

Carrera: Tecnicatura Universitaria en Programación

Materia: Programación I

Año: 2025

Alumnos: Javier Soliz

Emanuel Ortigoza

Título del trabajo:

Gestión de Países

Introducción

El presente trabajo integrador tiene como objetivo aplicar los conocimientos adquiridos durante la cursada de Programación I, desarrollando un sistema completo de gestión de países con Python.

A través de distintas funciones, se busca leer, modificar y analizar datos almacenados en un archivo CSV.

El sistema fue pensado para ser simple de usar, pero cumpliendo con buenas prácticas de programación, validaciones de entrada y sin utilizar bloques de excepciones (try/except), tal como fue indicado por la cátedra.

Marco Teórico

A continuación, se resumen los principales conceptos teóricos aplicados en el trabajo:

- **Listas:**
Son estructuras de datos que permiten almacenar múltiples elementos ordenados. Se usaron para mantener la colección de países, donde cada elemento es un diccionario con la información correspondiente.
- **Diccionarios:**
Son colecciones de pares “clave: valor”. Se utilizaron para guardar los datos de cada país (nombre, población, superficie, continente) de manera estructurada.
- **Funciones:**
Permiten dividir el programa en partes más pequeñas y reutilizables. Cada

funcionalidad (como agregar, buscar o actualizar países) se implementó en una función independiente para mantener el código ordenado.

- **Condicionales:**

Se emplearon para validar entradas del usuario y controlar el flujo lógico del programa. Gracias a ellos se evitan datos vacíos, valores negativos o formatos incorrectos.

- **Ordenamientos:**

El ordenamiento es un proceso que permite organizar los datos según un determinado criterio, por ejemplo de menor a mayor o alfabéticamente.

En este trabajo se aplicó para mostrar los países ordenados por población o por superficie, facilitando la comparación entre ellos.

Desarrollo Práctico

El sistema se divide en tres módulos principales:

- **funciones_basicas.py:**

Contiene las funciones de lectura, validación, agregado, búsqueda y actualización de los países.

- **funciones_avanzadas.py:**

Incluye funciones de análisis como ordenamientos, filtros y estadísticas.

- **main.py:**

Implementa el menú principal del programa, permitiendo al usuario interactuar con todas las funciones del sistema.

El archivo data_paises.csv almacena los datos iniciales y es modificado a medida que se agregan o actualizan países.

Pseudocódigo del Programa

```
INICIO
  Cargar archivo CSV
  MIENTRAS el programa esté en ejecución:
    Mostrar menú principal
    SEGÚN opción elegida:
      1 → Mostrar países
      2 → Agregar país
      3 → Actualizar país
      4 → Buscar país
      5 → Ordenar países
      6 → Filtrar por continente
      7 → Ver estadísticas
      8 → Salir
  FIN MIENTRAS
FIN
```

Validaciones Implementadas

- Control de que los campos no estén vacíos.
- Validación de que población y superficie sean enteros positivos.
- Rechazo de países duplicados.
- Validación del nombre (solo letras, espacios o guiones).
- Restricción de continentes a los seis válidos: América, Europa, Asia, África, Oceanía y Antártida.
- Mensajes claros para cada tipo de error o acción realizada.
- Carga y actualización automática del archivo CSV sin errores.

Resultados

Durante las pruebas se verificó que el programa cumple con todos los requerimientos solicitados.

Se pudo agregar, modificar y buscar países correctamente, manteniendo la integridad de los datos.

También se confirmaron los cálculos de promedio y el correcto funcionamiento de las funciones de ordenamiento y filtrado.

Conclusiones

El desarrollo de este trabajo nos permitió consolidar los conocimientos aprendidos sobre estructuras de datos, funciones, validaciones y modularización.

Además, fue una buena práctica para entender el trabajo en equipo y cómo se manejan archivos CSV en Python y cómo validar correctamente la información de entrada.

Aprendizaje Personal

Durante el desarrollo del TPI pudimos reforzar conceptos de validaciones, modularidad y estructuras de control.

Aprendimos la importancia de validar todos los datos que se ingresan y de mantener un código limpio y ordenado.

Otro aspecto para destacar es el trabajo en equipo. Aunque cada integrante tuvo su parte, aprendimos a coordinarnos, compartir ideas y ayudarnos mutuamente cuando surgían problemas. Esa colaboración fue clave para que el proyecto avance de forma ordenada y sin frustraciones.