Cégep de Sherbrooke

**Développement d’applications avec Objets Connectés. Laboratoire 3 – Collections**

# Buts :

L’utilisation de Listes L’utilisation de Tuples L’utilisation de Dictionnaires

# Exercice 1. Écrire les scripts qui répondent aux requis suivants :

1. Le script nommé nombre\_elements\_tuple.py qui donnera le nombre de valeurs qui forment une tuple donné comme constant au début du script, par exemple, pour le tuple t1=(1,2,3) votre script affichera la valeur de 3.

Tuple1=1,2,3

Print(max(tuple1))

1. Faire un script nommé semaine\_laboral.py qui prendra un tuple nommé jours\_semaine avec les noms de sept jours de la semaine, en commençant par dimanche et créera un nouveau tuple nommé semaine\_laboral qui contiendra seulement les jours de lundi à vendredi.

Jours\_semaine = “Dimanche”, “Lundi”, “Mardi”, “Mercredi », « jeudi », « vendredi », « Samedi »

Semaine\_laboral = jours\_semaine[1 :6]

1. Faire un script nommé les\_mots.py qui demandera à l’utilisateur d’entre un mot, et si le mot a déjà été entré le script affichera « La valeur a déjà été entré » sinon affichera « Nouvelle valeur » et procédera a la garder. Si le mot « Affiche » est rentré votre programme affichera la liste de mots à ce moment.

listeMot[“”]

def question():

mot=input(“Entrer un mot : « )

for word in listeMot :

if (mot ==word) :

print(La valeur a déjà été entre)

question()

elif (mot == Affiche)

print(listeMot)

question()

break

print(Nouvelle valeur)

listeMot.append(mot)

while(True) :

question()

1. Pour créer de nombres aléatoires vous pouvez écrire les lignes suivantes :

import random print(random.randint(-30, 30))

Ces lignes vous permettront de générer de valeurs aléatoires entre -30 et 30 à chaque fois que la méthode random.randint(-30, 30) est appellée. Dans les lignes, la valeur va être imprimée.

Faire un script nommé couleurs\_LED.py dans lequel vous avez un tuple nommé ranges, définit de la manière suivante :

ranges = (-30, -10,10, 30)

vous devez générer 10 valeurs aléatoires et vous devez afficher la couleur d’une LED (que sous peu on pourra allumer) selon le tableau suivant :

|  |  |  |
| --- | --- | --- |
| Limite inférieure | Limite supérieure | Couleur de la LED |
| -∞ | -30 | Bleu |
| -29 | -10 | Vert |
| -9 | 10 | Jaune |
| 11 | 30 | Orange |
| 31 | +∞ | Rouge |

Import random

Ranges = (-30, -10, 10, 30)

listeRand = []

for i in range(10) :

listeRand.append(random.randint(ranges[0],ranges[3]))

print(listeRand)

for num in listeRand:

if (num <= -30):

print(“Bleu”)

elif num <= -10

vert

elif num <= 10

jaune

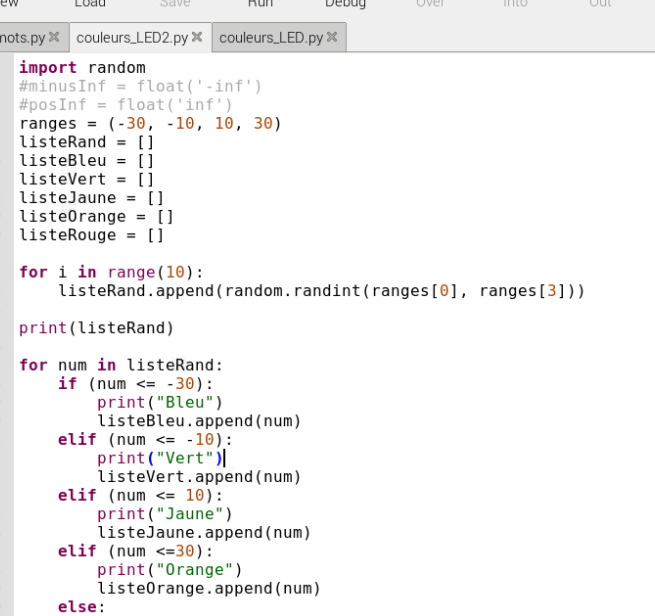
elif num <=30

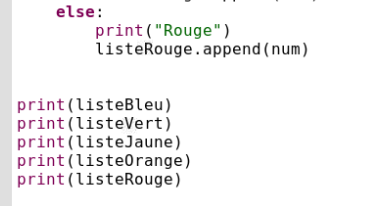
orange

else:

rouge

1. Faire une copie de votre script couleurs\_LED.py en couleurs\_LED2.py vous devez modifier le programme pour pouvoir garder les valeurs générés dans une liste par couleur selon les ranges indiqués. La valeur doit se garder le nombre de fois qu’elle apparaît. À la fin de l’exécution, vous devez afficher le contenu de chacune des listes.





1. Faire une copie de votre script couleurs\_LED2.py en couleurs\_LED3.py, vous devez modifier le programme pour garder une seule fois la valeur généré de manière aléatoire. À la fin de l’exécution, vous devez afficher le contenu de chacune des listes.

import random

#minusInf = float('-inf')

#posInf = float('inf')

ranges = (-30, -10, 10, 30)

listeRand = []

listeBleu = []

listeVert = []

listeJaune = []

listeOrange = []

listeRouge = []

x=0

for i in range(10):

listeRand.append(random.randint(ranges[0], ranges[3]))

if (i > 0):

x+=1

for j in range(10):

if (x > j):

if (listeRand[x] == listeRand[j]):

del listeRand[x]

x-=1

print(listeRand)

for num in listeRand:

if (num <= -30):

print("Bleu")

listeBleu.append(num)

elif (num <= -10):

print("Vert")

listeVert.append(num)

elif (num <= 10):

print("Jaune")

listeJaune.append(num)

elif (num <=30):

print("Orange")

listeOrange.append(num)

else:

print("Rouge")

listeRouge.append(num)

print(listeBleu)

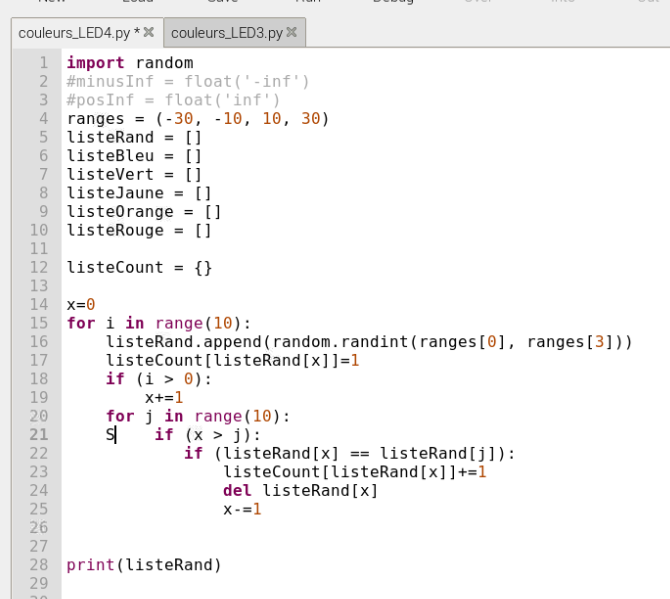
print(listeVert)

print(listeJaune)

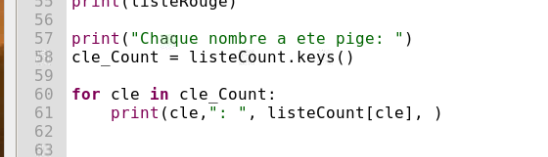
print(listeOrange)

print(listeRouge)

1. Faire une copie de votre script couleurs\_LED3.py en couleurs\_LED4.py et ajoutez la possibilité de compter combien de fois que chaque valeur a été généré. À la fin de l’exécution, vous devez afficher le contenu de chacune des listes.





 Présentez votre travail au Professeur