



Ejercicio Práctico: Evaluador de Estudiantes con Funciones

Descripción

Vas a crear un programa en Python que reciba el nombre de un estudiante y su promedio final, determine si aprobó o reprobó (nota mínima 6.0), y almacene los resultados. Usarás una **función definida por el usuario** para realizar la evaluación. El programa debe permitir ingresar múltiples estudiantes hasta que el usuario decida salir, y al final, debe mostrar un resumen.

Objetivos del ejercicio

- Definir funciones con parámetros y retorno.
 - Usar `input()` dentro de un ciclo `while`.
 - Aplicar estructuras condicionales para evaluar el resultado.
 - Utilizar listas o contadores como acumuladores.
 - Mostrar un resumen final con resultados agrupados.
-

Instrucciones

1. Crea un archivo llamado `evaluador_estudiantes.py`.
2. Define una función llamada `evaluar_estudiante(nombre, promedio)` que:
 - Imprima si el estudiante aprobó o reprobó.
 - Retorne `True` si aprobó, `False` si reprobó.

3. Desde el programa principal:

- Usa un ciclo `while` para permitir ingresar datos de varios estudiantes.
- Pide el nombre y el promedio del estudiante.
- Llama a la función y cuenta cuántos aprobaron y reprobaron.
- Finaliza el ciclo si el usuario escribe "fin" como nombre.

4. Al terminar, imprime un resumen con:

- Total de estudiantes evaluados
- Cantidad de aprobados
- Cantidad de reprobados

Ejemplo de ejecución

Evaluación de estudiantes

Ingrese el nombre del estudiante (o escriba 'fin' para salir): Ana

Ingrese el promedio final: 7.2

✅ Ana ha aprobado.

Ingrese el nombre del estudiante (o escriba 'fin' para salir): Luis

Ingrese el promedio final: 5.3

❌ Luis ha reprobado.

Ingrese el nombre del estudiante (o escriba 'fin' para salir): fin

Resumen:

Total evaluados: 2

Aprobados: 1

Reprobados: 1

✅ Solución: `evaluador_estudiantes.py`

Función para evaluar si un estudiante aprueba o reprueba

```
def evaluar_estudiante(nombre, promedio):
```

```
    if promedio >= 6.0:
```

```
        print(f"✅ {nombre} ha aprobado.")
```

```
        return True
```

```

else:
    print(f"❌ {nombre} ha reprobado.")
    return False

# Programa principal
print("📋 Evaluación de Estudiantes")
print("-----")

# Contadores
aprobados = 0
reprobados = 0
total = 0

while True:
    nombre = input("\nIngrese el nombre del estudiante (o escriba 'fin' para salir): ")

    if nombre.lower() == "fin":
        break

    try:
        promedio = float(input("Ingrese el promedio final: "))
        resultado = evaluar_estudiante(nombre, promedio)
        total += 1


        if resultado:
            aprobados += 1
        else:
            reprobados += 1

    except ValueError:
        print("⚠️ Error: El promedio debe ser un número. Intente nuevamente.")

# Resumen final
print("\n📊 Resumen:")
print(f"Total evaluados: {total}")
print(f"Aprobados: {aprobados}")
print(f"Reprobados: {reprobados}")

```

Ejemplo de ejecución:

 Evaluación de Estudiantes

Ingrese el nombre del estudiante (o escriba 'fin' para salir): Ana

Ingrese el promedio final: 7.8

✅ Ana ha aprobado.

Ingrese el nombre del estudiante (o escriba 'fin' para salir): Pedro

Ingrese el promedio final: 5.9

✗ Pedro ha reprobado.

Ingrese el nombre del estudiante (o escriba 'fin' para salir): fin



Resumen:

Total evaluados: 2

Aprobados: 1

Reprobados: 1

Este ejercicio permite aplicar:

- Funciones con lógica de retorno (`True` / `False`)
- Control de flujo con `if/else`
- Ciclo `while` con salida personalizada
- Validación básica con `try/except`
- Acumulación de resultados y reporte final