

Funciones

Python

Funciones

- Hemos estado funciones como `print()`, `input()`, `range()`, `len()`, etc.

Function Header:

Keyword

def	sayHello	()	:
print ("Hello")			

El nombre de la función puede tener letras [A-Z][a-z] e incluido el “_”. También puede tener dígitos [0-9], pero, el nombre de la función

No puede comenzar con número.

La sentencia de retorno

- Esta función retorna un valor. Para eso usamos la palabra **return**.

Function Header has
two parameters

Function Body has
a **return** statement

```
def getSum ( a , b ) :
```

```
    mysum = a + b
```

```
    return mysum
```

**Funciones que python nos
brinda**

divmod()

```
1  print(13%3)
2  print(divmod(13, 3))
3
4
5  q, r = divmod(13, 3)
6  print("Cociente:", q)
7  print("Residuo:", r)
```

reversed()

base6 >  reversed.py > ...

```
1  for a in reversed([1,2,3,4,5]):  
2      print(a)  
3  
4  for b in reversed("neib"):  
5      print(b)
```

round()


Utiliza un redondeo así: lo que significa que los números que son exactamente la mitad (que terminan en .5) se redondean al número **par** más cercano.

```
> round.py > ...
1 print('value\tfloor\tceiling\teven-rounded:')
2
3 for tentimes in range(30, 47):
4     c = tentimes/10
5     print(c, '\t', round(c-0.49), '\t', round(c+0.49), '\t', round(c))
6
7
8 print(round(5.5))
```

hex() y bin()

```
> radix.py > ...
1 sourceNum = int(input("Ingrese un número en decimal: "))
2 targetRadix = int(input("Ingrese la base a convertir 2 o 16: "))
3
4 if (targetRadix==2):
5     print(bin(sourceNum))
6 else:
7     if(targetRadix==16):
8         print(hex(sourceNum))
9     else:
0         print("Base no soportada")
```


sorting()

class6 >  sorting.py > ...

```
1  nums = [4, 2, 5, 4]
2  print("Original: ", nums)
3  print("Sorted: ", sorted(nums))
4
5  alphas = 'zigZAG'
6  print("Original: ", alphas)
7  print("Sorted: ", sorted(alphas))
```

Funciones para manejar Strings



Nota: Todas retornan un nuevo string sin cambiar el original

Is...

is_functions.py ×

clase7 > is_functions.py

```
1 print("a5".isalnum())
2 print("abc".isalpha())
3 print("5".isdigit())
4 print("abc".islower())
5 print("\t".isspace())
6 print("Welcome to Educative".istitle())
7 print("EDUCATIVE".isupper())
```

Funciones relacionadas al cambio de case

is_functions.py

case_change.py ×

clase7 > case_change.py

```
1 print("educative".capitalize())  
2 print("EDUCATIVE".lower())  
3 print("eDUCATIVE".swapcase())  
4 print("welcome to educative".title())  
5 print("educative".upper())
```

Funciones para buscar ocurrencias en String

is_functions.py

case_change.py

ocurrencias.py ×

clase7 > ocurrencias.py

```
1 print("educative".endswith('ve'))
2 print("educative".find('e'))
3 print("educative".rfind('e'))
4 print("educative".count('e'))
5 print("educative".replace('e', '..E..'))
6 print("educative".startswith('edu'))
7
```

Funciones para Listas

```
1 a = [1,2,3]
2 b = [8,9]
3 print("List a:",a)
4 print("List b:",b)
5
6 print("\nAppending 4 in list a")
7 a.append(4) # Appending 4 in list a
8 print("List a:",a)
9
10 print("\nAppending complete list b as sublist in a ")
11 a.append(b) # Appending complete list b as sublist in a
12 print("List a:",a)
13 print("List b:",b)
14
15 print("\nCreating a new copy of list b in list c")
16 c = b.copy() # Creating a new copy of list b in list c
17 print("List b:",b)
18 print("List c:",c)
19
```

```
20 print("\n Referencing: list d is an other name of list b")
21 d = b # Referencing: list d is an other name of list b
22 print("List d:",d)
23
24 print("\n Appends all members of b to list c")
25 c.extend(b) # Appends all members of b to list c
26 print("List c:",c)
27
28 print ("\nInserting 15 at index 1 in list b")
29 b.insert(1,15)# Inserting 15 at index 1 in list b
30 print("List b:",b)
31 print("List d:",d)
```

Funciones para remover valores de listas

```
listas.py  remove.py ×
Split Terminal (⌘) remove.py > ...
1  d = [1, 2, 3, 4, 5, 6]
2
3  print("List d: ", d)
4
5  # remove: remove the value at index 1 from list d using pop() function
6  d.pop(1)
7  print("List d: ", d)
8
9  # remove: remove the value 3 from list d using remove() function
10 d.remove(3)
11 print("List d: ", d)
12
13 # clear: remove all elements from list d
14 d.clear()
15 print("List d: ", d)
```


Funciones relacionadas a búsquedas

```
listas.py  remove.py  busquedas.py ×
clase7 > busquedas.py > ...
1  e = [1, 1, 3, 40, 4, 4, 4, 5]
2
3  print("List e: ", e)
4  print("El  numero de 1s en la lista es: ", e.count(1))
5  print("El  indice de 40 es : ", e.index(40))
```


Funciones relacionadas al orden

```
class7 > orden.py > ...  
1  f = [1, 1, 30, 40, 4, 4, 40, 5]  
2  
3  print("List f: ", f)  
4  f.reverse()  
5  print("List f: ", f)  
6  f.sort()  
7  print("List f: ", f)
```