

```
Python 3.7.5 Shell
File Edit Shell Debug Options Window Help
Python 3.7.5 (default, Apr 19 2020, 20:18:17)
[GCC 9.2.1 20191008] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>> ['a','b','c'].index("c")
2
>>> ('a','b','c').index("c")
2
>>> [3,2,1].pop(2)
1
>>> (3,2,1).pop(2)
Traceback (most recent call last):
  File "<pyshell#3>", line 1, in <module>
    (3,2,1).pop(2)
AttributeError: 'tuple' object has no attribute 'pop'
>>> dic1={'prof1':'g_ingles','prof2':'g_frances'}
>>> dic1.update({'prof3':'g_aleman'})
>>> print(dic1.keys())
dict_keys(['prof1', 'prof2', 'prof3'])
>>> {1,2,3}&{2,4,6}
{2}
>>> {1,2,3}|{2,4,6}
{1, 2, 3, 4, 6}
>>> {1,2,3}&{1}
{1}
>>> {1,2,3}-{2,4,6}
{1, 3}
>>> {1,2,3}^{2,4,6}
{1, 3, 4, 6}
>>> 3 in {1,2,3}^{3,4,5}
False
>>> 3 not in (1,2,3)
False
>>> |
```

1. >>> ['a','b','c'].index("c")

El método index devuelve la posición de la primera ocurrencia que coincida con el valor especificado. Al ser el valor especificado "c", devuelve la posición 2, que es donde se encuentra la primera 'c' de lista.

2. >>> ('a','b','c').index("c")

Mismo caso que el 1, pero aplicado a una tupla en lugar de a una lista.

El método index devuelve la posición de la primera ocurrencia que coincida con el valor especificado. Al ser el valor especificado "c", devuelve la posición 2, que es donde se encuentra la primera 'c' de la tupla.

3. >>> [3,2,1].pop(2)

El método pop retorna y elimina el elemento de la lista de la posición especificada. En la posición 2 de la lista se encuentra el valor 1, por ello retorna el 1, y a su vez es eliminado, tal y como se muestra en la siguiente imagen:

```
1 array=[3,2,1]
2 array.pop(2)
3 for i in range(len(array)):
4     print(array[i])
```

OUTPUT TERMINAL DEBUG CONSOLE PROBLEMS

```
juanjo@Juanjo:~/Documentos/curso_uned_python/tema3-tuplas$ python pruebas3.py
3
2
```

4. >>> (3,2,1).pop(2)

Al tratarse de una tupla no dispone del método pop ya que por definición, una tupla agrupa elementos de forma inmutable (y .pop() permite modificar los elementos).

```
5. >>> dic1={'prof1':'g_ingles','prof2':'g_frances'}
>>> dic1.update({'prof3':'g_aleman'})
>>> print(dic1.keys())
```

El resultado es dict_keys(['prof1', 'prof2', 'prof3']) ya que estamos actualizando el diccionario con un nuevo item (clave:valor) y estamos haciendo el print de las claves de dicho diccionario.

6. >>> {1,2,3}&{2,4,6}

Retorna la intersección, es decir el/los valores que se encuentren en ambos conjuntos. En este caso es {2}.

7. >>> {1,2,3}|{2,4,6}

Retorna la unión, es decir, los valores de ambos conjuntos sin repetirse. En este caso {1,2,3,4,6}. El 2 se encuentra en ambos conjuntos, pero sólo retorna uno (sin repetición).

8. >>> {1,2,3}&{1}

Mismo caso que el ejemplo 6. Retorna la intersección. En este caso el {1}.

9. >>> {1,2,3}-{2,4,6}

Retorna la diferencia, es decir, los valores que están en el primer conjunto que no se encuentran en el segundo. En este caso el 2 se encuentra en ambos, por ello devuelve {1,3}.

10. `>>> {1,2,3}^{2,4,6}`

Retorna los valores que están en el primero o en el segundo, pero no en ambos. El único valor que está en ambos es el 2, por ello devuelve todo lo demás $\{1,3,4,6\}$.

11. `>>> 3 in {1,2,3}^{3,4,5}`

Devuelve un booleano atendiendo a la condición de si el valor 3 se encuentra en $\{1,2,3\}^{3,4,5}$. Como $\{1,2,3\}^{3,4,5}$ retorna $\{1,2,4,5\}$ y el 3 no se encuentra entre los valores, retorna false.

12. `>>> 3 not in (1,2,3)`

Retorna un booleano. En este caso false, ya que 3 sí se encuentra en la tupla.