

# Tipo cadena

En Python es muy sencillo manejar "palabras", es decir, cadenas de caracteres.

```
In [1]: a = 'Hola'
```

```
In [2]: b = 'Adios'
print a, b
```

Hola Adios

```
In [3]: #Podemos concatenar cadenas
c = a + b
c
```

Out[3]: 'HolaAdios'

**Cuidado:** No hay que confundir las cadenas con otros tipos de datos, sobre todo numéricos.

```
In [4]: a = '12'
b = 12
print a+a, b+b
```

1212 24

```
In [6]: #Las cadenas se pueden multiplicar
print a*10
```

12121212121212121212

```
In [7]: #Las cadenas se pueden comparar
a = 'acacia'
b = 'burro'
c = 'acaso'
a == b
```

Out[7]: False

```
In [8]: a < b
```

Out[8]: True

```
In [9]: a < c
```

Out[9]: True

## Algunas funciones útiles

```
In [12]: a = 123
b = int(a) + 1
b
```

Out[12]: 124

```
In [13]: c = '3.14'
```

```
d = float(c)
d
```

Out[13]: 3.14

```
In [14]: e = str(d)
         c == e
```

Out[14]: True

```
In [15]: m = 'a'
         ord(m)
```

Out[15]: 97

```
In [16]: chr(ord(m)+1)
```

Out[16]: 'b'

```
In [17]: a = 'Hola'
         a.upper()
```

Out[17]: 'HOLA'

```
In [18]: a.lower()
```

Out[18]: 'hola'

## Caracteres especiales

Además de las "letras" ordinarias, hay caracteres especiales. Hablaremos de dos:

- Salto de línea \n
- Tabulador \t

```
In [23]: s = 'Hace un bonito\n día'
```

```
In [24]: print s
```

```
Hace un bonito
día
```

```
In [25]: s = 'Hace un bonito\t día'
         print s
```

```
Hace un bonito   día
```

## Más operaciones con cadenas.

Podemos calcular su longitud con len

```
In [26]: a = 'Hola'
         len(a)
```

Out[26]: 4

```
In [27]: b = '' #cadena vacía
```

```
len(b)
```

Out[27]: 0

```
In [30]: #Podemos acceder a cada caracter con su índice
cad2 = 'En un lugar de la Mancha'
print cad2
for i in range(len(cad2)):
    print cad2[i], '- ',
```

```
En un lugar de la Mancha
E - n -   - u - n -   - l - u - g - a - r -   - d - e -   - l - a -   - M - a - n -
c - h - a -
```

```
In [32]: for j in range(len(cad2)):
        print cad2[-(j+1)],
```

```
a h c n a M   a l   e d   r a g u l   n u   n E
```

```
In [33]: #podemos manejar SUBCADENAS
print cad2[3:5]
print cad2[:10]
print cad2[-6:-1]
print cad2[:]
```

```
un
En un luga
Manch
En un lugar de la Mancha
```

## Ejemplos

### Calcula la letra del DNI

```
In [37]: def dni(num):
        '''
        Función que calcula la letra del DNI
        a partir del número
        '''
        tabla = 'TRWAGMYFPDXBNJZSQVHLCKE'
        posicion = num % 23
        return tabla[posicion]
```

```
In [38]: dni(1111111)
```

Out[38]: 'G'

### Espacios en blanco

```
In [39]: def blancos(cadena):
        '''
        Lee una cadena y muestra el número de espacios
        en blanco
        '''
        contador = 0
```

```
for i in range(len(cadena)):
    if cadena[i] == ' ':
        contador+=1
return contador
```

```
In [40]: s = 'En un lugar de la Mancha'
        blancos(s)
```

```
Out[40]: 5
```

## Cadenas alfabéticas

```
In [41]: '''
Una palabra es "alfabética" si todas
sus letras están ordenadas alfabéticamente.
'''

def alfabetica(cadena):
    alfa = True
    i = 0
    while (i<len(cadena)-1) and alfa:
        alfa = cadena[i] < cadena[i+1]
        i+=1
    return alfa
```

```
In [42]: a = 'abcdexy'
        alfabetica(a)
```

```
Out[42]: True
```

```
In [43]: b = 'abcxzy'
        alfabetica(b)
```

```
Out[43]: False
```

## Palíndromos

```
In [44]: '''
Determina si una palabra es un palíndromo
'''

def palindromo(cadena):
    palin = True
    s = len(cadena)
    i = 0
    while (i<s/2) and palin:
        palin = cadena[i] == cadena[s-i-1]
        i+=1
    return palin
```

```
In [45]: palindromo('abcba')
```

```
Out[45]: True
```

In [46]: palindromo('H**o**l**a**')

Out[46]: False

In [ ]: