

Listas

Lógica y Algoritmia

Listas



Listas (1)

- Son conjuntos dinámicos de cualquier tipo de dato que se acceden desde un solo nombre a través de un índice.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	Hola	5	45	Z	99	16	A	2	10	4

Imprimir dato en el índice 6



Listas:

Ejemplo 1

Código:

```
lista=['Hola', 5, 45, 'z', 99, 16, 'A', 2, 10, 4]
print(lista)
print(lista[6])
```

Ejecución:


```
>>>
['Hola', 5, 45, 'z', 99, 16, 'A', 2, 10, 4]
A
>>>
```

Listas (2)

- Las listas pueden ser modificadas agregando datos.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	Hola	5	45	Z	99	16	A	2	10	4

Agregar dato en el índice 5:



Índice	0	1	2	3	4	5	6	7	8	9	10
Dato	Hola	5	45	Z	99	P	16	A	2	10	4

La lista crece,
las posiciones siguientes se desplazan hacia la derecha



Listas:

Ejemplo 2

Código:

```
lista=['Hola',5,45,'Z',99,16,'A',2,10,4]
print(lista)
lista.insert(5,'P')
print(lista)
```

Ejecución:

```
>>>
['Hola', 5, 45, 'Z', 99, 16, 'A', 2, 10, 4]
['Hola', 5, 45, 'Z', 99, 'P', 16, 'A', 2, 10, 4]
>>>
```

Listas (3)

- Las listas también pueden ser modificadas eliminando datos.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	Hola	5	45	Z	99	16	A	2	10	4

Eliminar dato en el índice 5:

Índice	0	1	2	3	4	5	6	7	8
Dato	Hola	5	45	Z	99	A	2	10	4

La lista se reduce,
las posiciones posteriores se desplaza a la izquierda

Listas:

Ejemplo 3

Código:

```
lista=['Hola', 5, 45, 'Z', 99, 16, 'A', 2, 10, 4]
print(lista)
del lista[5]
print(lista)
```

Ejecución:

```
>>>
['Hola', 5, 45, 'Z', 99, 16, 'A', 2, 10, 4]
['Hola', 5, 45, 'Z', 99, 'A', 2, 10, 4]
>>>
```


Listas (4)

- Una de las formas de saber la cantidad de elementos que posee una lista es utilizando la función “len()”.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	Hola	5	45	Z	99	16	A	2	10	4

El tamaño del arreglo es de 10 elementos

Listas:

Ejemplo 4

Código:

```
lista=['Hola',5,45,'z',99,16,'A',2,10,4]  
print(len(lista))
```


Ejecución:

```
>>>  
10  
>>>
```

Listas (5)

- Las listas pueden ser tomadas en segmentos más pequeños de la misma.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	Hola	5	45	Z	99	16	A	2	10	4



Imprimir datos desde el 4 hasta el 7 sin incluir el 7

Listas:

Ejemplo 5

Código:

```
lista=['Hola', 5, 45, 'z', 99, 16, 'A', 2, 10, 4]  
print(lista)  
print(lista[4:7])
```


Ejecución:

```
>>>  
['Hola', 5, 45, 'z', 99, 16, 'A', 2, 10, 4]  
[99, 16, 'A']  
>>>
```

Listas (6)

- Una manera acceder a todos los datos de una lista uno por uno es utilizando el ciclo repetitivo “for”.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	Hola	5	45	Z	99	16	A	2	10	4



Imprimir todos los datos de la lista uno por uno

Listas:

Ejemplo 6

Código:

```
lista=['Hola', 5, 45, 'Z', 99, 16, 'A', 2, 10, 4]
for elemento in lista:
    print(elemento)
```

Ejecución:

```
>>>
Hola
5
45
Z
99
16
A
2
10
4
>>>
```

Listas (7)

- Las listas con datos numéricos (enteros o decimales) pueden ser ordenadas de manera ascendente.

Índice	0	1	2	3	4	5	6	7	8	9
Dato	2.5	3	4.5	5	1	1.5	2	3.5	5.5	4

Índice	0	1	2	3	4	5	6	7	8	9
Dato	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5

Listas:

Ejemplo 7

Código:

```
lista=[2.5,3,4.5,5,1,1.5,2,3.5,5.5,4]  
print(lista)  
lista.sort()  
print(lista)
```

Ejecución:

```
>>>  
[2.5, 3, 4.5, 5, 1, 1.5, 2, 3.5, 5.5, 4]  
[1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5]  
>>>
```


Listas (8)

- Las listas con caracteres y cadenas de caracteres también pueden ser ordenadas de manera ascendente por orden alfabético.

Índice	0	1	2	3	4
Dato	'Zapato'	'Sandalia'	'Armario'	'B'	'Espejo'

Índice	0	1	2	3	4
Dato	'Armario'	'B'	'Espejo'	'Sandalia'	'Zapato'

Listas:

Ejemplo 8

Código:

```
lista=['Zapato', 'Sandalia', 'Armario', 'B', 'Espejo']  
print(lista)  
lista.sort()  
print(lista)
```

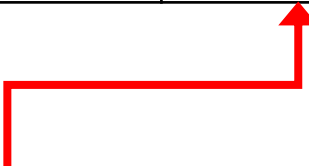
Ejecución:

```
>>>  
['Zapato', 'Sandalia', 'Armario', 'B', 'Espejo']  
['Armario', 'B', 'Espejo', 'Sandalia', 'Zapato']  
>>>
```

Listas (9)

- Cuando se quiere ordenar una lista, es importante tener en cuenta que no se pueden mezclar caracteres y cadenas de caracteres con números, ya que Python no tiene establecido un criterio de comparación para ordenarlos.

Índice	0	1	2	3	4
Dato	'Zapato'	'Sandalia'	'Armario'	5	'Espejo'



El número 5 no puede compararse con las demás cadenas de caracteres:

Referencias Bibliográficas

- [1] Zelle, John M. Python Programming an Introduction to Computer Science. -- 2nd ed. -- Washington : Franklin, Beedle & Associates Inc, 2010.
- [2] Dawson, Michael. Python Programming for the Absolute Beginner. -- 3th ed. -- Australia : Cengage Learning, 2010.
- [3] Rosaura Gutiérrez Almeyda, Urbano Eliécer Gómez Prada, Jairo Viola, y Diana Teresa Gómez Forero. Presentación de clase: Listas en Python. Universidad Pontificia Bolivariana Seccional Bucaramanga. 2015.