
Trabajo Práctico Integrador Final

Gestor de países

Diego Fernández - Renzo Galetto

Comisión 5

Prof. Cinthia Rigoni

Prof. Matias Torres

Tecnicatura Universitaria en Programación

Programación 1





Introducción

Se desarrolló una aplicación de consola en Python para gestionar y analizar información demográfica y geográfica de países.

Fuente de Datos:

El sistema opera a partir de un archivo de entrada en formato CSV (`.csv`).

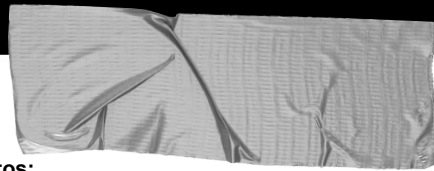
Objetivos Principales:

Aplicar los conceptos fundamentales de Programación 1 (estructuras de datos, funciones, control de flujo).

Resolver un problema práctico de manejo de datos.

Implementar funcionalidades de consulta, filtrado, ordenamiento y estadísticas básicas.

Alcance y Funcionalidades del Sistema



Carga de Datos:

- Lectura y parseo del archivo CSV inicial.

Consultas y Filtros:

- Búsqueda por nombre (coincidencia parcial).
- Filtrado por Continente.
- Filtrado por rango de Población y Superficie.

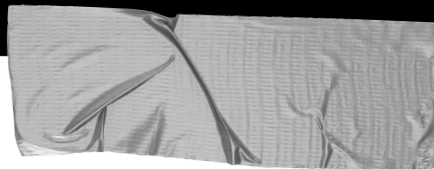
Ordenamiento:

- Clasificación de países por Nombre, Población o Superficie.
- Soporte para orden Ascendente y Descendente.

Análisis Estadístico:

- Cálculo de promedios (Población, Superficie).
- Identificación de máximos y mínimos (País con mayor/menor población).
- Conteo de países por continente

Conceptos Técnicos Aplicados



Estructuras de Datos:

- **Listas:** Utilizadas como el contenedor principal (`países`) para almacenar la colección de registros. Permiten el almacenamiento ordenado y la iteración.
- **Diccionarios:** Usados para representar cada país de forma individual.
 - Facilitan el acceso semántico a los datos (pares `clave:valor`).
 - *Ejemplo de acceso:*
`pais['poblacion']`

Modularización:

Funciones: El código se estructuró en funciones con responsabilidad única (ej. `cargar_paises()`, `buscar_pais()`, `estadisticas()`).

- *Beneficios:* Reutilización de código, legibilidad y facilidad de mantenimiento.

Manejo de Archivos:

Módulo CSV: Se empleó el módulo `csv` de Python para la lectura estructurada del *dataset*, transformando eficientemente cada fila en un diccionario.

Control de Flujo:

Bucles (`for` `while`): Esenciales para recorrer las listas (en búsquedas, filtros) y para gestionar la persistencia del menú principal.

Condicionales (`if` `elif` : Utilizados para la lógica de selección del menú y la aplicación de filtros.

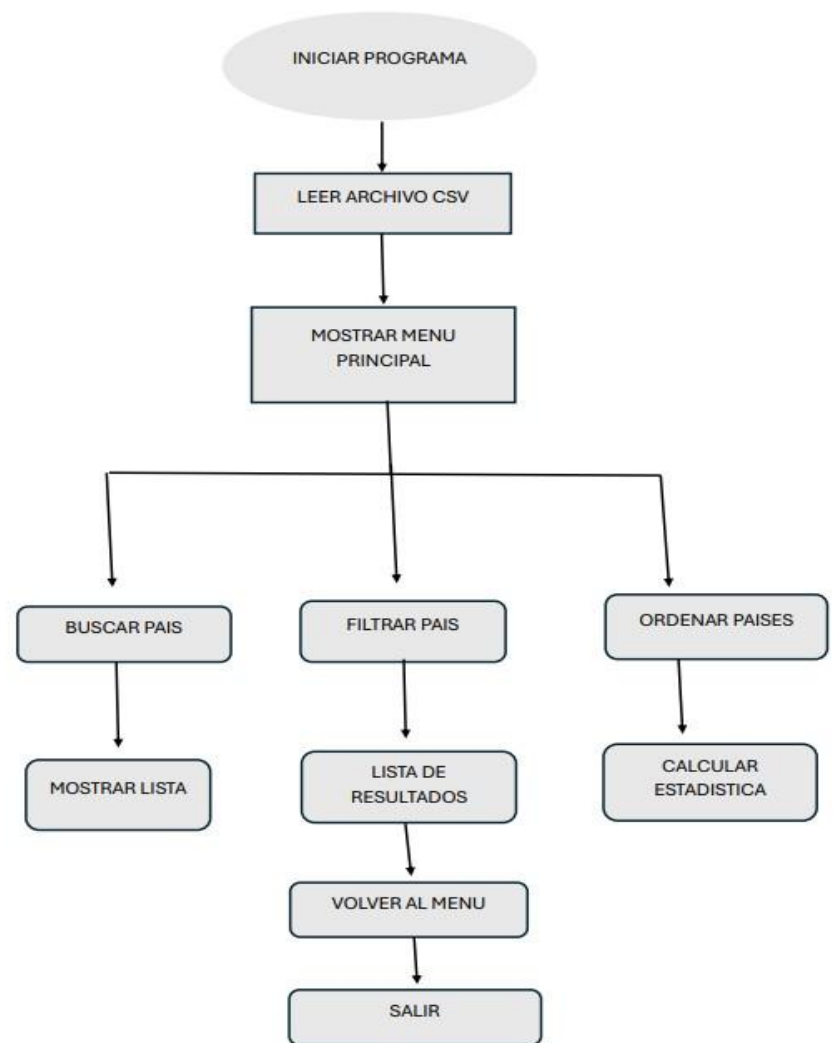
Flujo del programa y validaciones:

- [Inicio] -> [Leer Archivo CSV] -> [Mostrar Menú Principal]
- [Menú Principal] -> [Selección de Usuario (Buscar, Filtrar, Ordenar, Estadísticas, Salir)]
- [Procesar Opción] -> [Mostrar Resultados] -> [Volver al Menú Principal]
- [Salir] -> [Fin]

Validaciones y Manejo de Errores:

- Verificación de la existencia del archivo CSV antes de la carga.
- Control de formato de datos numéricos (población, superficie).
- Manejo de búsquedas o filtros que no arrojan resultados.
- Emisión de mensajes claros de error y éxito al usuario.

Diagrama de flujo del programa





Resultados Esperados

El sistema permite ejecutar consultas ágiles desde la consola.

La información se presenta de manera clara y organizada.

El usuario puede comparar países y analizar datos demográficos básicos de forma interactiva.

Conclusiones y Aprendizajes

El proyecto permitió aplicar de forma integrada los conocimientos de Programación 1, especialmente:

- Estructuras de datos (Listas y Diccionarios).
- Modularización mediante funciones.
- Lectura de archivos externos y validación de datos.

Competencias Adquiridas:

- Fortalecimiento de la lógica algorítmica.
- Desarrollo de buenas prácticas en la organización y estructuración del código.

Proyección:

- El trabajo sienta las bases para el desarrollo de proyectos de software más complejos y el manejo de mayores volúmenes de datos.