

Tipo cadena

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

```
In [30]: a = 'Hola'
```

```
In [31]: b = 'Adios'
print a, b
```

```
Hola Adios
```

```
In [32]: #Podemos concatenar cadenas
c = a + b
c
```

```
Out[32]: 'HolaAdios'
```

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

```
In [33]: a = '12'
b = 12
print a+a, b+b
```

```
1212 24
```

```
In [34]: #Las cadenas se pueden multiplicar
print a*10
```

```
12121212121212121212
```

```
In [35]: #Las cadenas se pueden comparar
a = 'acacia'
b = 'burro'
c = 'acaso'
a == b
```

```
Out[35]: False
```

```
In [36]: a < b
```

```
Out[36]: True
```

```
In [37]: a < c
```

```
Out[37]: True
```

Algunas funciones útiles

```
In [38]: a = '123'  
b = int(a) + 1  
b
```

```
Out[38]: 124
```

```
In [39]: c = '3.14'  
d = float(c)  
d
```

```
Out[39]: 3.14
```

```
In [40]: e = str(d)  
c == e
```

```
Out[40]: True
```

Con las funciones **int()**, **float()** y **str()** podemos intercambiar información sobre datos numéricos y cadenas de caracteres.

Existen funciones especiales para las cadenas de longitud 1 (es decir, caracteres individuales)

```
In [41]: m = 'a'  
ord(m)
```

```
Out[41]: 97
```

```
In [42]: chr(ord(m)+1)
```

```
Out[42]: 'b'
```

```
In [43]: a = 'Hola'  
a.upper()
```

```
Out[43]: 'HOLA'
```

```
In [44]: a.lower()
```

```
Out[44]: 'hola'
```

Caracteres especiales

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

- Salto de línea `\n`
- Tabulador `\t`

```
In [45]: s = 'Hace un bonito\n día'
```

```
In [46]: print s
```

```
Hace un bonito
 día
```

```
In [47]: s = 'Hace un bonito\t día'
print s
```

```
Hace un bonito   día
```

Más operaciones con cadenas.

Podemos calcular su longitud con `len`

```
In [48]: a = 'Hola'
len(a)
```

```
Out[48]: 4
```

```
In [49]: b = '' #cadena vacía
len(b)
```

```
Out[49]: 0
```

```
In [50]: #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 [51]: 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 [52]: #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 [53]: def dni(num):
        '''
        Función que calcula la letra del DNI
        a partir del número
        '''
        tabla = 'TRWAGMYFPDXBNJZSQVHLCKE'
        posicion = num % 23
        return tabla[posicion]
```

```
In [54]: dni(1111111)
```

```
Out[54]: 'G'
```

Espacios en blanco

```
In [55]: 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 [56]: s = 'En un lugar de la Mancha'  
blancos(s)
```

```
Out[56]: 5
```

Cadenas alfabéticas

```
In [57]: '''  
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 [58]: a = 'abcdexy'  
alfabetica(a)
```

```
Out[58]: True
```

```
In [59]: b = 'abcxzy'
alfabetica(b)
```

```
Out[59]: False
```

Palíndromos

```
In [60]: '''
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 [61]: palindromo('abcba')
```

```
Out[61]: True
```

```
In [62]: palindromo('Hola')
```

```
Out[62]: False
```

Cadenas y listas

Podemos transformar la cadenas en listas y viceversa.

```
In [63]: #podemos pasar de una cadena a una lista con split()
cadena = 'En un lugar de la mancha'
print cadena.split()
```

```
['En', 'un', 'lugar', 'de', 'la', 'mancha']
```

```
In [64]: cadena = 'Ana, Juan Carlos, Pedro, Sonia'
print cadena.split(',')
```

```
['Ana', ' Juan Carlos', ' Pedro', ' Sonia']
```

```
In [65]: #Realizamos el proceso inverso con join()
lista = ['Ana', 'Juan Carlos', 'Pedro', 'Sonia']
print ','.join(lista)
```

```
Ana,Juan Carlos,Pedro,Sonia
```

```
In [66]: #Con list podemos pasar directamente de cadenas a listas
lista2 = list(cadena)
print lista2
```

```
['A', 'n', 'a', ',', ' ', 'J', 'u', 'a', 'n', ' ', 'C', 'a', 'r', 'l', 'o', 's', ' ', 'P', 'e', 'd', 'r', 'o', ' ', 'S', 'o', 'n', 'i', 'a']
```

```
In [67]: #podemos eliminar los espacios blancos en los extremos con strip()
cadena2 = ' una bonita lista '
lista3 = cadena2.strip().split()
print lista3
```

```
['una', 'bonita', 'lista']
```

```
In [68]: #hay que tener cuidado con los espacios
cadena2 = ' una , bonita , lista '
lista3 = cadena2.strip().split(',')
print lista3
```

```
['una ', ' bonita ', ' lista']
```

```
In [69]: for i in range(len(lista3)):
    lista3[i] = lista3[i].strip()
print lista3
```

```
['una', 'bonita', 'lista']
```

In [69]: