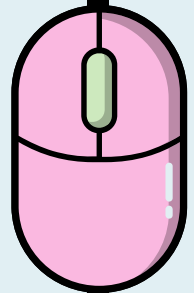
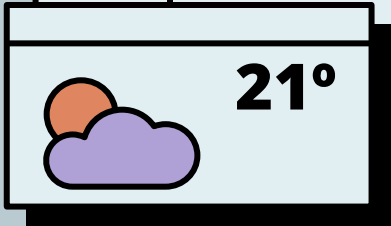
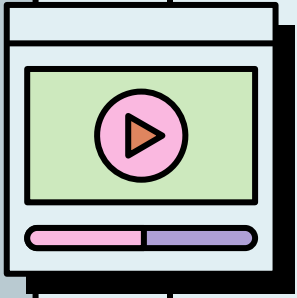


Ayudantía 12

Programación



>>>>

Diego Duhalde

.....



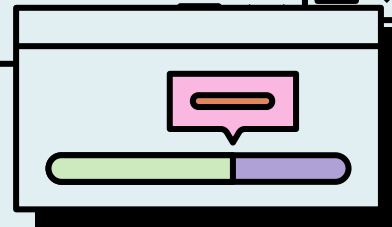


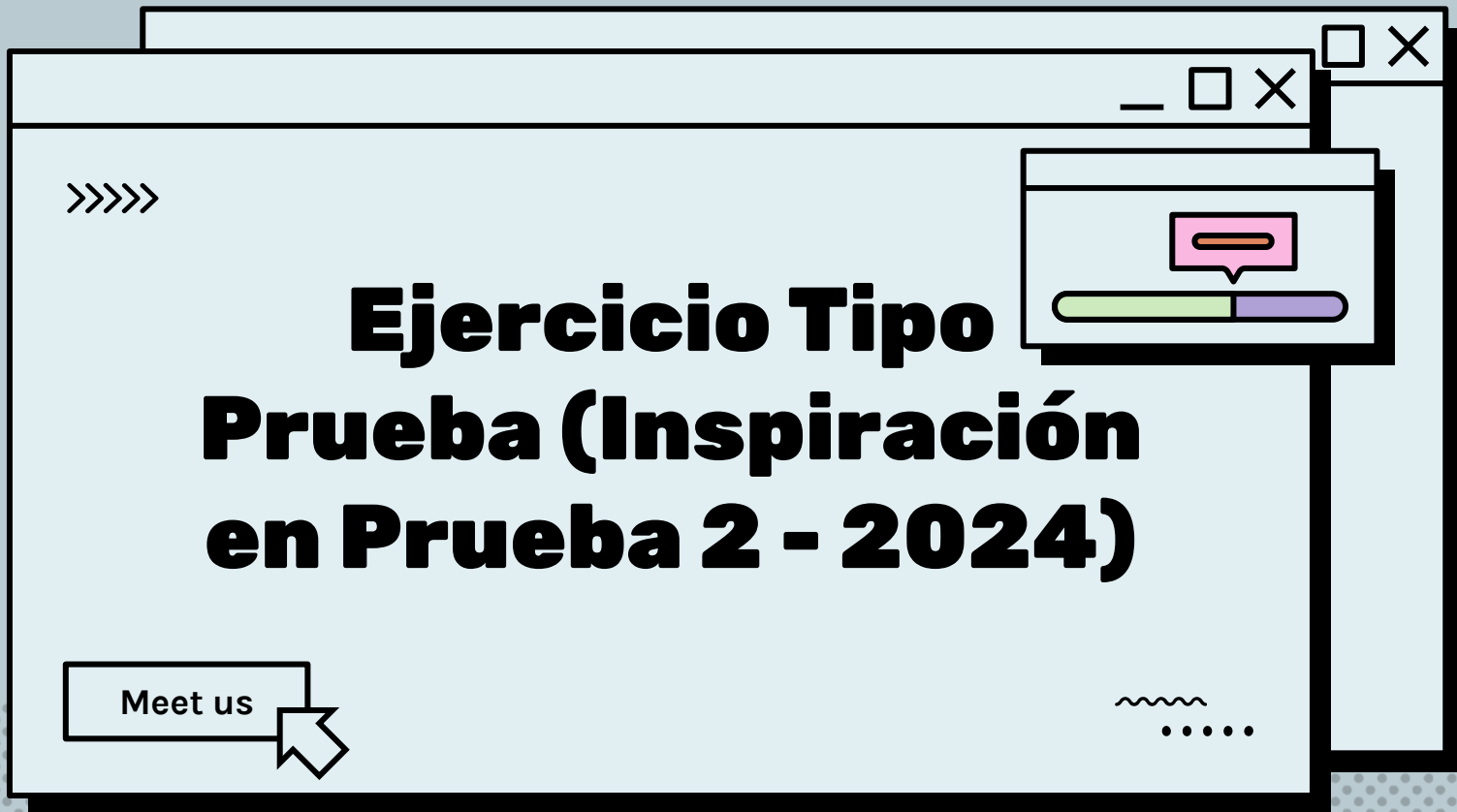
>>>>

Pasos típicos

1. Identificar **que nos están pidiendo** (Que debemos retornar o mostrar por pantalla)
2. Identificar **que tenemos** (Pueden ser cosas que ingrese el usuario o cosas ya creadas)
3. Si nos dice que el usuario lo debe ingresar **si o si** input(), si nos dice que ya existe **variableUsar**, si o si ver como podemos usar esa variable.
4. Identificar los **pasos** y **variables** que debemos usar para poder llegar al punto 1.
5. Usar **hints** y verificar que cumplimos las especificaciones.
6. Ejecutar el código en **nuestra cabeza**, paso por paso.

Meet us







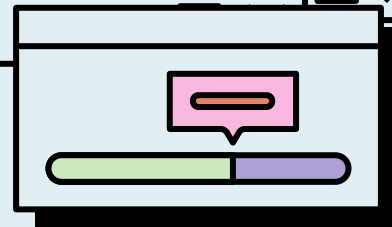
>>>>

Ejercicio 1

Un gimnasio desea analizar el peso de sus usuarios para diseñar mejor sus rutinas. Cree un programa que:

- Solicite al usuario la cantidad de usuarios a registrar (numUsuarios).
- Cree un arreglo de una dimensión (NO usar listas simples) del tamaño numUsuarios.
- Solicite al usuario ingresar el peso de cada uno (en kilogramos) y guarde estos datos en el arreglo.
- Muestre por pantalla: el peso mínimo, el peso máximo, y el peso promedio utilizando funciones de NumPy

Meet us





>>>>

Ejercicio 2

Se quiere imprimir la letra **T** en pantalla, de acuerdo al número de filas y columnas ingresadas por el usuario.

Ingrese número de columnas: 5

Ingrese número de filas: 4

*

*

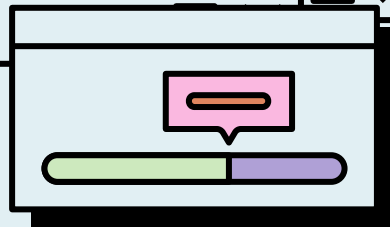
*

Sin embargo, el siguiente código tiene errores y no imprime la figura correctamente:

Meet us



~~~~~  
.....





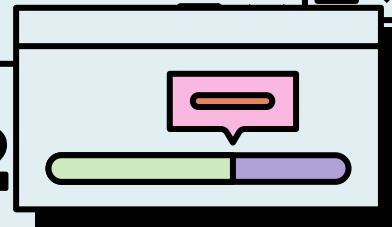
>>>>

## Cont Ejercicio 2

```
filas = int(input("Ingrese el número de filas"))
columnas = int(input("Ingrese el número de columnas"))
for i in range(filas):
    for j in range(columnas):
        if i == 0 or j == columnas//2:
            print("*", end="")
        print()
```

- Identifique y explique **dos errores** presentes en la lógica o estructura del código.
- Corrija el código para que imprima la letra **T** como se espera.

Meet us





>>>>

## Ejercicio 3

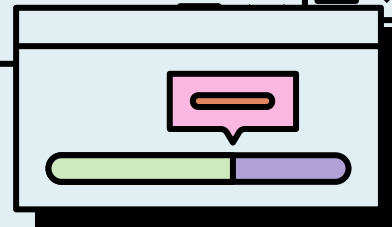
Escriba una función llamada `detectarPozos` que reciba una matriz de temperatura en grados Celsius. Cada número representa la temperatura en un punto del suelo.

Un pozo térmico es un punto que tiene una temperatura menor que todos sus 8 vecinos. La función debe:

Retornar la **cantidad total de pozos térmicos**

Imprimir por pantalla la **temperatura más baja** entre los pozos encontrados.

Meet us





>>>>

# Cont Ejercicio 3

```
temperaturas = [  
    [22, 25, 20, 21, 24],  
    [26, 19, 22, 18, 23],  
    [27, 21, 17, 20, 26],  
    [28, 22, 19, 22, 24],  
    [29, 25, 23, 24, 22]  
]  
detectarPozos(temperaturas) # return 1
```

“La temperatura más baja en un pozo es 17°C.”

Meet us

