

Instituto Tecnológico y de Estudios Superiores de Monterrey

Escuela de Ingeniería

Maestría en Inteligencia Artificial Aplicada.

Pruebas de Software y aseguramiento de la calidad.



Actividad 5.2 Ejercicio de programación 2 y análisis estático.

David Elías Limas Palacios

A01560923

Fecha de entrega: 15 de febrero de 2026

Evidencia de ejecución y análisis estático

Análisis estático con Pylint

Se ejecutó el analizador estático pylint para verificar que el programa no presenta errores, advertencias ni problemas de calidad de código después de aplicar el estándar PEP-8 y corregir los detalles detectados.

- Comando ejecutado: *pylint computeSales.py*
- Resultado: El código fue calificado con 10.0/10, lo cual indica que cumple con las recomendaciones de calidad y buenas prácticas de programación en Python. El análisis confirmó que el sistema no presenta problemas estructurales, uso incorrecto de variables, errores potenciales de ejecución ni malas prácticas de diseño.

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> pylint computeSales.py

-----
Your code has been rated at 10.00/10 (previous run: 9.87/10, +0.13)

(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> 
```

Análisis con Flake8

Se ejecutó la herramienta flake8 para verificar el cumplimiento del estándar de codificación PEP-8. Esta herramienta permite detectar problemas de formato, espaciado, longitud de líneas y uso incorrecto de estructuras del lenguaje Python.

- Comando ejecutado: *flake8 computeSales.py*
- Resultado: No se reportaron errores ni advertencias, lo cual indica que el código cumple completamente con las reglas del estándar PEP-8.

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> python computeSales.py TC1.ProductList.json TC1.Sales.json
Compute Sales - Results
-----
Catalogue file: TC1.ProductList.json
Sales file: TC1.Sales.json

TOTAL SALES: 2481.86

Elapsed time (s): 0.0

(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> flake8 computeSales.py
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> 
```

Ejecución de casos de prueba

Se ejecutaron todos los casos de prueba proporcionados para validar la correcta funcionalidad del programa.

- Se utilizó el comando: *python computeSales.py TC1.ProductList.json TC[X].Sales.json*
 - Donde [X] corresponde al número de caso de prueba.
 - Este comando se ejecutó para **tres casos de prueba diferentes (TC1, TC2 y TC3)** proporcionados en los archivos de apoyo.
- Durante cada ejecución, el programa:
 - mostró los resultados en consola,
 - calculó el total de ventas,
 - indicó el tiempo de ejecución,
 - y generó automáticamente el archivo **SalesResults.txt**.
 - Adicionalmente, la salida en consola fue guardada en archivos de texto, permitiendo conservar los resultados mostrados por el programa durante la ejecución.
 - A continuación, se muestran las imágenes de la consola como evidencia de TC1, TC2 y TC3 respectivamente.

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> python computeSales.py TC1.ProductList.json TC1.Sales.json
Compute Sales - Results
-----
Catalogue file: TC1.ProductList.json
Sales file: TC1.Sales.json

TOTAL SALES: 2481.86

Elapsed time (s): 0.0
```

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> python computeSales.py TC1.ProductList.json TC2.Sales.json
Compute Sales - Results
-----
Catalogue file: TC1.ProductList.json
Sales file: TC2.Sales.json

TOTAL SALES: 166568.23

Elapsed time (s): 0.0
```

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 5.2 Ejercicios de Programacion 2\A5.2 Activities\A5.2 Archivos de Apoyo\TC1> python computeSales.py TC1.ProductList.json TC3.Sales.json
ERROR: Product 'Elotes' not found in catalogue (skipped)
ERROR: Product 'Frijoles' not found in catalogue (skipped)
Compute Sales - Results
-----
Catalogue file: TC1.ProductList.json
Sales file: TC3.Sales.json

TOTAL SALES: 165235.37

Elapsed time (s): 0.002000570297241211
```

Resultados

Se ejecutaron correctamente los tres programas desarrollados utilizando todos los archivos de prueba proporcionados por la materia. Para cada programa se corrieron los casos de prueba indicados y se generaron los reportes de salida correspondientes desde la consola.

Posteriormente, los resultados obtenidos por los programas fueron comparados directamente con los resultados esperados provistos en los archivos de apoyo de la actividad. En todos los casos, los valores calculados coincidieron con los resultados esperados, por lo que se determinó que los programas cumplen con los requerimientos funcionales establecidos y que los casos de prueba fueron aprobados satisfactoriamente.

Adicionalmente, se ejecutó el análisis estático mediante la herramienta pylint y se corrigieron los problemas detectados hasta no presentar errores ni advertencias de calidad de código.

En consecuencia, considerando tanto la correcta ejecución de los casos de prueba como la validación mediante análisis estático, se concluye que los programas funcionan adecuadamente y satisfacen los requisitos solicitados en la actividad.

Los reportes de casos de prueba consolidados para cada programa, se encuentran en el repositorio como evidencia.