

Quelques instructions d'EduPython



Pour utiliser la bibliothèque lycee, vos programmes doivent commencer par cette ligne
from lycee import * (automatiquement insérée si vous cliquez sur *Nouveau Fichier* puis *Lycée*)

Affectations / Calculs :

a = 3	Affecte à la variable a la valeur 3	2 ** 3	donne 8 (=2 ³)
a = a + 1	Calcule a + 1 et affecte le résultat à la variable a (c'est à dire que a augmente de 1)	14 // 3	donne 4 (quotient de 14 ÷ 3)
		reste(14, 3)	donne 2 (reste de 14 ÷ 3)

Entrées/Sortie :

- Afficher à l'écran :
print(a) Affiche la valeur de la variable a (si elle existe)
print("a") Affiche la lettre « a »
print("La valeur de a est :", a) Affichage mixte (texte et valeurs)
- Demander un nombre et stocker la réponse dans une variable
x = demande("Nombre de côtés ?")
La question s'affiche et la réponse est attribuée à la variable numérique x
- Demander un texte :
rep=texte_demande("Quelle est la couleur du cheval blanc d'Henry IV ?")

Tests :

Programme demandé	Algorithme	Programme en Python	Remarques
Tester si un nombre entré est pair ou non : Nous allons regarder si le nombre est divisible par 2, c'est-à-dire si son reste vaut 0	Demander un nombre x Si le reste de x ÷ 2 vaut 0 Alors Afficher « Pair » Sinon Afficher « Impair » Fin du SI	x=demande("nombre?") if reste(x,2) == 0 : print("pair") else : print("impair")	En Python, le ALORS se traduit par deux points et un décalage des instructions (alinéa). De même que pour le SINON qui se traduit par else :

== Egal à != Différent de >= Supérieur ou égal

Boucles :

Comme dans la plupart des langages, il existe en Python principalement deux manières de réaliser une boucle, c'est-à-dire une répétition d'un bloc d'instructions. Comme pour la commande *si*, la partie à répéter sera indentée vers la droite, ce qui permet en plus une bonne visibilité de l'algorithme.

for v in ["a","e","i","o","u","y"] : Effectue la boucle, la variable v prenant à chaque tour successivement les valeurs de la liste (ici les voyelles)
 BLOC D'INSTRUCTIONS...

for i in range(5) : ici, i prend tour à tour les nombres de 0 à 4 (intervalle [0 ; 5 [)
 BLOC D'INSTRUCTIONS...

for i in range(3,11,2) : Parcourt l'intervalle [3 ; 11 [avec un pas de 2 : 3, 5, 7 et 9
 BLOC D'INSTRUCTIONS...

while i<10 : Exécute la suite d'instructions tant que i<10: le test sera effectué au départ et à chaque fois que bloc d'instructions est fini, avant de l'exécuter à nouveau
 BLOC D'INSTRUCTIONS...