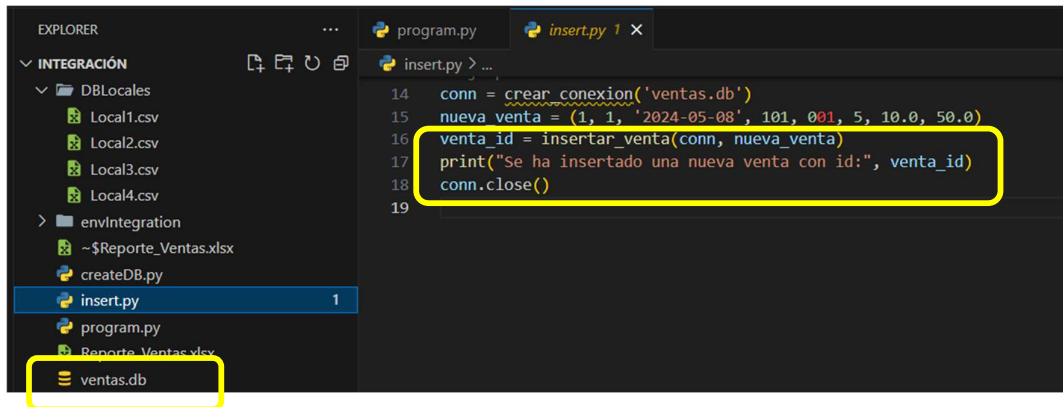


Según la problemática presentada se decidió crear un script en Python. Para comenzar se creó una base de datos en la herramienta de Sqlite3 que viene por defecto en el paquete de Python, se ubicó dentro del proyecto y se insertó un solo dato en la tabla de nombre 'ventas\_consolidadas':

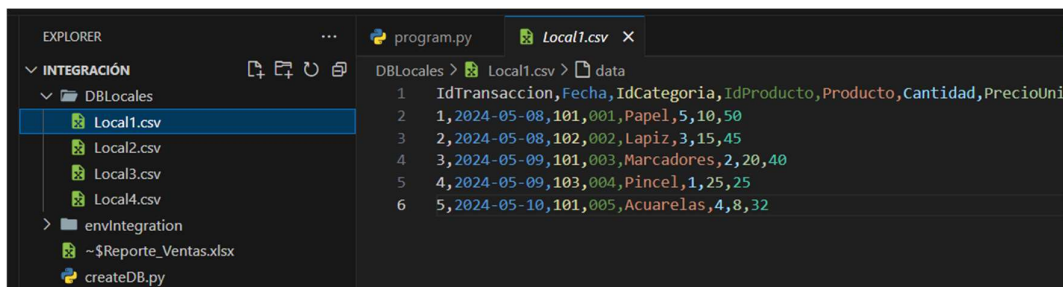


The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays a file tree under the 'INTEGRACIÓN' folder. The 'DBLocales' subfolder contains 'Local1.csv', 'Local2.csv', 'Local3.csv', and 'Local4.csv'. Below this, 'envIntegration' contains '\$Reporte\_Ventas.xlsx', 'createDB.py', 'insert.py', 'program.py', and 'Reporte\_Ventas.xlsx'. The 'ventas.db' file is highlighted with a yellow box. The main editor window shows the 'insert.py' file with the following code:

```
14 conn = crear_conexion('ventas.db')
15 nueva_venta = (1, 1, '2024-05-08', 101, 001, 5, 10.0, 50.0)
16 venta_id = insertar_venta(conn, nueva_venta)
17 print("Se ha insertado una nueva venta con id:", venta_id)
18 conn.close()
19
```

The code lines 15, 16, 17, and 18 are highlighted with a yellow box.

Luego se creó los respectivos archivos .csv, suponiendo que se han generado en el día actual de los 4 locales.



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left is the same as in the previous image, but 'Local1.csv' is now selected. The main editor window displays the content of 'Local1.csv' with the following data:

IdTransaccion	Fecha	IdCategoria	IdProducto	Producto	Cantidad	PrecioUnit
1	2024-05-08	101	001	Papel	5	10,50
2	2024-05-08	102	002	Lapiz	3	15,45
3	2024-05-09	101	003	Marcadores	2	20,40
4	2024-05-09	103	004	Pincel	1	25,25
5	2024-05-10	101	005	Acuarelas	4	8,32

Debido a que no se disponía de la herramienta para la visualización de los datos dentro de la base de datos, se agregó un método que imprimiera los datos ingresados en un archivo .xlsx, de nombre 'Reporte\_Ventas.xlsx'

The image shows a Visual Studio Code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'INTEGRACIÓN' with subfolders 'DBLocales' and 'envIntegration'. The 'envIntegration' folder contains files like '.gitignore', 'Reporte\_Ventas.xlsx', 'createDB.py', 'insert.py', 'program.py', and 'ventas.db'. The 'program.py' file is selected and its code is visible in the editor. The code defines a 'main()' function that connects to a SQLite database, exports data to an Excel file, and closes the connection. The terminal shows the output of running the script, including the installation of 'xlswriter' and the successful execution of the program.

```
28 def main():
43     conn = crear_conexion(ventas.db)
46
47     # Si no hay una tabla, crea una o reemplaza
48     df_consolidado.to_sql('ventas_consolidadas', conn, if_exists='repl
49
50     # Exportar a Excel
51     exportar_a_excel(conn)
52
53     # Cerrar la conexión a la base de datos
54     conn.close()
55
56 if __name__ == "__main__":
57     main()
58
```

File "C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración\envIntegration\Lib\site-packages\pandas\io\excel\xlswriter.py", line 197, in \_init\_ from xlswriter import Workbook  
ModuleNotFoundError: No module named 'xlswriter'  
(envIntegration) PS C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración> pip install xlswriter  
Collecting xlswriter  
• Using cached Xlswriter-3.2.0-py3-none-any.whl (159 kB)  
Installing collected packages: xlswriter  
Successfully installed xlswriter-3.2.0  
[notice] A new release of pip is available: 23.1.2 -> 24.0  
[notice] To update, run: C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración\envIntegration\Scripts\python.exe -m pip install --upgrade pip  
(envIntegration) PS C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración> python program.py  
• Conexión exitosa a SQLite.  
Datos exportados a Excel en 'Reporte\_Ventas.xlsx'.  
(envIntegration) PS C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración>

Este es el resultado en la primera prueba, 5 datos de cada Local y 1 que fue adicionado luego de crear la ase de datos:

	A	B	C	D	E	F	G	H	
1	Transacción	Fecha	IdCategoría	IdProducto	Cantidad	PrecioUnitario	TotalVenta	IdLocal	
2	1	2024-05-0	101	1	5	10	50	1	
3	2	2024-05-0	102	2	3	15	45	1	
4	3	2024-05-0	101	3	2	20	40	1	
5	4	2024-05-0	103	4	1	25	25	1	
6	5	2024-05-1	101	5	4	8	32	1	
7	1	2024-05-0	101	1	5	10	50	2	
8	2	2024-05-0	102	2	3	15	45	2	
9	3	2024-05-0	101	3	2	20	40	2	
10	4	2024-05-0	103	4	1	25	25	2	
11	5	2024-05-1	101	5	4	8	32	2	
12	1	2024-05-0	101	1	5	10	50	3	
13	2	2024-05-0	102	2	3	15	45	3	
14	3	2024-05-0	101	3	2	20	40	3	
15	4	2024-05-0	103	4	1	25	25	3	
16	5	2024-05-1	101	5	4	8	32	3	
17	1	2024-05-0	101	1	5	10	50	4	
18	2	2024-05-0	102	2	3	15	45	4	
19	3	2024-05-0	101	3	2	20	40	4	
20	4	2024-05-0	103	4	1	25	25	4	
21	5	2024-05-1	101	5	4	8	32	4	
22									
23									

Para ejecutarlo cada día a cierta hora, es recomendable una tarea programada como en IIS, sin embargo, en este ejercicio se colocará una variable de tiempo en la que se deberá ejecutar el script de actualización de los datos, no es eficiente puesto que el script estaría en constante ejecución, pero para la solución actual es útil hasta para revisar su funcionamiento:

The image shows a Visual Studio Code editor window with a Python script named `program.py` and its terminal output. The script is located in a folder named `INTEGRACIÓN`, which contains several CSV files and a subfolder `envIntegration`. The script `program.py` is highlighted in the file explorer.

```
56 conn.close()
57
58 def main():
59     while True:
60         try:
61             print("Iniciando el proceso de consolidación de datos...")
62             proceso_principal()
63             print("Proceso completado. Esperando hasta la próxima ejecución.")
64         except Exception as e:
65             print(f"Error durante la ejecución: {e}")
66
67         # 3600 segundos (1 hora) antes de la próxima ejecución
68         time.sleep(3600)
69
70 if __name__ == "__main__":
71     main()
72
```

The terminal output shows the execution of the script, which prints the following messages:

```
File "C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración\program.py", line 68, in main
time.sleep(3600)
KeyboardInterrupt
(envIntegration) PS C:\Users\fernando.lopez\OneDrive - PichinchaCorp\Documentos\Programas\Integración> python .\program.py
Iniciando el proceso de consolidación de datos...
Conexión exitosa a SQLite.
Datos exportados a Excel
Proceso completado. Esperando hasta la próxima ejecución.
```