

FUNCIONES Y METODOS PARA VARIABLES

TIPO LISTA (LIST) Y TUPLA (TUPLE)

1) append(x)

→ Agrega al final de una lista un elemento descrito por el argumento “x”, donde “x” puede ser cualquier tipo de variable.

EJM 1:

```
>>> variable1 = [1, 2, 3, 4, 5, 6]
```

```
>>> variable1.append(7)
```

```
>>> print (variable1)
```

```
[1, 2, 3, 4, 5, 6, 7]
```

```
>>>
```

EJM 2:

```
>>> variable1 = [1, 2, 3, 4, 5, 6]
```

```
>>> variable1.append("7")
```

```
>>> print (variable1)
```

```
[1, 2, 3, 4, 5, 6, "7"]
```

```
>>>
```

EJM 3:

```
>>> variable1 = [1, 2, 3, 4, 5, 6]
```

```
>>> variable1.append(1.25)
```

```
>>> print (variable1)
```

```
[1, 2, 3, 4, 5, 6, 1.25]
```

```
>>>
```

EJM 4:

```
>>>variable1 = [1, 2, 3, 4, 5, 6]
>>>variable1.append([1, 2, 3, 4, 5])
>>>print (variable1)
[1, 2, 3, 4, 5, 6, [1, 2, 3, 4, 5]]
>>>
```

2) extend(x)

→ Agrega al final de una lista todos los elementos de una variable “x” la cual tiene que ser una variable ITERABLE (string, list, tuple, set o dic).

EJM 1:

```
>>> variable1 = [1, 2, 3]
>>>variable1.extend("HOLA")
>>>print (variable1)
[1, 2, 3,"H","O","L","A"]
>>>
```

EJM 2:

```
>>> variable1 = [1, 2, 3]
>>>variable1.extend([4, 5, 6])
>>>print (variable1)
[1, 2, 3, 4, 5, 6]
>>>
```

EJM 3:

```
>>> variable1 = [1, 2, 3]
>>> variable1.extend("HOLA")
>>> print (variable1)
[1, 2, 3,"H","O","L","A"]
>>>
```

EJM 4:

```
>>> variable1 = [1, 2, 3]
>>> variable2 = (1, 2, 3)
>>> variable1.extend(variable2)
>>> print (variable1)
[1, 2, 3, 1, 2, 3]
>>>
```

3) insert(i,x)

→ Agrega un elemento descrito por la variable "x" en una posición específica descrita por la variable "i", esta última variable debido a que denota posición en una lista, debe de ser siempre una variable del tipo "int".

EJM 1:

```
>>> variable1 = [1, 2, 3]
>>> variable1.insert(1, 20)
>>> print (variable1)
[1, 20, 2, 3]
>>>
```

EJM 2:

```
>>> variable1 = [1, 2, 3]
>>> variable1.insert(3, 20)
>>> print (variable1)
[1, 2, 3, 20]
>>>
```

EJM 3:

```
>>> variable1 = [1, 2, 3]
>>> variable1.insert(10, 20)
>>> print (variable1)
[1, 2, 3, 20]
>>>
```

EJM 4:

```
>>> variable1 = [1, 2, 3]
>>> variable1.insert(-1, 20)
>>> print (variable1)
[1, 2, 20, 3]
>>>
```

EJM 5:

```
>>> variable1 = [1, 2, 3]
>>> variable1.insert(-4, 20)
>>> print (variable1)
[20, 1, 2, 3]
>>>
```

4) count(x)

→ Devuelve el número de veces que se repite la variable “x” es una lista o tupla específica.

EJM 1:

```
>>> variable1 = [1, 2, 3]
```

```
>>>variable1.count(1)
```

```
>>>print (variable1)
```

```
1
```

```
>>>
```

EJM 2:

```
>>> variable1 = (1, 2, 3)
```

```
>>>variable1.count(4)
```

```
>>>print (variable1)
```

```
0
```

```
>>>
```

EJM 3:

```
>>> variable1 = [1, 2, 3, 1, 2, 1]
```

```
>>>variable1.count(1)
```

```
>>>print (variable1)
```

```
3
```

```
>>>
```

5) index(x)

→ Devuelve la posición de un elemento en una lista o tupla específica, si el elemento no se encuentra en la lista o tupla, esta nos devuelve un error.

EJM 1:

```
>>> variable1 = [1, "a", 3,"b"]
```

```
>>>variable1.index(1)
```

```
0
```

```
>>>
```

EJM 2:

```
>>> variable1 = (1, "a", 3,"b")
```

```
>>>variable1.index("b")
```

```
3
```

```
>>>
```

EJM 3:

```
>>> variable1 = [1, "a", 3,"b"]
```

```
>>>variable1.index("c")
```

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

ValueError: 'c' in not in list

```
>>>
```

6) len(x)

→ Permite saber el número de elementos que posee una lista o tupla colocando el nombre de la variable en el lugar de "x".

EJM 1:

```
>>> variable1 = ["a","b","c","d"]
```

```
>>>len(variable1)
```

```
4
```

```
>>>
```

EJM 2:

```
>>> variable1 = (1, 2, 3)
```

```
>>>len(variable1)
```

```
3
```

```
>>>
```

EJM 3:

```
>>> variable1 = [ ]
```

```
>>>len(variable1)
```

```
0
```

```
>>>
```

7) del x[i]

→ Elimina el elemento de posición específica “i” de una lista “x”. Si intentamos eliminar una posición en la que no existe variable, nos arrojará un error.

EJM 1:

```
>>> variable1 = [1, 2, 3]
```

```
>>>del variable1[0]
```

```
>>>print (variable1)
```

```
[2, 3]
```

```
>>>
```

EJM 2:

```
>>> variable1 = [1, 2, 3]
```

```
>>>del variable1[2]
```

```
>>>print (variable1)
```

```
[1, 2]
```

```
>>>
```

8) list(x)

→ Convierte una variable “x” tipo string a una lista cuyos elementos individuales son cada carácter del string “x”.

EJM 1:

```
>>> variable1 = "hola mundo"
```

```
>>> variable2 = list(variable1)
```

```
>>>print (variable2)
```

```
['h','o','l','a',' ','m','u','n','d','o']
```

```
>>>
```


9) "".join(x)

→ Convierte una variable "x" tipo Lista a una variable tipo String en la cual todos los elementos de la lista se convierten en caracteres del String. Todos los elementos de la lista deben de ser tipo String para que esta función tenga efecto, caso contrario nos arrojará un error.

EJM 1:

```
>>> variable1 = ["a", "bc", "def"]
```

```
>>> variable2 = "".join(variable1)
```

```
>>>print (variable2)
```

```
'abcdef'
```

```
>>>
```

10) x.split(y)

→ Separa los caracteres de una variable "x" tipo String por un carácter común "y" que esta contenga y los pasa a una lista.

EJM 1:

```
>>> variable1 = "12,15,17,15,12,13"
```

```
>>> variable2 = variable1.split(",")
```

```
>>>print (variable2)
```

```
[12, 15, 17, 12, 13]
```

```
>>>
```

EJM 2:

```
>>> variable1 = "12.19/18.3/22.22/20.4/23.8/22.35"
```

```
>>> variable2 = variable1.split("/")
```

```
>>> print (variable2)
```

```
[12.19, 18.3, 22.22, 20.4, 23.8, 22.35]
```

```
>>>
```

11) sorted(x)

→ Ordena de manera ascendente una lista "x", todos sus elementos deben de ser del tipo numérico real (int o float).

EJM 1:

```
>>> variable1 = [6, 3, 8, 2, 15, -1, 9, 20, 1]
```

```
>>> variable2 = sorted(variable1)
```

```
>>> print (variable2)
```

```
[-1, 1, 2, 3, 6, 8, 9, 15, 20]
```

```
>>>
```

12) x.reverse()

→ Invierte el orden de la lista "x" en sí misma, es decir, no requiere ser guardado en otra variable para guardar el cambio pues, lo realiza sobre la misma lista.

EJM 1:

```
>>> variable1 = [6, 3, 8, 2, 15, -1, 9, 20, 1]
```

```
>>> variable1.reverse()
```

```
>>> print (variable1)
```

```
[1, 20, 9, -1, 15, 2, 8, 3, 6]
```

```
>>>
```

13) x.pop()

→ Elimina el último elemento de una lista "x", independientemente de su posición actual en la que se encuentre este ultimo elemento.

EJM 1:

```
>>> variable1 = ["a", "b", "c"]
>>> variable1.pop()
>>> print (variable1)
["a", "b"]
>>>
```

14) set(x)

→ Elimina todos los elementos repetidos de una lista "x".

EJM 1:

```
>>> variable1 = ["a", "a", 1, 2, 1, "b"]
>>> set(variable1)
>>> print (variable1)
[1, "a", 2, "b"]
>>>
```

13) list(map(i,x))

→ Convierte todos los elementos de una lista "x", tal que si todos eran variables tipo String y si al argumento "i" es explícitamente "int" convierte todos sus elementos a tipo Int (se debe de tomar en cuenta que todos estos String deben de ser caracteres numéricos). Lo mismo ocurre con el caso contrario, es decir de Int a String.

EJM 1:

```
>>> variable1 = ["1", "10", "100", "1000"]
```

```
>>> variable2 = list(map(int, variable1))
```

```
>>> print (variable2)
```

```
[1, 10, 100, 1000]
```

```
>>>
```

EJM 2:

```
>>> variable1 = [5, 25, 125, 625]
```

```
>>> variable2 = list(map(string, variable1))
```

```
>>> print (variable2)
```

```
["5", "25", "125", "625"]
```

```
>>>
```

EJM 3:

```
>>> variable1 = ["5.625", "25.55", "125.1", "625.25"]
```

```
>>> variable2 = list(map(float, variable1))
```

```
>>> print (variable2)
```

```
[5.625, 25.55, 125.1, 625.25]
```

```
>>>
```