

# Cadenas en Python



# Instrucciones Condicionales

- ✓ Los **cadenas** en Python, es un tipo de dato que sirve para almacenar textos.
- ✓ Internamente es una **lista de caracteres** que se representan como una secuencia de caracteres encerrados entre comillas simples o dobles.  
Ejemplo: ("Hola mundo" u 'Hola mundo').



# Un tipo de datos compuesto

- ✓ Las cadenas son cualitativamente diferentes de los otros cuatro tipos de datos, porque están hechas de piezas más pequeñas — los **caracteres**.
- ✓ Dependiendo de lo que hagamos, podemos tratar un tipo de datos compuesto como una única cosa o podemos acceder a sus partes.



# Como acceder a una parte de la cadena

La forma de representarlo es:

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| H | O | L | A |   | M | U | N | D | O |

Para acceder al caracter de la posición 0 : saludo[0],

Para acceder al caracter de la posición 1 : saludo[1],  
sucesivamente.



# Como acceder a una parte de la cadena

- ✓ El operador corchete selecciona sólo un carácter de una cadena:

```
File Edit Format Run Options Windows Help
saludo = 'hola'
letra = saludo[1]
print letra
```

La expresión `saludo[1]` selecciona el carácter número 1 de saludo:

```
File Edit Shell Debug Options Windows Help
Python 2.7.5 (default, May 15 2013, 22:43:36) [P
32
Type "copyright", "credits" or "license()" for r
>>> ===== RESTART =====
>>>
o
>>>
```



# Como repetir una cadena

Los strings, tienen operaciones y funciones nativas que permiten operar con ellas:

- ✓ Repetición: `<string> * <int>`, al igual que el operador de repetición de las listas, repite una cadena varias veces.

Ejemplo `print 'hola ' * 5`

```
File Edit Shell Debug Options Windows T
Python 2.7.5 (default, May 15 2013,
32
Type "copyright", "credits" or "lic
>>> =====
>>>
hola hola hola hola hola
>>>
```



# Como concatenar una cadena

- ✓ Concatenación: <string> + <string>, al igual que el operador concatenación de las listas, une dos cadenas en una sola.

Ejemplo : `print 'Buenos ' + 'Días'`

```
File Edit Shell Debug Options Windows |
Python 2.7.5 (default, May 15 2013,
32
Type "copyright", "credits" or "lic
>>> =====
>>>
Buenos Dias
>>> |
```



# Largo de una cadena

- ✓ Largo: `len(<string>)`, devuelve la cantidad de elementos de una cadena.

```
File Edit Format Run Options Windows Help
import string
frase=raw_input ("Ingrese una palabra : ")
largo=len(frase)
print "Largo es : ", largo
```

```
File Edit Shell Debug Options Windows Help
Python 2.7.5 (default, May 15 2013, 22:4
32
Type "copyright", "credits" or "license
>>> ===== RE
>>>
Ingrese una palabra : Buenos Dias
Largo es : 11
>>>
```





# Recorrido instrucción While

File Edit Format Run Options Windows Help

```
indice = 0
cadena=raw_input('Ingrese palabra :|')
while indice < len(cadena):
    letra = cadena[indice]
    print letra
    indice += 1
```

File Edit Shell Debug Options Windows Help

Python 2.7.5 (default, May 15 2013, 22:43:36) [MS  
32

Type "copyright", "credits" or "license()" for mo

>>> ===== RESTART =====

>>>

Ingrese palabra :Buenos Dias

B

u

e

n

o

s

D

i

a

s

>>> |



# Recorrido instrucción For

File Edit Format Run Options Windows Help

```
indice = 0
cadena=raw_input('Ingrese palabra :')
for indice in cadena:
    print indice
```

File Edit Shell Debug Options Windows Help

```
Python 2.7.5 (default, May 15 2013, 22:43:30)
32
Type "copyright", "credits" or "license()" :
>>> ===== RESTART
>>>
Ingrese palabra :hola
h
o
l
a
>>> |
```



# Concatenar y la instrucción For

File Edit Format Run Options Windows Help

```
prefijos = "HOLA"  
sufijo = "ack"  
for letra in prefijos:  
    print letra + sufijo
```

|

File Edit Shell Debug Options Windows Help

Python 2.7.5 (default, May 15 2013, 22:43:36) [MSC v.1500  
32

Type "copyright", "credits" or "license()" for more infor

>>> ===== RESTART =====

>>>

Hack

Oack

Lack

Aack

>>> |



# Porciones de cadenas

La selección de una porción es similar a la selección de un carácter:

```
File Edit Format Run Options Window File Edit Shell Debug Options Windows Help
s='Hola Buenos Días'
print s[0:4]
print s[5:11]
print s[12:15]

Python 2.7.5 (default, May 15 2013, 22:43:36) [P
32
Type "copyright", "credits" or "license()" for n
>>> ===== RESTART ===
>>>
Hola
Buenos
Día
>>>
```

El operador `[n:m]` devuelve la parte de la cadena desde el "n-ésimo" carácter hasta el "m-ésimo",



# Porciones de cadenas

Si omite el primer índice (antes de los dos puntos).

File Edit Format Run Options Windows

```
palabra= 'Hola'|  
print palabra[:3]
```

File Edit Shell Debug Options Windows Help

```
Python 2.7.5 (default, May 15 2013, 22:43:36) [I  
32  
Type "copyright", "credits" or "license()" for :  
>>> ===== RESTART =====  
>>>  
Hol  
>>> |
```



# Porciones de cadenas

Si omite el segundo índice.

```
File Edit Format Run Options Windows Help
```

```
palabra= 'Hola'  
print palabra[3:]  
|
```

```
File Edit Shell Debug Options Windows Help
```

```
Python 2.7.5 (default, May 15 2013, 22:4  
32  
Type "copyright", "credits" or "license(  
>>> ===== RES:  
>>>  
a  
>>>
```



# Comparación de cadenas

## Preguntar por un caracter

```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-
palabra=raw_input('Ingrese una palabra : ')
if palabra == "hola":
    print "Sí es hola :"
else:
    print "No es hola :"
```



# El operador in

El operador in prueba si una cadena es una subcadena de otra:

```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-
if 'z' in 'manzana':
    print "SI"
else:
    print "NO"
```





# Ejemplos

Combinando el operador in con la concatenación de cadenas usando +

Función que elimina todas las vocales de una cadena:

```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-
def elimina_vocales(s):
    vocales = "aeiouAEIOU"
    s_sin_vocales = ""
    for letra in s:
        if letra not in vocales:
            s_sin_vocales += letra
    return s_sin_vocales
```



# Ejemplos

Función que cuenta el número de veces que aparece la letra a en una cadena

```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-
fruta = "platano"
cuenta = 0
for carac in fruta:
    if carac == 'a':
        cuenta += 1
print cuenta
```



Encuentra las posiciones de un carácter en una cadena:

```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-
def encuentra2(cadena, carac, inicio):
    indice = inicio
    while indice < len(cadena):
        if cadena[indice] == carac:
            return indice
        indice += 1
    return -1
```



# Biblioteca string

---

Esta biblioteca define diversas variables y funciones útiles para manipular cadenas.

Se usa, haciendo: **import string**



# Algunas funciones

- ✓ `atof(s)` convierte la cadena `s` en un flotante.
- ✓ `atoi(s,[b])` convierte la cadena `s` en un entero (en base `b` si se especifica).
- ✓ `atol(s,[b])` convierte la cadena `s` en un entero largo (en base `b` si se especifica).
- ✓ `lower(s)` pasa `s` a minúsculas.
- ✓ `join(l[,p])` devuelve la cadena que se obtiene al unir los elementos de la lista `l` separados por blancos o por `p` si se especifica.
- ✓ `split(s[,p])` devuelve la lista que se obtiene al separar la cadena `s` por los blancos o por `p` si



# Las variables

digits '0123456789'.

hexdigits '0123456789abcdefABCDEF'.

letters Letras minúsculas y mayúsculas.

lowercase Letras minúsculas.

octdigits '01234567'.

uppercase Letras mayúsculas.

whitespace Blancos



# Ejemplo

```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-
import string
print string.letters
```

Muestra :

```
Python 2.7.5 (default, May 15 2013, 22:43:36) [MSC v.1500 32 bit (Intel)] on win
32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzŠẂšœžŸ°µ°ÀÁÂÃÄÅÆÇÈÉÊËÌÍÎÏÐ
ÑÒÓÔÕÖØÙÚÛÜÝÞßàáâãäåæçèéêëìíîïðñòóôõöøùúûüýþ
>>> |
```

