

## Programas que involucran datos estructurados (Listas y Tuplas)

Los **corchetes []** se utiliza para declarar una lista.

La **coma**, se utiliza para separar los elementos dentro de una lista.

[1, 2, 3, 5, 6, 7, 8, 10]

Los elementos de una lista puede ser cualquier dato primitivo.

### Ejercicio 1

Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una lista y la muestre por pantalla.

```
materias = ["Matemáticas", "Física", "Química", "Historia", "Lengua"]
for materia in materias:
    print(materia)
```

### Ejercicio 2

Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una lista y la muestre por pantalla el mensaje Yo estudio <asignatura>, donde <asignatura> es cada una de las asignaturas de la lista.

```
materias = ["Matemáticas", "Física", "Química", "Historia", "Lengua"]
for materia in materias:
    print("Yo estudio " + materia)
```

### Ejercicio 3

Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una lista, pregunte al usuario la nota que ha sacado en cada asignatura, y después las muestre por pantalla con el mensaje En <asignatura> has sacado <nota> donde <asignatura> es cada una de las asignaturas de la lista y <nota> cada una de las correspondientes notas introducidas por el usuario.

```
materias = ["Matemáticas", "Física", "Química", "Historia", "Lengua"]
calificaciones = []
for materia in materias:
    calificacion = float(input("Cual es tu calificacion en " + materia + ": "))
    calificaciones.append(calificacion)
for i in range(len(materias)):
    print("En " + materias[i] + " saque: " + str(calificaciones[i]))
```

### Ejercicio 4

Escribir un programa que pregunte al usuario los números ganadores de la lotería primitiva, los almacene en una lista y los muestre por pantalla ordenados de menor a mayor.

```
loteria = []
for i in range(6):
    loteria.append(int(input("Introduce el numero ganador: ")))
loteria.sort()
print("Los numeros ganadores son " + str(loteria))
```

### Ejercicio 5

Escribir un programa que almacene en una lista los números del 1 al 10 y los muestre por pantalla en orden inverso separados por comas.

```
numeros = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

# Solucion 1
reversa=""
for i in range(1, 11):
    reversa+=str(numeros[-i])+ ", "

print(reversa)

# Solucion 2
numeros.reverse()

reversa=""
for numero in numeros:
    reversa+=str(numero)+ ", "

print(reversa)
```

### Ejercicio 6

Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una lista, pregunte al usuario la nota que ha sacado en cada asignatura y elimine de la lista las asignaturas aprobadas. Al final el programa debe mostrar por pantalla las asignaturas que el usuario tiene que repetir.

```
materias = ["Matematicas", "Fisica", "Quimica", "Historia", "Lengua"]

reprobadas = []

for materia in materias:
    calificacion = float(raw_input("Cual es tu calificacion en " + materia + ": "))
    if calificacion <= 5:
        reprobadas.append(materia)

text="Usted reprobó: "
for reprobada in reprobadas:
    text+= reprobada + ", "

print(text)
```

### Ejercicio 7

Escribir un programa que almacene el abecedario en una lista, elimine de la lista las letras que ocupen posiciones múltiples de 3, y muestre por pantalla la lista resultante.

```
alfabeto = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']

for i in range(len(alfabeto), 1, -1):
    if i % 3 == 0:
        alfabeto.pop(i-1)

print(alfabeto)

alfabeto = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']

for i in range(len(alfabeto), 1, -1):
    if i % 3 == 0:
        alfabeto.pop(-i)

print(alfabeto)
```

### Ejercicio 8

Escribir un programa que pida al usuario una palabra y muestre por pantalla si es un palíndromo.

```
palabra = list(raw_input("introduce una palabra: "))

if palabra[::-1] == palabra:
    print("Palindromo")
else:
    print("No es Palindromo")
```

```
palabra = list(raw_input("Escribe una palabra: "))

boolean = False

for i in range(len(palabra)):
    if palabra[i] == palabra[len(palabra)-1-i]:
        boolean = True
    else:
        break

if boolean == True:
    print ("Es un palindromo")
else:
    print("No es un palindromo")
```

### Ejercicio 9

Escribir un programa que pida al usuario una palabra y muestre por pantalla el número de veces que contiene cada vocal.

```
palabra = raw_input("Introduce una palabra: ")
vocales = ['a', 'e', 'i', 'o', 'u']

for vocal in vocales:
    veces = 0
    for letra in palabra:
        if letra == vocal:
            veces+=1
    print("La vocal " + vocal + " aparece " + str(veces) + " veces")
```

### Ejercicio 10

Escribir un programa que almacene en una lista los siguientes precios, 50, 75, 46, 22, 80, 65, 8, y muestre por pantalla el menor y el mayor de los precios.

```
numeros = [50, 75, 46, 22, 80, 65, 8]

numeros.sort()

print("El minimo es: " + str(numeros[0]) + " el maximo es: " + str(numeros[len(numeros)-1] ))
```

### Ejercicio 11

Escribir un programa que almacene los vectores (1,2,3) y (-1,0,2) en dos listas y muestre por pantalla su producto escalar.

### Ejercicio 12

Escribir un programa que almacene las matrices en una lista y muestre por pantalla su producto.

Nota: Para representar matrices mediante listas usar listas anidadas, representando cada vector fila en una lista.

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix} \quad y \quad B = \begin{pmatrix} -1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

### Ejercicio 13

Escribir un programa que pregunte por una muestra de números, separados por comas, los guarde en una lista y muestre por pantalla su media y desviación típica.