Programación Y Robotica
3º y 4º ESO 24_25

Clase 5 – Python: for, tuplas y formatos-2

Noviembre 2024

Colegio Santo Domingo FESD - Madrid

Profesor voluntario :

Jose Carlos Santamaria Poza : jcspoza@gmail.com

3y4ESO-CL5 – Indice / Objetivos

- Tipo Tuplas **tuple**: objeto compuesto
- Bucles **for** para recorrer objetos compuestos y rangos
- + formatos

1. Tuplas objetos compuestos
2. Bucles con for : aprender
3. Cambiar el programa de Bromas para usar tuplas y for : analisis
4. Hacer Un programa de cuenta adelante con for y formatos f-string (básico) : analisis
5. Reto CL5: Mostrar lista de primos hasta 150 (dados) alineados a la izq en 3 posiciones enteras

Pensar: Tuplas : objetos compuestos

Definición de tuplas

Python tiene varios tipos de datos compuestos, utilizados para agrupar valores, el más sencillo es la tupla

1. Puede almacenar varios elementos en una misma variable
2. Una tupla está formada por un número de valores separados por comas
3. Se puede acceder a cada elemento de manera ordenada
4. No se puede modificar el valor (Inmutable) una vez que se han creado
5. Son más eficientes en el sentido de usar menos espacio y tiempo de ejecución que otros objetos compuestos

Link a tabla de operaciones con Tuplas

```
>>> tLetrasyNum = 'd','c','b','a',4,3,2,1
>>> type(tLetrasyNum)
<class 'tuple'>
>>> len(tLetrasyNum)
8
>>> 4 in tLetrasyNum
True
>>> 'A' in tLetrasyNum
False
>>> tLetrasyNum[0]
'd'
>>> tLetrasyNum[6]
2
>>> tLetrasyNum.index('a')
3
>>> tLetrasyNum.count('a')
1
>>> tLetrasyNum[1:4]
('c', 'b', 'a')
```

Pensar: for : recorrer objetos compuestos

10'

Definición de for

- En Python **for** se usa para construir **bucles que recorren objetos compuestos**
- En estos bucles el número de repeticiones **está definido de ANTEMANO** por la longitud del objeto compuesto
- Los objetos compuestos que se pueden usar en bucles **for tienen que ser 'iterables'** o dicho de otra forma que puedan ser indexados
- El uso más simple de **for** es con un objeto compuesto de tipo ***range*** (que es iterable)
- Parametro de **range**(inicio, fin, paso)
  - inicio, fin, paso han de ser números enteros (simplificando)
  - Se puede solo usar fin, y entonces inicio= 0 y paso=1
  - fin no se llega alcanzar, es decir **range**(3) => 0,1, **y 2**

```
<sin nombre> *
1  tLetrasyNum = 'd','c','b','a',4,3,2,1
2
3  for elemento in tLetrasyNum:
4      print(elemento)
5
```

```
Consola
>>> %Run -c $EDITOR_CONTENT

d
c
b
a
4
3
2
1
```

```
1  for i in range(1,10,2):
2      print(i)
3
```

```
Consola
>>> %Run -c $EDITOR_CONTENT

1
3
5
7
9

>>>
```

```
1  for i in range(3):
2      print(i)
3
```

```
Consola
>>> %Run -c $EDITOR_CONTENT

0
1
2

>>>
```


Profesor voluntario J.C. Santamaria Poza

Analisis: Cambiar el programa de Bromas para usar tuplas y for

https://www.online-python.com/

Creamos una variable ‘chistesTupla’ de tipo tupla con los chistes en cadenas de caracteres (usamos comillas “”” para poder extendernos por varias líneas)

El bloque **AZUL** es el bloque de inicialización

El bloque **ROJO** es el bloque de que se va a repetir por cada chiste de la variable ‘chistesTupla’

P_bromas_for.py

10’

P_bromas_for.py

```
40
41  titulo = 'CHISTES malos CONTADOS por 2 MARCIANOS'
42  print(titulo.capitalize())
43  print(len(titulo) * '=')
44  print()
45
46  # Empieza el bloque de chistes
47  for chiste in chistesTupla:
48      print(chiste)
49      print('ja' * randint(2,5))
50      print('--------')
51      sleep(PAUSAFIN)
52
53  print('Fin de los chistes')
```

Consola

```
          Marciano 1: Son CIEN-tíficos

jajaja
--------

          Marciano 1: Ayer a la abuela le dió un infarto y le rev
          Marciano 2: Latía?
          Marciano 1: La tia no, la abuela, burro, la abuela

jajaja
--------
Fin de los chistes
```

Analisis: Hacer un programa de cuenta adelante con for y formatos f-string (básico) :

https://www.online-python.com/

P_esconditeFor_1_0.py

10’

El bloque **AZUL** es el bloque de inicialización

El bloque **ROJO** es el bloque de que se va a repetir desde 1 hasta MAXCUENTA
:>2d quiere decir formato decimal 2 cifras alineadas derecha

Una linea de finalización que se ejecuta al acabar el bucle

```
from time import sleep # importamos SOLO la funcion sleep
MAXCUENTA = 20
PAUSACUENTA = 1.5 # 1 SEGUNDO Y MEDIO

titulo = 'empieza la cuenta del escondite'
print(titulo.upper()) # Cambia un texto a modo frase
print(len(titulo) * '=') # hace un subrallado de *'s
print()

for c in range(1,MAXCUENTA+1):
    print(F'Cuento {c:>2d}’) # preferido desde Python 3.7
    sleep(PAUSACUENTA)

print('Fin de la cuenta del escondite')
```

Reto CL5: Mostrar lista de primos hasta 150 (dados) alineados a la izq en 3 posiciones enteras

Sugerencias:

1- La lista de numero primos en una tupa la doy en un fichero de Python que usaras para continuar el programa

‘Tupla_primosCL5RETO.py’

2- Usa un bucle **for** para recorrer la variable con los numero primos

3 – Imprime con **print** cada número primo con un formato de

- 3 números decimales
- SIN rellenar con ceros
- Justificando a la derecha

- Ayuda el formato quedaría como ‘print(F'Es primo el numero {primo:>3d}')’