

CURSO DE PROGRAMACIÓN FULL-STACK

Python / Guía 3

# Ejercicios

## Control de flujos



## Revisemos lo aprendido hasta aquí y lo que vamos a ejercitar!

- ✓ Controlar el flujo de datos con los bucles for
- ✓ Usar las funciones incorporadas range y enumerate

## Ejercicios de aprendizaje

1. Imprimir por pantalla los números impares del 1 al 10 al revés
2. Crear un bucle que sume los pares del 0 al 100.
3. Escribir un programa que requiera del usuario números, hasta que ingrese 0. Cuando lo haga, mostrar por pantalla la suma de todos los números ingresados.
4. Pedirle a un usuario que ingrese un número, y devolver los dígitos totales del número. Por ejemplo, si el número es 75869, la salida debería ser 5
5. Escribir un programa que lea un número impar por teclado. Si el usuario no introduce un número impar, debe repetirse el proceso hasta que lo introduzca correctamente.
6. Escribir un programa que sume todos los números enteros impares desde el 0 hasta el 100.

Ayuda. Prueba usar la función incorporada de Python sum de la siguiente forma: `sum(range(... ))`

7. Escribir un programa que permita al usuario ingresar dos años y luego imprima todos los años en ese rango. Tienen que ser los años bisiestos. Nota: para que un año sea bisiesto debe ser divisible por 4 y no debe ser divisible por 100, excepto si también es divisible por 400.

8. Se decide organizar un juego de estrategia. Se forman dos equipos de seis integrantes cada uno. Un integrante del grupo es "el jefe". Se comunicarán mediante un canal común: deben buscar la forma de ocultar el contenido de sus mensajes. Uno de los equipos decide utilizar un método antiguo de encriptación llamado "la cifra del César", que consiste en correr cada letra del mensaje –considerando la posición de cada una en el alfabeto– una determinada cantidad de lugares. Ejemplo: si el valor de "corrimiento" es 2, la palabra "ATAQUE" se transforma en "CVCSWG". Cada día, el "jefe" del equipo debe enviar un mensaje a su grupo. Escribir un programa que permita encriptar los 5 mensajes. El "corrimiento" (cantidad de letras que se correrán hacia la derecha) será dado por el usuario antes de comenzar a encriptar. Los 5 mensajes usarán el mismo "corrimiento". Nota: si el alfabeto termina antes de poder correr la cantidad de lugares necesarios, se vuelve a comenzar desde la letra "a". Ejemplo: la palabra "EXTRA" corrida 3 lugares se convierte en "HAWUD". Utilizando el alfabeto español, de 27 letras, el siguiente cálculo matemático permite volver a comenzar por el principio una vez que se llegó a la "z":  $(\text{índice de la letra a correr} + \text{corrimiento}) \% 27$ . Solo se encriptarán las letras de los mensajes, dejando al resto de caracteres sin modificación.

9. Realizar el siguiente ejercicio, y después ver el video:

Escribir un programa que lea dos números por teclado y permite elegir entre 4 opciones en un menú: 1) Mostrar una suma de los dos números. 2) Mostrar una resta de los dos números (el primero menos el segundo). 3) Mostrar una multiplicación de los dos números. 4) Si elige esta opción se interrumpirá la impresión del menú y el programa finalizará. 5) En caso de no introducir una opción válida, el programa informará de que no es correcta.



**Ver video de "Ejemplo operaciones matemáticas"**

**10.** Realizar el siguiente ejercicio, y después ver el video:

Escribir un programa que reciba tres entradas:

- usuario: str
- password: str
- repetir\_password: str

Luego debe validar que:

- La longitud de usuario sea mayor que 5
- El primer carácter de usuario no debe ser un número
- El password debe ser menor a 4 caracteres
- repetir\_password debe ser igual a password



**Ver video de “Ejemplo usuario y contraseña”**