, three=3,four=4)	Création d'un dico à partir de couples
{x :x**2 for x in range(1,5)}	Création par compréhension
dico.keys()	Accès aux clés
dico.values()	Accès aux valeurs
idico.items()	Accès aux couples clès-valeurs
valeur in dico	Pour tester l'appartenance d'une clè
dico[clé]	Accès à une valeur
dico.get(clè)	Accès à une valeur (renvoie None si non présente)
dico[cle]=nouvelle_valeur	Modification ou création d'une valeur
len(dico)	Longueur d'un dictionnaire
del(dico[clè])	Supprimer un élément
dico1.keys()&dico2.keys()	Retourne les clés communes
d2.keys()-d1.keys()	Retourne les clés présentent dans d2 mais pas dans d1
d1.items()&d2.items()	Retourne les couples clé-valeur communs
clear(dico)	Vider un dictionnaire

arb 2023-2024

Les tuples

⚠Les valeurs d'un dictionnaire ne sont pas modifiables.

t = ()	Création d'un tuple vide
t = (1, 2, 3)	Création d'un triplet
t[indice]	Accès aux éléments
len(t)	Longueur d'un tuple
t = (0,) + t	Ajout d'un élément au début
t = t + (4,)	Ajout d'un élément en fin
élément in t	Tester l'appartenance d'un élément
*t	Disperser un tuple

Les listes

L = []	liste vide
L = [3, "oui"] L = [3, 4, 2, [0, 1, 2]]	Exemples de liste contenant des objets de types différents
L = [0]*5	Créé une liste de 5 objets, tous égaux à 0.
L = list(range(1, 11, 3))	Créé la liste des nombres de 1 à 11 avec un pas de 3 : [1, 4, 7, 10]
L[0]	Premier objet de la liste, ou objet de rang 0.
L[k]	k+1 ème objet de la liste, ou objet de rang k
L[-1]	Dernier objet de la liste.
len(L)	Longueur, ou nombre d'objets, de la liste
L.append(objet)	Ajoute un objet en fin de liste
L.remove(3)	Enlève l'objet ou les objets de valeur 3 de la liste
L[n :m] = []	Supprime les objets entre les rangs n à m-1
L1[n :m] = L2	Remplace les objets entre les rangs n à m-1 par la liste L2
del(L[k])	Enlève l'objet de rang k de la liste
L.insert(rang,objet)	Insère un objet au rang k.
L.sort()	Ordonne les valeurs d'une liste dans l'ordre croissant
L.reverse()	Inverse la liste
L1 + L2	Concatène (rassemble) deux listes
L1 = L2[:]	Copie la liste L2 dans la liste L1. ⚠L1 = L2 ne fonctionne pas car cela copie uniquement l'adresse de la liste. Cette méthode ne fonctionne pas pour les listes de listes de listes (tableaux à 3 dimensions).
listeA = [f(x) for x in listeD\ if test(x)]	Crée la listeA avec les images par la fonction f des élèments de listeD qui vérifient test(x)