

---

# Programación I - Práctica III

Estructuras de Control y Estructuras de Datos en Python - If - For - While - Listas - Diccionarios

---

- [Ejemplos en GitHub](#)

## Ejercicios:

1. Desarrolle un programa que pida dos palabras y diga si riman o no. Si coinciden las tres últimas letras tiene que decir que riman. Si coinciden sólo las dos últimas tiene que decir que riman un poco y si no, que no riman.
  2. Desarrolle un programa donde cree un diccionario llamado **cuenta** cuyos siguientes datos deberán ser leídos de teclado: **nombre**, **apellido**, **edad**, **dni** y **clave**. Donde **clave** es el producto de la multiplicación del número de **dni** y la longitud del **nombre** concatenado al **apellido**. Imprima dicho diccionario al finalizar.
  3. Desarrolle un programa que sume en 1 a una variable de tipo **int** e imprima dicho número en pantalla hasta que dicha variable no sea igual al campo **clave** utilizado anteriormente.
  4. Desarrolle un programa que cree una estructura con variables como temperatura, humedad, probabilidad de lluvias y características del viento como velocidad y dirección y que además almacene el nombre y número de día y la hora. Los datos de cada variable deben ser ingresados por teclado. Analice qué estructura de datos y de control le es conveniente para almacenar estos datos.
  5. Desarrolle un programa en el cual dadas un conjunto de las estructuras creadas anteriormente se le pueda consultar el día con mayor temperatura y el día con menores probabilidades de lluvia. Analice qué estructura de datos y de control le es más conveniente para almacenar estos datos.
  6. Desarrolle un programa que lea palabras (strings) del teclado y almacene cada palabra en una lista llamada **comandos = []** (para agregar elementos a una lista utilice la función **lista.append(elemento)**). El programa deberá de dejar de pedir palabras del teclado cuando se ingresen 4 números 0 ("0000") el cual no deberá ser agregado a la lista. Imprima la lista al finalizar.
-

7. Desarrolle un programa que dada una lista de palabras (**palabras** = [**"string1"**, **"string2"**, **"stringN"**]) cree un diccionario en el cual las claves del mismo son las palabras en la lista y sus valores correspondientes sean la cantidad de veces que dicha palabra aparece en la lista (**dict** = {**"string1"**: 1, **"string2"**: 5, **"string3"**:2}). Imprima el diccionario y la lista al finalizar.
8. Desarrolle un programa que dado un diccionario como el explicado en el punto anterior agregue a una lista las palabras (**claves**) del diccionario y sus repeticiones (**valor**) ordenadas por la de mayor valor a la menor como **tuplas** (**apariciones** = [**(string2,5)**, **(string3,2)**, **(string1,1)**]. Imprima la lista al finalizar.
9. Desarrolle un programa que dada la lista realizada en el ejercicio anterior y la lista realizada en el primer ejercicio cree una lista de diccionarios que tengan la siguiente estructura:

**arbolDeComandos =**

```
{"string1":{"apariciones":1,"izq":"palabra_izquierda","der":"palabra_derecha"},"string2":{"apariciones":5,"izq":"palabra_izquierda","der":"palabra_derecha"},"string3":{"apariciones":2,"izq":"palabra_izquierda","der":"palabra_derecha"}}
```

Otra Forma de verlo:

**arbolDeComandos = {**

```
  "string1": {
    "apariciones":1,
    "izq":"palabra_izquierda",
    "der":"palabra_derecha"
  },
  "string2": {
    "apariciones":5,
    "izq":"palabra_izquierda",
    "der":"palabra_derecha"
  },
  "string3": {
    "apariciones":2,
```

```

        "izq": "palabra_izquierda",
        "der": "palabra_derecha"
    }
}

```

Dónde palabra izquierda y palabra derecha son las palabras de la izquierda y de la derecha (**una posición menos y una posición más**) en la lista creada en el ejercicio 1 con respecto a la palabra clave del diccionario y apariciones son las apariciones de la misma en dicha lista.

- Desarrolle un programa que utilizando la librería **random (import random)** cree un número de **N dígitos utilizando la función random.randint(0,N+1)** donde **N** es ingresado desde teclado. Luego de ingresar el número se debe informar lo siguiente: Si **N < 4** imprimir el string **"Easy"**, si **4 < N <= 8** imprimir el string **"Medium"** y **8 < N <= 16** imprimir el string **"Hard!"**.  
Luego de imprimir el mensaje el programa de entrar en un **while** que seguirá hasta que la condición **finalizar = True**. Dentro del **while** debe mostrarse cada dígito del número creado con **random** pero con un guión bajo "\_" suplantando cada uno y debe solicitar al usuario un número desde el teclado. Si el número ingresado es parte del número creado con **random** deberá mostrarse ese número en vez de el guión bajo en el próximo **loop** del **while**. Al terminar de adivinar el número o ingresar **0000** deberá imprimirse la cantidad de veces que se acertó un número y la cantidad de veces que no se acertó.