

FUNDAMENTOS DE PROGRAMACIÓN EN PYTHON

[CC-BY-NC-SA] Leonardo Marco

PPP-PE-P4. Tipos de datos mutables e inmutables

ALUMNO: Alejandro Bermúdez

1) Crea una variable `ruta` con el valor `c:\\datos\\`.

- Muestra la identidad del objeto almacenado en la variable.
- Modifica la variable `ruta` para añadirle al final la cadena `fichero.txt`.
- Muestra de nuevo la identidad de la variable. ¿Es la misma? ¿Por qué?
- Crea una nueva variable `ruta2` que sea igual a la variable `ruta`.
- Muestra la identidad de ambas variables ¿son la misma?
- Modifica la variable `ruta2` para añadirle al final la cadena `.old`.
- Muestra el contenido de ambas variables ¿son el mismo? ¿por qué?
- Muestra la identidad de ambas variables ¿son la misma?

```
>>> ruta = "c:\\datos\\"

>>> print(id(ruta))

140705306020080

>>> ruta += "fichero.txt"

>>> print(id(ruta))

140705306019888

>>> ruta2 = ruta

>>> print(id(ruta))

140705306019888

>>> print(id(ruta2))

140705306019888
```

```
>>> ruta2 += ".old"

>>> print(ruta)

c:\datos\fichero.txt

>>> print(ruta2)

c:\datos\fichero.txt.old

>>> print(id(ruta))

140705306019888

>>> print(id(ruta2))

140705306021136
```

La identidad de la variable cambia al modificarla porque las cadenas son inmutables

Cuando se igualan las dos variables apuntan al mismo objeto de memoria

Cuando la segunda ruta se modifica se crea un nuevo objeto entonces las identidades son diferentes otra vez.

2) Crea una variable `lista` con una lista de 5 valores numéricos.

- Muestra la identidad del objeto almacenado en la variable.
- Modifica el valor del elemento de la posición [0].
- Muestra de nuevo la identidad de la variable. ¿Es la misma? ¿Por qué?
- Crea una nueva variable `lista2` que sea igual a la variable `lista`.
- Muestra la identidad de ambas variables ¿son la misma?
- Modifica la variable `lista2` para añadirle al final un nuevo elemento.
- Muestra el contenido de ambas variables ¿son el mismo? ¿por qué?
- Muestra la identidad de ambas variables ¿son la misma?

```
>>> lista = [1, 2, 3, 4, 5]

>>> print(id(lista))

2089348438720
```

```
>>> lista[0] = 99  
  
>>> print(id(lista))  
  
2089348438720  
  
>>> lista2 = lista  
  
>>> print(id(lista))  
  
2089348438720  
  
>>> print(id(lista2))  
  
2089348438720  
  
>>> lista2.append(6)  
  
>>> print(lista)  
  
[99, 2, 3, 4, 5, 6]  
  
>>> print(lista2)  
  
[99, 2, 3, 4, 5, 6]  
  
>>> print(id(lista))  
  
2089348438720  
  
>>> print(id(lista2))  
  
2089348438720
```

La identidad de la variable no cambia al modificarla porque las listas son mutables

Cuando se igualan las dos variables apuntan al mismo objeto de memoria

Cuando la segunda ruta se modifica, lo hace también la otra, ocupan la misma referencia en memoria.