Assignment – Data Processing & Deployment Pipeline

In this assignment, I prepared raw sales data and transformed it using Python.

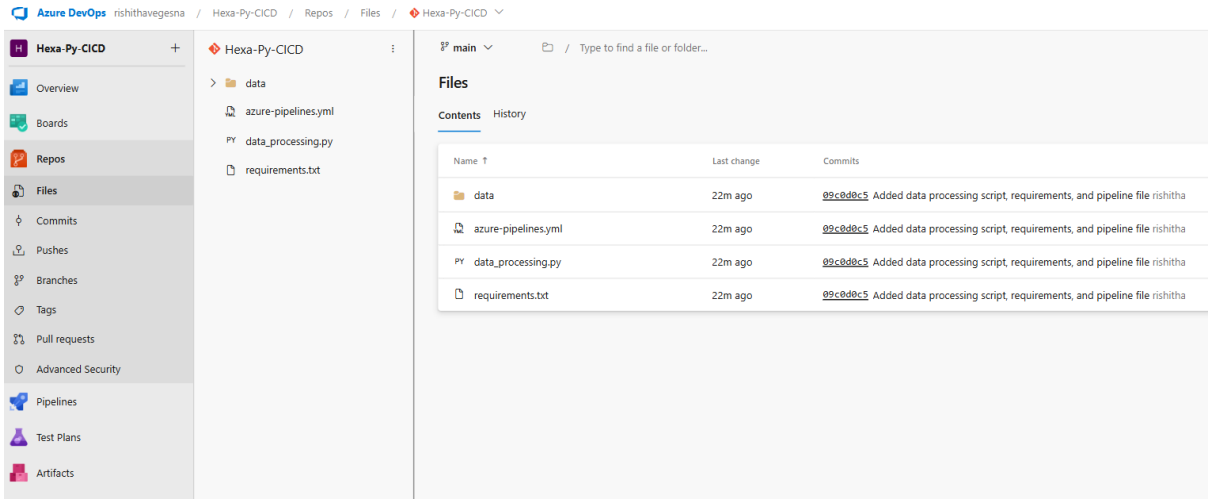
The raw data was cleaned by removing missing values and duplicates, formatting

dates into YYYY-MM-DD, and normalizing column names. Both raw and cleaned files

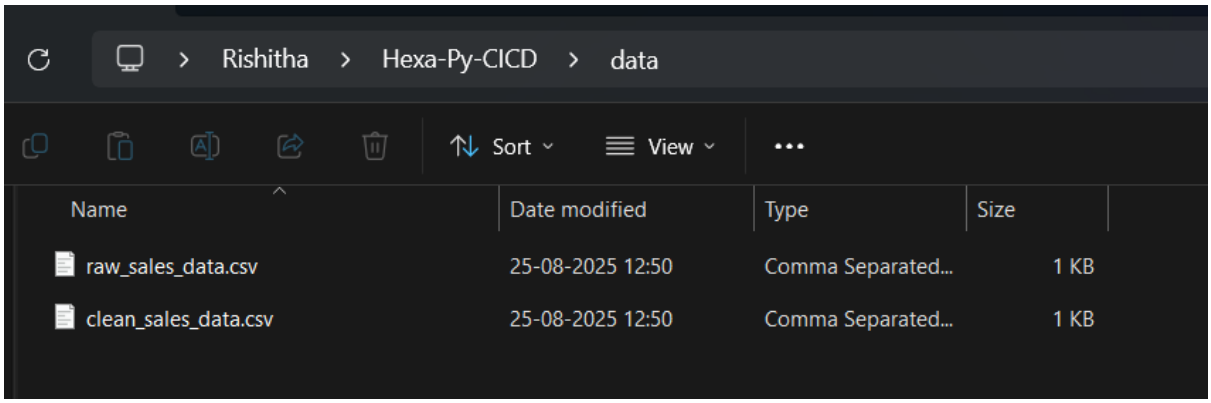
were then saved locally. An Azure DevOps repository and pipeline configuration

were also created to demonstrate how this process would run in a CI/CD workflow.

Azure DevOps Repo Structure-Files uploaded in Azure DevOps repository

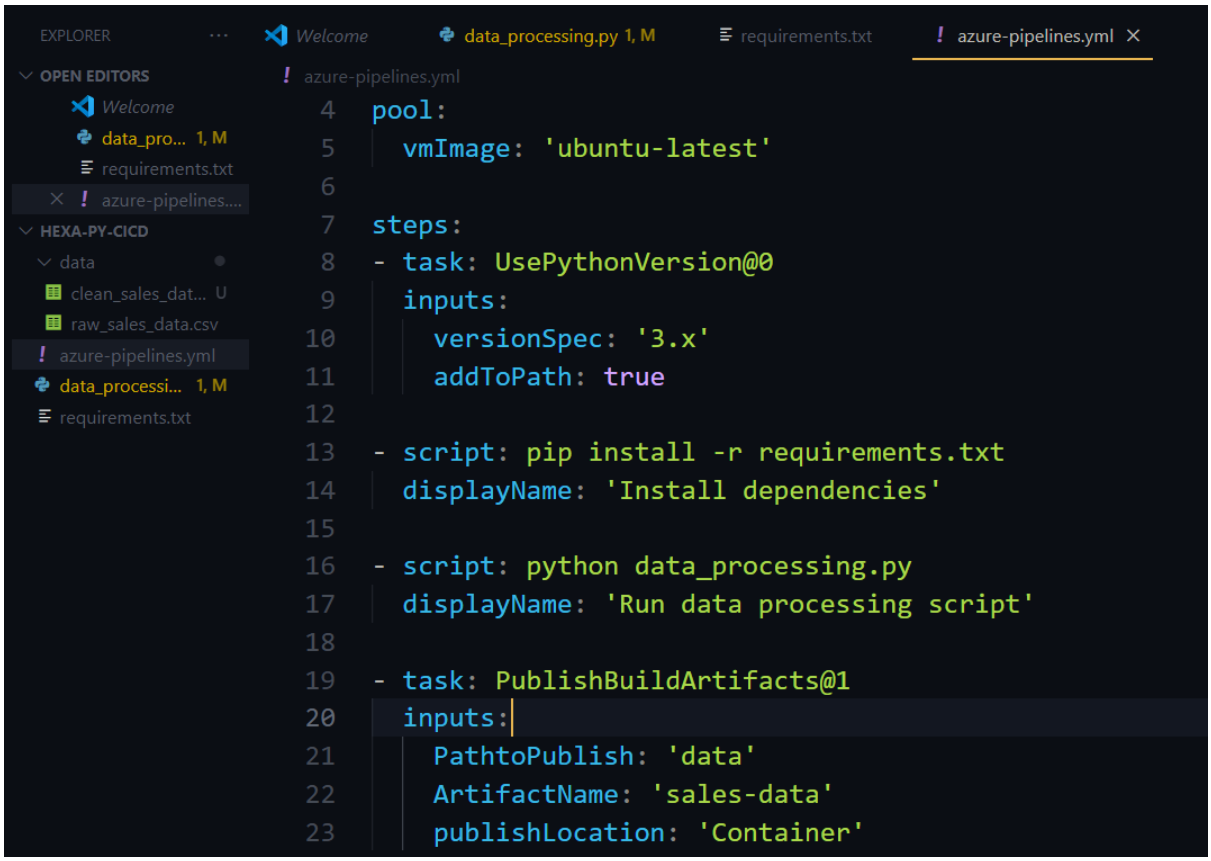


Local Data Folder-Local data folder containing raw and cleaned sales data



Running Script in VS Code Terminal-Execution of Python script generating cleaned data

azure-pipelines.yml in VS Code



Bonus Questions Section

Add answers in simple, clear language:

Q1: Why is data cleaning important in real-time data processing?

- Data cleaning ensures accuracy, removes duplicates, handles missing values, and keeps the data consistent. This makes analytics and decisions reliable in real-time systems.

Q2: What are pipeline artifacts and how are they used in DevOps workflows?

- Pipeline artifacts are output files generated by a pipeline (e.g., cleaned CSV). They are stored after the pipeline run and can be reused in later stages such as testing, deployment, or reporting.

Q3: How would you modify the pipeline to store the cleaned data into Azure Blob Storage?

- By adding an Azure task in the pipeline YAML that connects to Azure Storage using a service connection, and uploads clean_sales_data.csv from the data/ folder into a blob container.

CONCLUSION:

This assignment demonstrated the complete flow of preparing raw data, cleaning

it using Python, and setting up a CI/CD pipeline in Azure DevOps to manage

artifacts. The exercise helped understand the importance of data preparation

and automation in real-time data processing.