

Trabajo Práctico Integrador - Programación I COM 4

Carpeta digital

Alumnos:

Godoy Lucas – Salcedo Ramiro

Tecnicatura Universitaria en Programación - Universidad Tecnológica Nacional

Docente Titular

Cinthia Rigoni

13 de noviembre de 2025

Tabla de contenido

Introducción	3
Informe teórico	4
Capturas de pantalla ejecución	8
Conclusión	10
Referencias	11

Introducción

El presente informe acompaña el desarrollo del Trabajo Práctico Integrador, cuyo objetivo central es diseñar una aplicación modular en Python para la gestión y análisis de información sobre países. A partir de un archivo de datos en formato CSV, el sistema permite realizar operaciones de búsqueda, filtrado, ordenamiento y obtención de estadísticas generales.

La propuesta implica la integración de los contenidos teóricos fundamentales abordados durante la cursada, entre ellos: el uso de listas y diccionarios, la modularización mediante funciones, la aplicación de estructuras condicionales y repetitivas, el manejo de archivos, los principios de validación de datos, y la implementación de técnicas básicas de análisis y ordenamiento. El trabajo busca no solo resolver un problema práctico de manipulación de datos, sino también promover buenas prácticas de programación, fomentando la claridad, la legibilidad y la organización del código.

Informe teórico

A continuación, se desarrollan los conceptos teóricos aplicados en la resolución del proyecto:

-Listas

Las listas son estructuras de datos dinámicas y ordenadas que permiten almacenar múltiples valores en una sola variable. Se utilizan para guardar la totalidad de los países cargados desde el archivo CSV.

Las listas facilitan:

- Recorrido secuencial con bucles
- Inserción de nuevos elementos
- Ordenamientos mediante `sorted()`
- Filtrado mediante `list comprehensions`

En este trabajo, cada elemento de la lista representa un país almacenado como un diccionario.

-Diccionarios

Los diccionarios permiten almacenar información estructurada mediante pares clave: valor.

Cada país se representa con un diccionario de la forma:

```
{  
    "nombre": "...",  
    "continente": "...",  
    "poblacion": int,  
    "superficie": int  
}
```

El uso de diccionarios permite acceder a cada dato de forma clara y directa, y facilita la aplicación de ordenamientos y estadísticas.

-Funciones y Modularización

La modularización se basa en dividir el programa en partes independientes, donde cada función tiene una única responsabilidad.

En el proyecto se implementaron funciones específicas para:

- Mostrar el menú
- Buscar un país
- Filtrar por continente
- Ordenar por diferentes criterios
- Calcular estadísticas
- Leer el archivo CSV
- Inicializar el archivo si no existe

La separación en módulos permite un código más legible, flexible y mantenible, favoreciendo la reutilización y la detección de errores.

-Estructuras condicionales

Las estructuras condicionales (`if`, `elif`, `else`) se utilizan para:

- Validar entradas del usuario
- Controlar errores
- Determinar criterios de ordenamiento
- Gestionar las opciones del menú

Estas decisiones permiten adaptar el flujo del programa a distintas situaciones y datos ingresados.

-Estructuras repetitivas

Los bucles `for` y `while` se aplican para:

- Recorrer la lista de países
- Filtrar y comparar valores
- Iterar múltiples opciones del menú
- Calcular estadísticas acumuladas

El uso de estas estructuras permite procesar conjuntos de datos de manera eficiente.

-Ordenamientos

El ordenamiento se realiza mediante la función:

```
sorted(lista, key=lambda x: x[criterio], reverse=orden)
```

Permite ordenar la lista de países por:

- nombre
- población
- superficie

Tanto en orden ascendente como descendente. El uso de `lambda` como función clave garantiza flexibilidad en la selección del criterio.

-Estadísticas

Para obtener indicadores del dataset se emplean:

- `max()` -> país con mayor población
- `min()` -> país con menor población
- `sum()` -> superficie y población total
- Diccionarios acumuladores -> cantidad de países por continente

Estos cálculos permiten analizar los datos de forma cuantitativa, cumpliendo con los requerimientos técnicos del trabajo.

-Archivos CSV

El archivo CSV es la fuente de datos principal. Cada fila contiene:

nombre,poblacion,superficie,continente

Se utiliza:

- `csv.DictReader` para convertir cada fila en un diccionario
- Codificación `utf-8-sig` para evitar problemas con acentos
- Manejo de errores para registros mal formateados
- Lectura previa a la ejecución del sistema
- Creación automática del archivo si no existe

El manejo adecuado del archivo garantiza la persistencia y correcta manipulación de los datos.

Capturas de pantalla de ejecución

A continuación, se encuentran capturas de pantalla de la ejecución del código brindado:

```
=====
GESTIÓN DE PAÍSES
=====
[1] Buscar país
[2] Filtrar países
[3] Ordenar países
[4] Estadísticas
[5] Salir

Elegí una opción --> |
```

Imagen N°1 - Captura del menú

```
=====
GESTIÓN DE PAÍSES
=====
[1] Buscar país
[2] Filtrar países
[3] Ordenar países
[4] Estadísticas
[5] Salir

Elegí una opción --> 1
Ingresá el nombre del país: argentina
País encontrado: {'nombre': 'Argentina', 'poblacion': 45376763, 'superficie': 2780400, 'continente': 'america'}
```

Imagen N°2 - Captura opción “Buscar país”

```
Elegí una opción --> 2
Ingresá el continente: america

Países en america:
-----
• Argentina
  Población: 45.376.763
  Superficie: 2.780.400 km²
-----
• Brasil
  Población: 213.993.437
  Superficie: 8.515.767 km²
-----
• Canadá
  Población: 38.008.005
  Superficie: 9.984.670 km²
```

Imagen N°3 - Captura opción “Filtrar países”

```
Elegí una opción --> 3
Criterio (nombre, poblacion, superficie): nombre
¿Querés orden descendente? (s/n): s

Países ordenados por nombre:
-----
Suecia: Suecia
Sudáfrica: Sudáfrica
Rusia: Rusia
Reino Unido: Reino Unido
Nueva Zelanda: Nueva Zelanda
Nigeria: Nigeria
Mexico: Mexico
Japon: Japon
Italia: Italia
```

Imagen N°4 - Captura opción “Ordenar países”

```
Elegí una opción --> 4
=====
ESTADÍSTICAS
=====

País con mayor población: China (1.411.778.724)
País con menor población: Nueva Zelanda (5.185.288)

Población promedio: 207.431.218.33
Superficie promedio: 3.413.208.75 km2

Cantidad por continente:
-----
america: 6
asia: 6
europa: 7
oceania: 2
africa: 3
```

Imagen N°5 - Captura opción “Estadísticas”

Conclusión

El desarrollo de este Trabajo Práctico Integrador permitió aplicar de manera articulada los diversos contenidos trabajados. A través de la implementación de un sistema modular y funcional, se logró comprender con mayor profundidad el uso de estructuras de datos, el manejo de archivos, el diseño de funciones y las técnicas básicas de análisis y ordenamiento.

El enfoque práctico del proyecto contribuyó a consolidar la lógica algorítmica, mejorar la capacidad de resolución de problemas y afianzar buenas prácticas de programación. Asimismo, el trabajo colaborativo entre ambos permitió ejercitarse la revisión conjunta del código, la toma de decisiones, la división de trabajo y la organización del proyecto.

El resultado final constituye una base sólida para abordar desarrollos más complejos en cursos posteriores de la Tecnicatura.

Referencias

Python Software Foundation. Python Documentation. <https://docs.python.org>

Tutorial oficial del módulo csv. <https://docs.python.org/3/library/csv.html>

Material didáctico de la cátedra Programación 1 – UTN