

Simulacro de Examen – Programación en Python

Requisitos Generales

- El proyecto debe estar **modularizado** en varios archivos, organizados dentro de un paquete con `__init__.py`.
 - El menú debe implementarse en un **bucle infinito**.
 - La **última opción del menú será la única válida para terminar la ejecución del programa**.
 - Todas las operaciones del menú deben estar implementadas en **funciones separadas** dentro de módulos diferentes.
-

Consigna

Desarrollar un programa que muestre el siguiente menú:

```
----- MENÚ DE OPCIONES -----  
  
1. Contar cuántos números entre 1 y N son divisibles por 3 **o** terminan  
   en 3.  
  
2. Imprimir una pirámide de asteriscos de altura N.  
  
3. Dado un número N, mostrar la secuencia de los primeros N términos de la  
   serie: 1, -2, 3, -4, 5, -6, ...  
  
4. Mostrar la cantidad de números primos entre 1 y N.  
  
5. Verificar si un número es palíndromo (número capicúa).  
  
6. Salir del programa.  
  
Seleccione una opción:
```

Detalles por opción

Opción 1:

Función que reciba **N** e imprima cuántos números entre 1 y N cumplen **al menos una** de estas condiciones:

- Son divisibles por 3
 - Terminan en 3 (por ejemplo: 3, 13, 23...)
-

Opción 2:

Función que recibe un número **N** y muestra una **pirámide de asteriscos** centrada de altura **N**.

Ejemplo para N = 3:

```
*  
* * *  
* * * * *
```

Opción 3:

Función que genera e imprime los primeros **N** términos de la **serie alternada**:

1, -2, 3, -4, 5, -6, ...

Los valores deben imprimirse en una sola línea, separados por coma.

Opción 4:

Función que reciba un número **N** e imprima cuántos **números primos** hay entre 1 y **N**.

Opción 5:

Función que recibe un número entero positivo e indica si es un **palíndromo** (capicúa).

Debe validarse que el número ingresado sea mayor a 0.

Opción 6:

Mostrar un mensaje de despedida y finalizar el programa.

Requisitos técnicos

- El menú y la lógica principal deben estar en un archivo `main.py`.
 - Cada opción debe estar implementada en un archivo diferente dentro de un paquete llamado `operaciones`, que debe incluir su `__init__.py`.
 - El proyecto debe utilizar funciones, condicionales (`if`, `elif`), bucles (`for`, `while`), y entrada/salida de datos (`input`, `print`).
-

Entregables requeridos

- Crear un **entorno virtual** con nombre `simulacro`.
 - Exportar el archivo `.yaml` del entorno.
 - Generar un archivo de texto con la estructura de carpetas del proyecto.
 - Generar un ejecutable del programa principal.
 - Comprimir el proyecto completo y entregar en formato `.zip`.
-