

## 02-Colecciones

December 10, 2020

### 1 Colecciones en Python

Distinguir entre el uso de listas, tuplas, diccionarios

#### 1.1 Listas

Se distinguen por []

```
[1]: lst_x = ['Pedro', 'Miguelina', 'Fernando', 'Josefina']  
lst_x
```

```
[1]: ['Pedro', 'Miguelina', 'Fernando', 'Josefina']
```

```
[2]: lst_y = [25,45,29,15,30,6]  
lst_y
```

```
[2]: [25, 45, 29, 15, 30, 6]
```

```
[4]: lst_z = ['Pedro', 25, 'Miguelina', 56, 'Fernando', 45, 'Josefina', 49.3, False]  
lst_z
```

```
[4]: ['Pedro', 25, 'Miguelina', 56, 'Fernando', 45, 'Josefina', 49.3, False]
```

```
[5]: lst_x[0:2]
```

```
[5]: ['Pedro', 'Miguelina']
```

```
[7]: lst_y[2:5]
```

```
[7]: [29, 15, 30, 6]
```

```
[8]: lst_z[3:890]
```

```
[8]: [56, 'Fernando', 45, 'Josefina', 49.3, False]
```

```
[23]: lst_y = ['a','e','i','o','u',1,2,3,4,5]  
print("1:", lst_y[:2])
```

```
print("2:", lst_y[-3])
print("3:", lst_y[-3:])
print("3:", lst_y[3:5])
```

```
1: ['a', 'e']
2: 3
3: [3, 4, 5]
3: ['o', 'u']
```

```
[10]: # Agregar elementos a la lista
lst_z = lst_z + [89]
lst_z
```

```
[10]: ['Pedro', 25, 'Miguelina', 56, 'Fernando', 45, 'Josefina', 49.3, False, 89]
```

```
[16]: lst_w = [2,15,3]
print("Inicial : ", lst_w)
lst_w = lst_w * 2
print("Final   : ", lst_w)
```

```
Inicial : [2, 15, 3]
Final   : [2, 15, 3, 2, 15, 3]
```

```
[18]: print(len(lst_w))
```

```
6
```

```
[19]: # Eliminación de elementos
del(lst_w[len(lst_w)-2:])
lst_w
```

```
[19]: [2, 15, 3, 2]
```

```
[27]: # Referencias
x = ["a", "b", "c"]
y = x
print("x inicial = ", x)
print("y inicial = ", y)
x[2] = 'd'
print("x final = ", x)
print("y final = ", y)
```

```
x inicial = ['a', 'b', 'c']
y inicial = ['a', 'b', 'c']
x final = ['a', 'b', 'd']
y final = ['a', 'b', 'd']
```

```
[28]: # Ahora se realizan cambios en y
print("x inicial = ", x)
print("y inicial = ", y)
y[1] = 'e'
print("x final = ", x)
print("y final = ", y)
```

```
x inicial = ['a', 'b', 'd']
y inicial = ['a', 'b', 'd']
x final = ['a', 'e', 'd']
y final = ['a', 'e', 'd']
```

```
[29]: x = ["a", "b", "c"]
y = list(x)
print("x inicial = ", x)
print("y inicial = ", y)
y[1] = "f"
print("x final = ", x)
print("y final = ", y)
```

```
x inicial = ['a', 'b', 'c']
y inicial = ['a', 'b', 'c']
x final = ['a', 'b', 'c']
y final = ['a', 'f', 'c']
```

## 1.2 Tuplas

Se distinguen por () y son listas inmutables

```
[30]: tx = (10, 15, 3)
tx
```

```
[30]: (10, 15, 3)
```

```
[31]: ty = ("aire", 90, 'a', True)
ty
```

```
[31]: ('aire', 90, 'a', True)
```

```
[32]: tz = 29,
len(tz)
```

```
[32]: 1
```

```
[35]: ta = ('a', 'e', 'i')
tb = (15,9,3)
ta + tb
```

```
[35]: ('a', 'e', 'i', 15, 9, 3)
```

```
[36]: ta*3
```

```
[36]: ('a', 'e', 'i', 'a', 'e', 'i', 'a', 'e', 'i')
```

```
[37]: tx = (15,12,4)
      ty = (25,15,8)
      tx < ty
```

```
[37]: True
```

### 1.3 Conjuntos

Se denotan por {}

```
[39]: cx = {True, 'a', "aire", 29, 29 < 10, 20 % 2 == 0 }
      cx
```

```
[39]: {29, False, True, 'a', 'aire'}
```

```
[43]: print("1: ", type(cx))
      print("2: ", 29 in cx)
      print("3: ", 29 not in cx)
```

```
1: <class 'set'>
2: True
3: False
```

```
[44]: cx.add(25)
      cx
```

```
[44]: {25, 29, False, True, 'a', 'aire'}
```

```
[46]: cx.update([15,73,'b'])
      cx
```

```
[46]: {15, 25, 29, 73, False, True, 'a', 'aire', 'b'}
```

```
[52]: cx = {'a', "aire", "cocina", 26, 29, 1}
      print("cx antes : ", cx)
      cx.remove('a')
      print("cx despues : ", cx)
```

```
cx antes : {'aire', 'a', 1, 'cocina', 26, 29}
cx despues : {'aire', 1, 'cocina', 26, 29}
```

```
[53]: # Respecto del método remove() la consideración que se debe tener es que si el
      ↪ elemento no existe marca error
      cx = {'a', "aire", "cocina", 26, 29, 1}
      print("cx antes : ", cx)
      cx.discard('b')
      print("cx despues : ", cx)
```

```
cx antes : {'aire', 'a', 1, 'cocina', 26, 29}
cx despues : {'aire', 'a', 1, 'cocina', 26, 29}
```

## 1.4 Diccionarios

Definen una relación de uno a uno entre <clave, valor>

```
[54]: peliculas = {'Dr. Strange': 7, 'Capitan América: civil war': 6,
                  'Avengers: la era de ultron': 8, 'Avengers: end game': 10,
                  ↪ 'Pantera negra': 9}
      for nombre, nota in peliculas.items():
          print(nombre, "tiene nota", nota)
```

```
Dr. Strange tiene nota 7
Capitan América: civil war tiene nota 6
Avengers: la era de ultron tiene nota 8
Avengers: end game tiene nota 10
Pantera negra tiene nota 9
```

```
[56]: for nombre in peliculas:
      print(nombre, 'tiene nota', peliculas[nombre])
```

```
Dr. Strange tiene nota 7
Capitan América: civil war tiene nota 6
Avengers: la era de ultron tiene nota 8
Avengers: end game tiene nota 10
Pantera negra tiene nota 9
```

```
[57]: # Agrega un elemento al diccionario
      peliculas['Capitan América, soldado de invierno'] = 9
      peliculas
```

```
[57]: {'Dr. Strange': 7,
      'Capitan América: civil war': 6,
      'Avengers: la era de ultron': 8,
      'Avengers: end game': 10,
      'Pantera negra': 9,
      'Capitan América, soldado de invierno': 9}
```

```
[58]: "Dr. Strange" in peliculas
```

[58]: True

```
[63]: precios = dict(P15=50000, P73=76000, P3=4000)
      precios
```

[63]: {'P15': 50000, 'P73': 76000, 'P3': 4000}

```
[64]: precios.clear()
      precios
```

[64]: {}

```
[67]: print("1:", peliculas.get("Pantera negra"))
      print("2:", peliculas.keys())
      print("3:", peliculas.values())
```

```
1: 9
2: dict_keys(['Dr. Strange', 'Capitan América: civil war', 'Avengers: la era de
ultron', 'Avengers: end game', 'Pantera negra', 'Capitan América, soldado de
invierno'])
3: dict_values([7, 6, 8, 10, 9, 9])
```

```
[68]: estudiantes = ('HONORATO', 'WANDA', 'BEYONCE', 'LAURA')
      d_estudiantes = dict.fromkeys(estudiantes)
      d_estudiantes
```

[68]: {'HONORATO': None, 'WANDA': None, 'BEYONCE': None, 'LAURA': None}

```
[72]: estudiantes = ('HONORATO', 'WANDA', 'BEYONCE', 'LAURA')
      d_estudiantes = dict.fromkeys(estudiantes, 5)
      d_estudiantes
```

[72]: {'HONORATO': 5, 'WANDA': 5, 'BEYONCE': 5, 'LAURA': 5}

```
[73]: print(d_estudiantes.items())
```

```
dict_items([('HONORATO', 5), ('WANDA', 5), ('BEYONCE', 5), ('LAURA', 5)])
```