

Real-Time Credit Card Fraud Detection Pipeline on Azure

Problem Statement

In recent years, credit card fraud has become increasingly sophisticated, leveraging advanced techniques such as identity theft, phishing, and AI-driven fraudulent transactions. Traditional fraud detection methods often rely on static rule-based systems, which struggle to keep pace with real-time fraud patterns and large-scale data processing demands. As a result, financial institutions face significant monetary losses, reputational damage, and compromised customer trust.

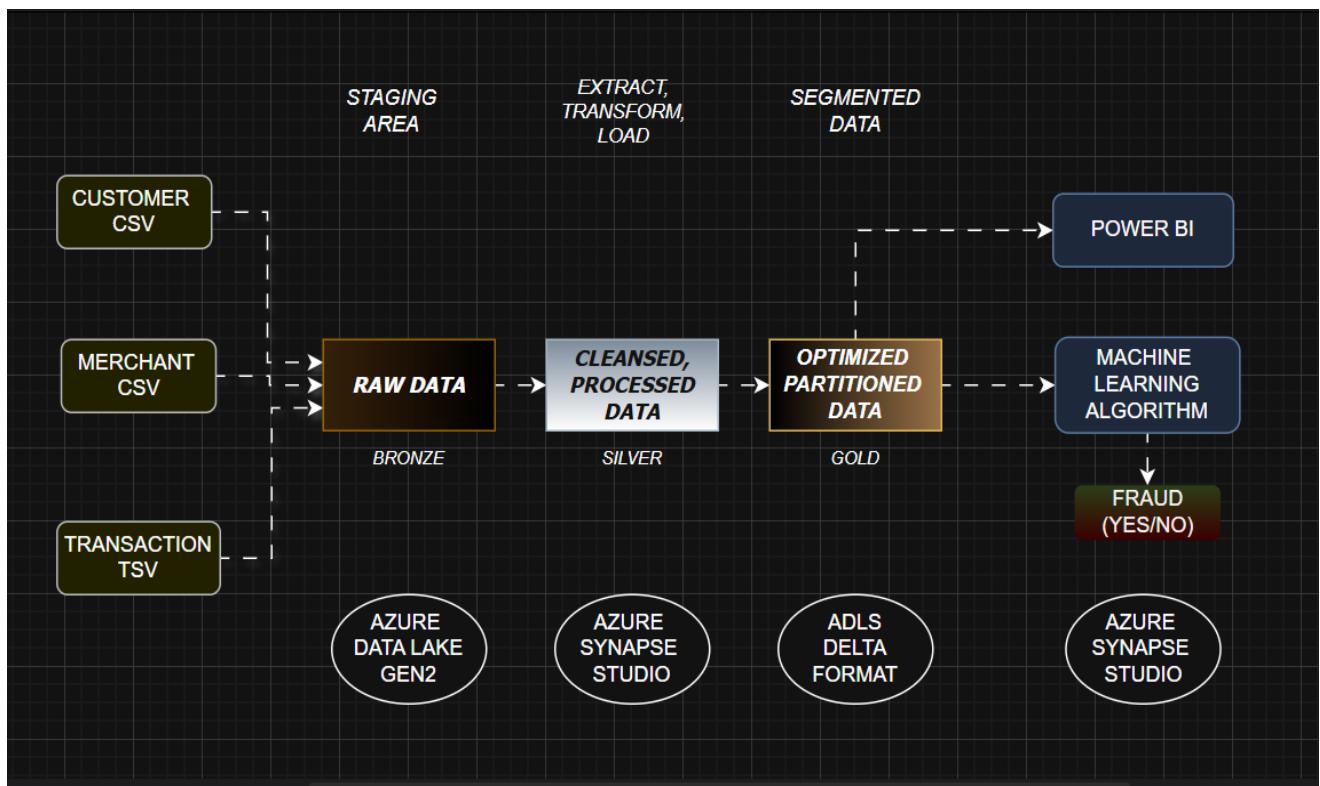
To effectively combat this growing threat, a leading bank seeks to develop a cloud-native, scalable, real-time fraud detection pipeline using Microsoft Azure. The system must be capable of processing massive volumes of transactional data, detecting anomalies instantly, and triggering fraud alerts for investigation, ensuring customer security and financial integrity.

Objectives

The goal of this project is to create a secure, scalable, and optimized data pipeline on Azure to support fraud detection and analysis. The key steps include:

- 1. Live Data Streaming & Static Data Ingestion** – Utilize Azure Event Hub to ingest real-time credit card transaction data. Stream incoming transactions into Azure Data Lake Storage Gen2 for processing and analysis.
- 2. Data Storage & Processing** – Load raw credit card transaction data (CSV & TSV) into Azure Data Lake Storage Gen2, clean it using PySpark in Azure Synapse Studio, and store optimized data in Parquet format for efficiency.
- 3. Data Optimization** – Improve data retrieval speed by applying bucketing and partitioning, then organize datasets into a gold-layer-data container for advanced analysis.
- 4. Fraud Classification using Machine Learning** – Implement a Logistic Regression model in Python within Synapse Studio to classify transactions as fraudulent or legitimate based on historical data.
- 5. Visualization & Reporting** – Connect Power BI to the optimized data layer to create interactive dashboards for fraud trend analysis and monitoring.

WORKFLOW:



Technical Stack

1. Data Storage & Processing

- Azure Data Lake Storage Gen2 (ADLS)
- Azure Synapse Studio(Pyspark , ML)

2. Security & Access Management

- Azure Key Vault

3. Visualization & Reporting

- Power BI

4. Live Data Streaming

- Azure event Hub

Dataset Description

This dataset was downloaded from **Kaggle** and contained structured data for **fraud detection** in credit card transactions. It was divided into **three datasets**, each serving a distinct purpose in the fraud detection pipeline.

1. Customer Dataset (15 Columns)

This dataset stored customer demographic details, transaction identifiers, and location-related information. It was used to analyse customer behaviour and assess fraud risk.

Column Name Description

trans_num Unique transaction ID associated with the customer.

cc_num Credit card number used for transactions.

gender Gender of the customer.

dob Date of birth of the customer.

job Occupation of the customer.

street Street address of the customer.

city City where the customer resided.

state State where the customer lived.

zip Zip code of the customer's location.

lat Latitude of the customer's location.

long Longitude of the customer's location.

city_pop Population of the city where the customer lived.

full_name Full name of the customer.

random_notes Additional metadata or random notes.

2. Merchant Dataset (6 Columns)

This dataset recorded merchant-related information linked to transactions. It was primarily used to track merchant categories and identify potential fraud hotspots.

Column Name Description

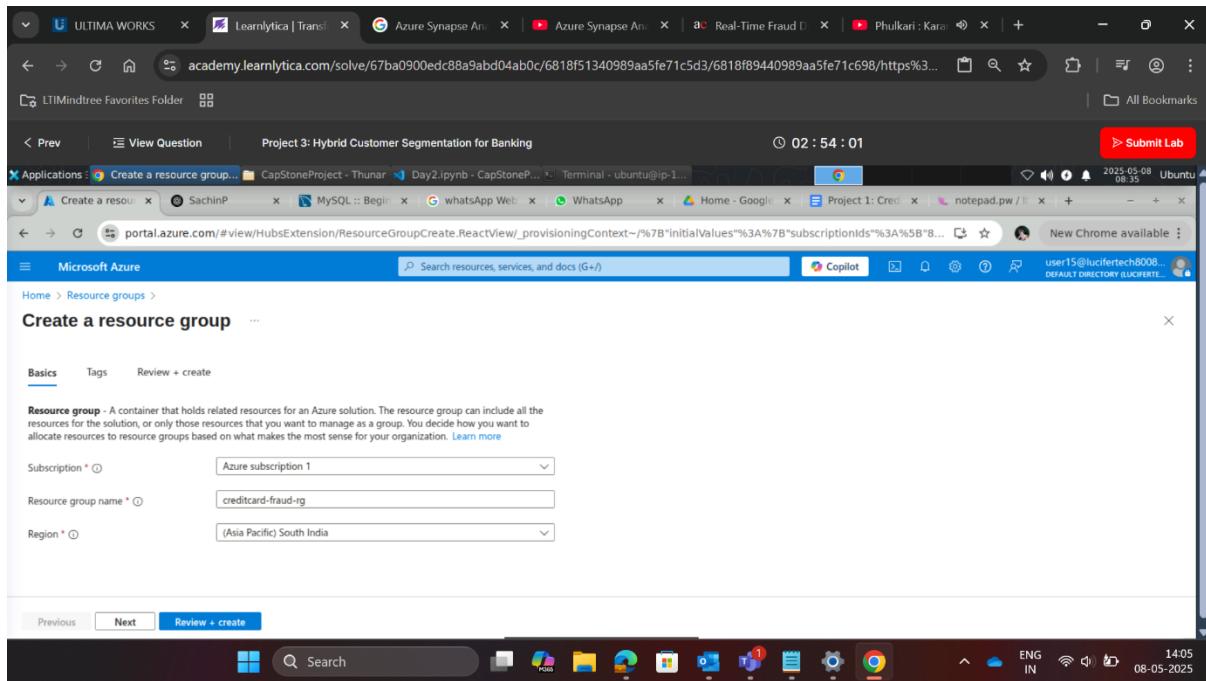
trans_num	Unique transaction ID linked to the merchant.
merchant	Name of the merchant involved in the transaction.
category	Business category of the merchant.
merch_lat	Latitude coordinate of the merchant's location.
merch_long	Longitude coordinate of the merchant's location.
merch_zipcode	Zip code of the merchant's business location.

3. Transactions Dataset (7 Columns)

This dataset stored individual **credit card transactions**, including timestamps, transaction amounts, and fraud labels. It was the core dataset for fraud classification using machine learning models.

Column Name	Description
trans_num	Unique transaction ID.
trans_date_trans_time	Timestamp of the transaction.
cc_num	Credit card number used for the transaction.
merchant	Merchant involved in the transaction.
amt	Amount spent in the transaction.
unix_time	Unix timestamp for tracking the transaction.
is_fraud	Fraud label (1 = Fraudulent, 0 = Legitimate).

Step 1:



- As part of the Azure infrastructure setup for the credit card fraud detection project, an Azure Resource Group named 'Creditcard-fraud-rg' was created. This resource group acts as a logical container for all related Azure resources, enabling better organization, access control, and cost management. All core services including Azure Synapse , Azure Data Lake Storage Gen2 and Power BI integrations are deployed within this resource group. This structure ensures a centralized and efficient management of all resources associated with the project.

Step 2:

The screenshot displays two side-by-side Azure Storage Account creation forms in a browser window.

Left Form (Standard Settings):

- Storage account name: ccfraudatalake
- Region: (Asia Pacific) South India
- Primary service: Azure Blob Storage or Azure Data Lake Storage Gen 2
- Performance: Standard (Recommended for most scenarios)
- Redundancy: Geo-redundant storage (GRS)

Right Form (Advanced Settings):

- Enable storage account key access: checked
- Default to Microsoft Entra authorization in the Azure portal: unchecked
- Minimum TLS version: Version 1.2
- Permitted scope for copy operations (preview): From any storage account
- Hierarchical Namespace:
Hierarchical namespace, complemented by Data Lake Storage Gen2 endpoint, enables file and directory semantics, accelerates big data analytics workloads, and enables access control lists (ACLs). [Learn more](#)
- Enable hierarchical namespace: unchecked
- Access protocols

Both forms include a "Review + create" button at the bottom.

- enable hierarchical namespace, which is currently unchecked.

The screenshot shows the Microsoft Azure portal interface. At the top, there's a blue header bar with the 'Microsoft Azure' logo, a search bar, and various navigation icons. Below the header, the URL 'user11@lucifertech8008... DEFAULT DIRECTORY (luciferte...' is visible. The main content area is titled 'Storage accounts > ccfrauddatalake'. The page displays the 'Essentials' section for the storage account, listing the following properties:

	:	
Resource group (...	:	creditcard-fraud-rg
Location	:	southindia
Subscription (move)	:	Azure subscription 1
Subscription ID	:	81c97be5-d6a6-4431-89a7-d743a0bfcb11
Disk state	:	Available
Performance	:	Standard
Replication	:	Locally-redundant storage (LRS)
Account kind	:	StorageV2 (general purpose v2)
Provisioning state	:	Succeeded
Created	:	5/8/2025, 9:03:51 AM

At the bottom left, there's a 'Tags (edit)' button and a link to 'Add tags'. On the right side of the essentials section, there's a 'JSON View' link.

- After setting up the Azure Resource Group 'Creditcard-fraud-rg', an Azure Data Lake Storage account was created to serve as the centralized storage layer for the project's datasets. This storage account is responsible for hosting all raw input data and intermediate processed files.

Step 3:

The screenshot shows a Microsoft Azure Storage container named "raw-fraud-data". The container overview page displays three CSV files:

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
MerchantLocationInfo.csv	5/8/2025, 8:46:53 AM	Hot (Inferred)		Block blob	122.59 MiB	Available
Modified_UserInfo.csv	5/8/2025, 8:47:08 AM	Hot (Inferred)		Block blob	210.82 MiB	Available
TransactionData.csv	5/8/2025, 8:47:21 AM	Hot (Inferred)		Block blob	139.32 MiB	Available

- displaying the contents of a container named "raw-fraud-data." The container includes three files: "MerchantLocationInfo.csv," "Modified_UserInfo.csv," and "TransactionData.csv." Each file has details such as the date modified, access tier, archive status, blob type, size, and lease state.

Step 4:

The screenshot shows the "Create Synapse workspace" page in the Microsoft Azure Marketplace. The user is configuring the workspace details, specifically selecting the resource group and managed resource group.

Subscription: Azure subscription 1

Resource group: creditcard-fraud-rg (Create new)

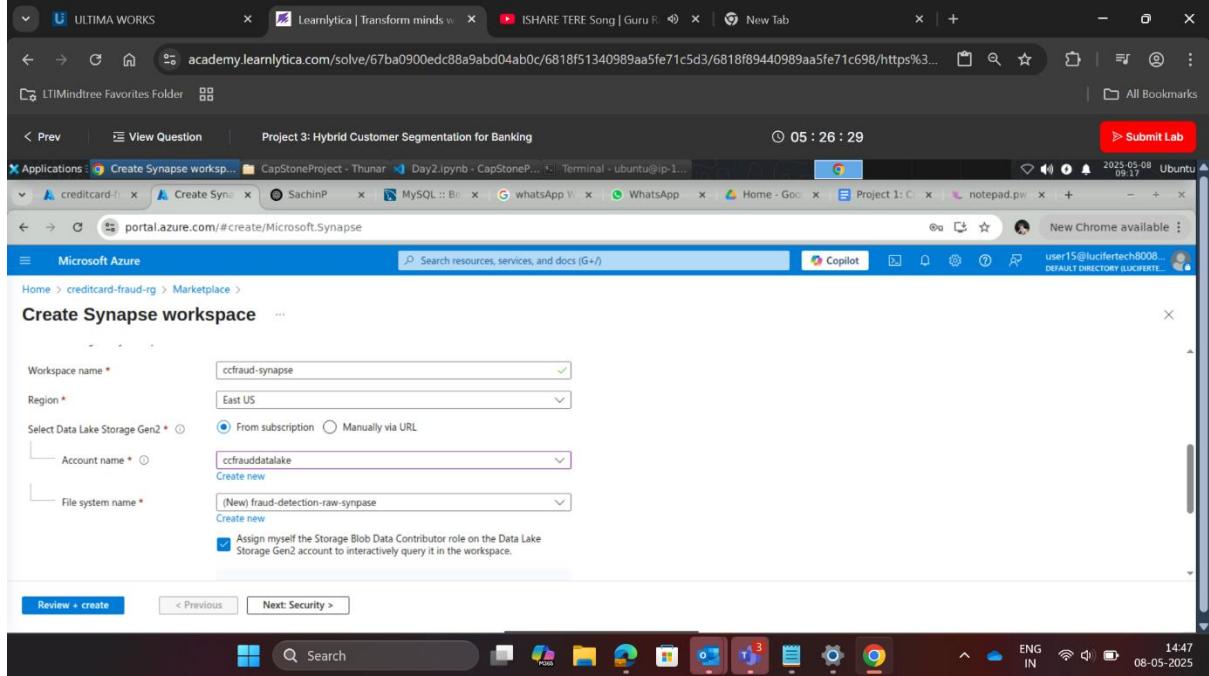
Managed resource group: synapse-ccfraud-managed-rg

Validation tooltip:

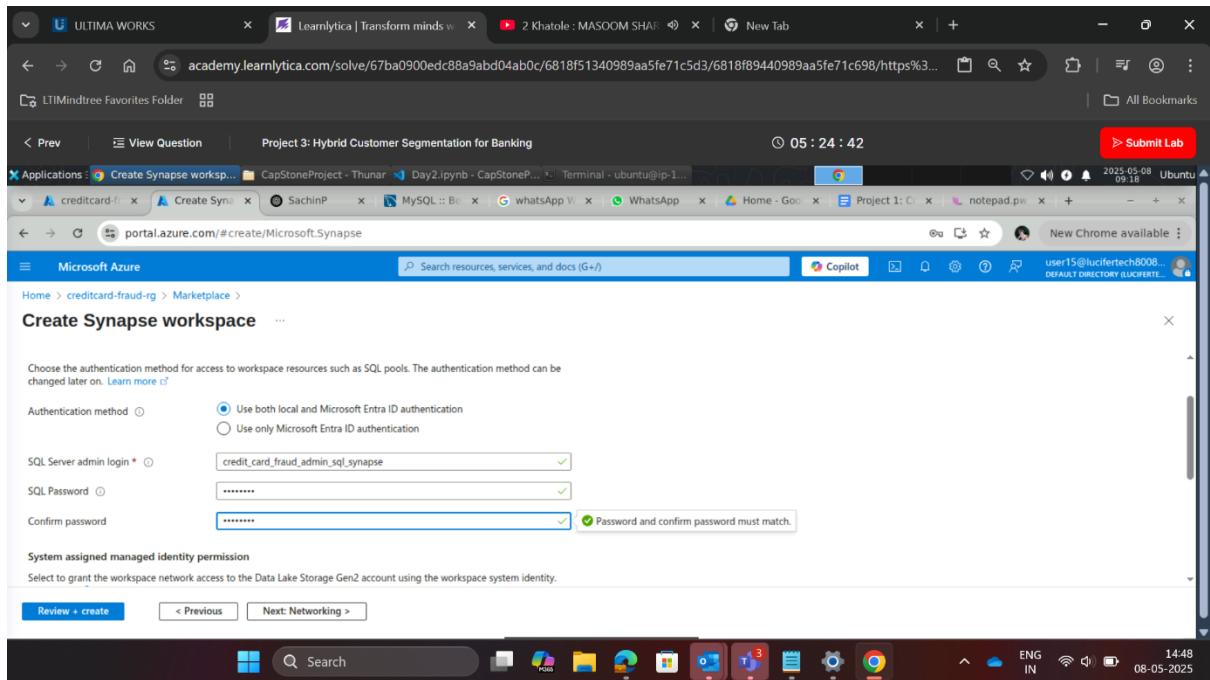
- Managed resource group names only allow up to 90 characters.
- Managed resource group names only allow alphanumeric characters, periods, underscores, hyphens and parenthesis and cannot end in a period.
- Managed resource group name must be unique in the selected subscription.

- The page includes options to select the subscription, resource group, and managed resource group for creating a Synapse workspace. The selected subscription is "Azure subscription 1," the resource group option is set to "Create new," and the managed resource group name entered is "synapse-cctfraud-managed-rg." There are also fields for entering a workspace name and selecting a primary Data Lake Storage Gen2 file system.

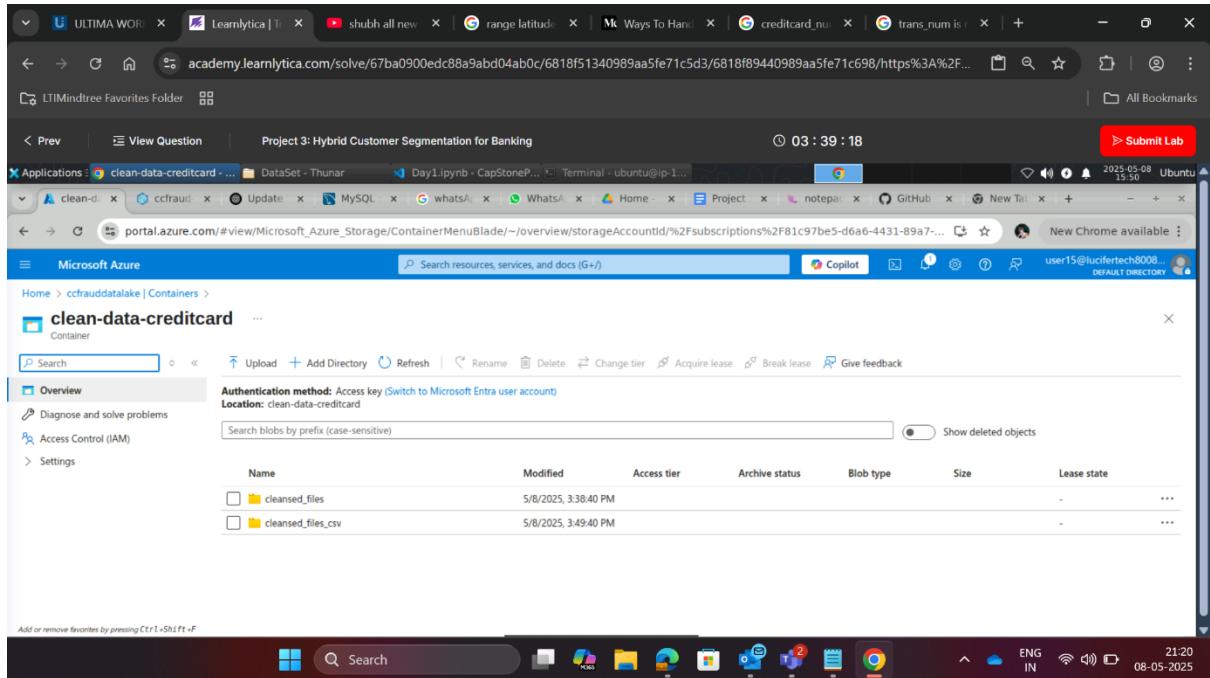
Step 5:



- **Workspace name:** "crtfraud-synapse" **Region:** "East US"
- **Select Data Lake Storage Gen2:** Options are "From subscription" and "Manually via URL"
- **Account name:** "crtfrauddatalake" with an option to create new **File system name:** New file system named "new-fraud-detection-raw-synapse"

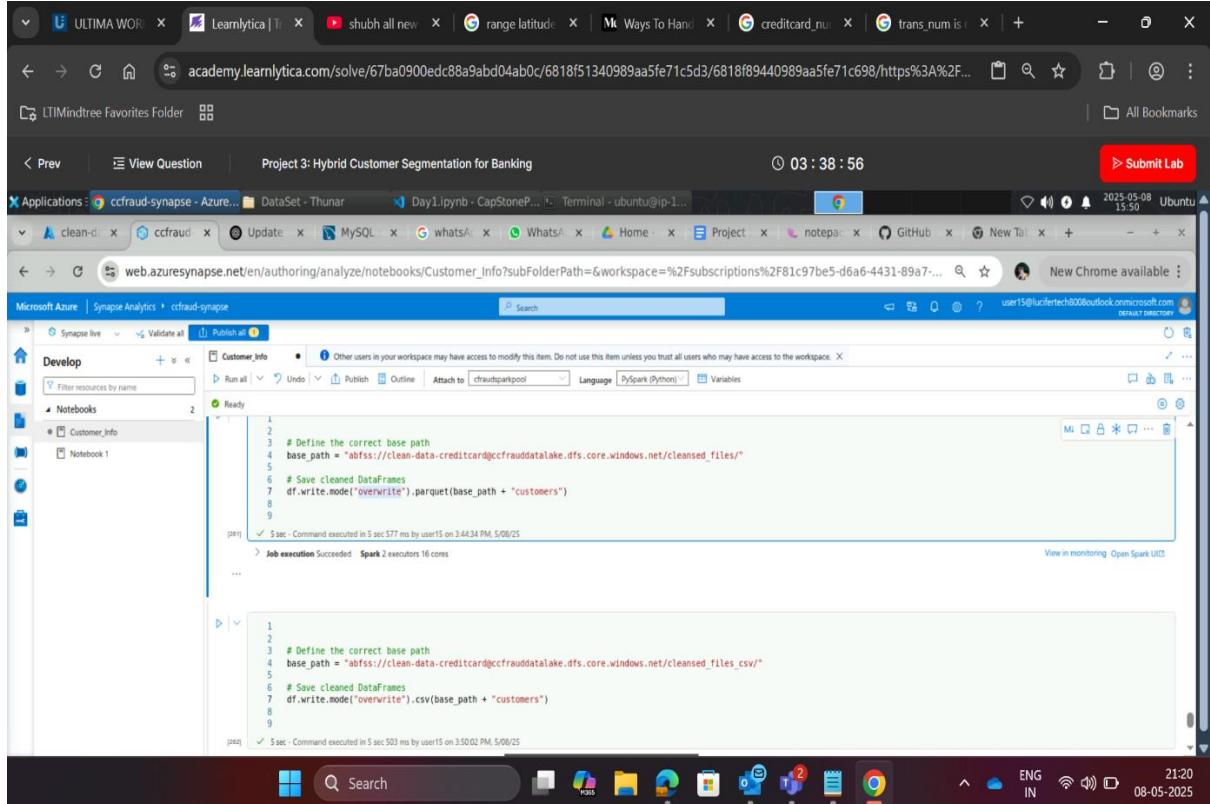


- It includes fields for setting up authentication and SQL server admin login details. The option for authentication method "Use both local and Microsoft Entra ID authentication" is enabled. The SQL Server admin login is set to "credit_card_fraud_admin_sql_synapse," with fields for entering and confirming the password.



- A container named "clean-data-creditcard" is created inside the ADLS for EDA process.
- The container has two files listed: "cleansed_files" and "cleansed_files.csv".

Step 6:



The screenshot shows a Microsoft Azure Synapse Analytics workspace interface. At the top, there's a browser tab for 'academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A%2F...'. Below it, a navigation bar includes 'View Question', 'Project 3: Hybrid Customer Segmentation for Banking', a timer showing '03 : 38 : 56', and a 'Submit Lab' button. The main area displays two notebooks in the 'Develop' section under the 'Customer_Info' folder. The top notebook has the following code:

```

1
2
3 # Define the correct base path
4 base_path = "abfss://clean-data-creditcard@ccfrauddatalake.dfs.core.windows.net/cleansed_files/"
5
6 # Save cleaned DataFrames
7 df.write.mode("overwrite").parquet(base_path + "customers")
8
9

```

The bottom notebook has similar code:

```

1
2
3 # Define the correct base path
4 base_path = "abfss://clean-data-creditcard@ccfrauddatalake.dfs.core.windows.net/cleansed_files_csv/"
5
6 # Save cleaned DataFrames
7 df.write.mode("overwrite").csv(base_path + "customers")
8
9

```

Both notebooks show a green status bar indicating successful execution. The bottom notebook's status bar also includes '5 sec - Command executed in 5 sec 577 ms by user15 on 3:44:34 PM, 5/08/25' and 'Job execution Succeeded - Spark 2 executors 16 cores'. The bottom right corner of the screen shows a Windows taskbar with various icons and the date/time '08-05-2025 21:20'.

Screenshot of a web browser showing a hybrid customer segmentation project for banking. The browser has multiple tabs open, including a Learnlytica question, a creditcard dataset, and a range latitude tool. The main content is a Microsoft Azure Synapse Analytics notebook titled "Customer_Info".

The notebook interface shows a sidebar with "Develop" and "Notebooks" sections, and a central workspace for running PySpark (Python) code. A command `df.show(2)` is run, displaying the first two rows of a DataFrame:

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| trans_num | cc_num|gender| dob| job| street| city|state| zip| lat| long|city_pop|firstName|lastName|age|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|43ec57a34709e1800...|6511349151460538| M|1946-08-24| Interpreter|Barner Point| Ruth| NV|89319|39.3426|-114.8859| 450| Robert| Nguyen| 78|
|cc80c5609c79326760...|372590258176518| F|1985-06-18|Learning disabili...| Andrea Glen|Goodrich| MZ|148438|42.9147| -83.4845| 6951| Kristen| Hanson| 39|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
```

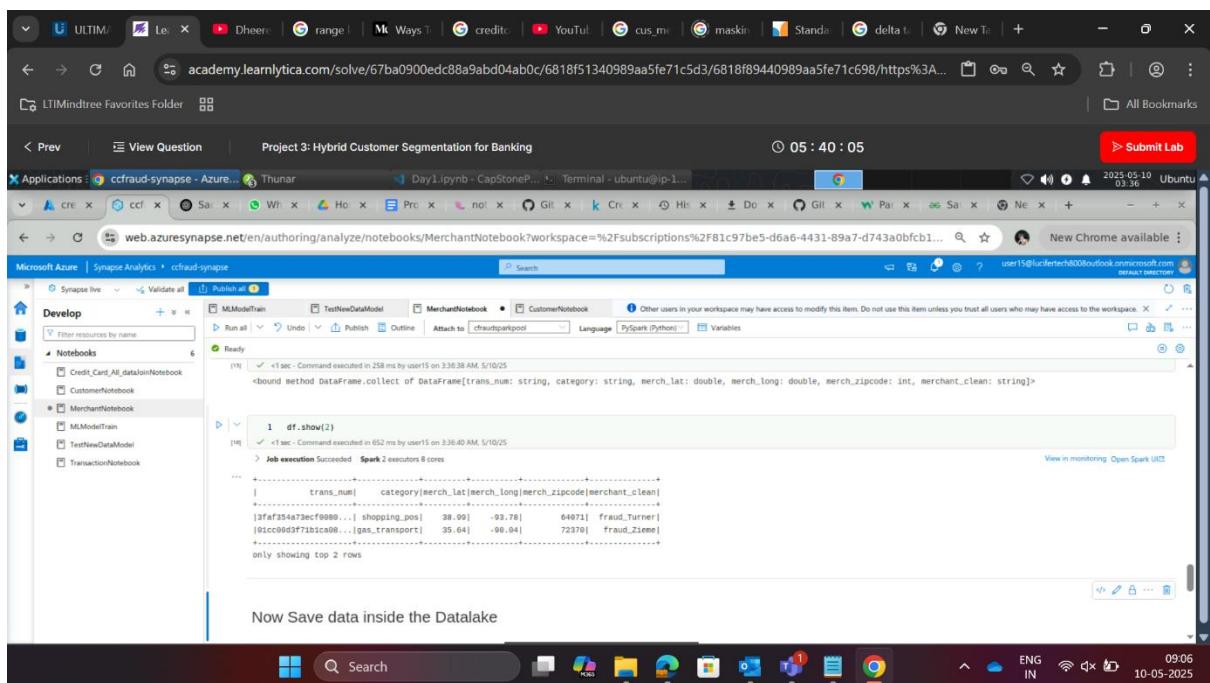
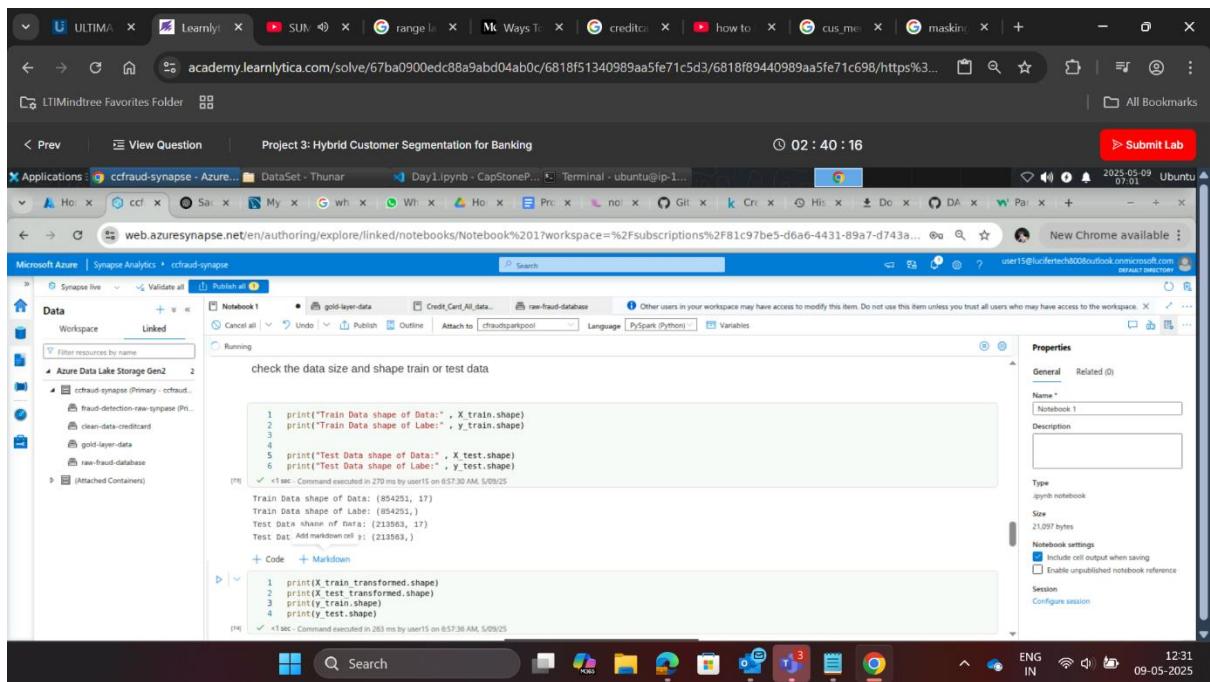
The notebook also includes a "Properties" panel on the right showing details like Name: Notebook 1, Type: ipynb notebook, and Size: 21,097 bytes.

Screenshot of a web browser showing the same hybrid customer segmentation project for banking. The browser has multiple tabs open, including a Learnlytica question, a creditcard dataset, and a range latitude tool. The main content is a Microsoft Azure Synapse Analytics notebook titled "Notebook 1".

The notebook interface shows a sidebar with "Data", "Workspace", and "Linked" sections, and a central workspace for running PySpark (Python) code. A command is run, showing the output:

```
from sklearn.linear_model import LogisticRegression
# Create logistic regression model
lr = LogisticRegression(penalty='none', solver='sag', max_iter=1000)
# Fit on training data
lr.fit(X_train_transformed, y_train)
```

The notebook also includes a "Properties" panel on the right showing details like Name: Notebook 1, Type: ipynb notebook, and Size: 21,097 bytes.



academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A... Submit Lab

05:39:43

Applications: ccfraud-synapse - Azure... Thunar Day1.ipynb - CapStoneP... Terminal - ubuntu@p-1...

web.azuresynapse.net/en/authoring/analyze/notebooks/MerchantNotebook?workspace=%2Fsubscriptions%2F81c97be5-d6a6-4431-89a7-d743a0bfc1... New Chrome available

Microsoft Azure | Synapse Analytics | ccfraud-synapse

Develop Validate all Publish all

Notbooks

- Credit_Card_All_datajoinNotebook
- CustomerNotebook
- MerchantNotebook
- MLModelTrain
- TestNewDataModel
- TransactionNotebook

Ready

```
1 from pyspark.sql.functions import col, sum
2
3 null_counts = df.select([sum(col(c).isNull()).cast("int")].alias(c) for c in df.columns)
4
5 null_counts.show()
6
```

29 sec - Command executed in 29 sec 444 ms by user15 on 3:36:12 AM, 5/10/25

> Job execution Succeeded. Spark 2 executors 8 cores

```
...+---+|trans_num|merchant|category|merch_lat|merch_long|merch_zipcode|
|     0|      0|    130931|       0|        0|      286883|
```

1 # df.select('merchant').collect()

<1 sec - Command executed in 270 ms by user15 on 3:36:12 AM, 5/10/25

View in monitoring Open Spark UI

09:07 IN 10-05-2025

academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A... Submit Lab

05:39:30

Applications: ccfraud-synapse - Azure... Thunar Day1.ipynb - CapStoneP... Terminal - ubuntu@p-1...

web.azuresynapse.net/en/authoring/analyze/notebooks/CustomerNotebook?workspace=%2Fsubscriptions%2F81c97be5-d6a6-4431-89a7-d743a0bfc1... New Chrome available

Microsoft Azure | Synapse Analytics | ccfraud-synapse

Develop Validate all Publish all

Notbooks

- Credit_Card_All_datajoinNotebook
- CustomerNotebook
- MerchantNotebook
- MLModelTrain
- TestNewDataModel
- TransactionNotebook

Not started

```
1 null_counts = df.select([sum(df[col].isnull().cast("int")).alias(c) for c in df.columns])
2
3 null_counts.show()
4
```

31 sec - Command executed in 31 sec 006 ms on 3:17:27 PM, 5/08/25

```
+---+|trans_num|cc_num|gender|dob|job|street|city|state|zip|lat|long|city_pop|full_name|random_notes|
|     0|      0|      0|130939| 0|130805| 0|130211| 0| 0|      0|      0|      0|
```

Find Out the Duplicate Row

```
1 from pyspark.sql.functions import col
2
3 duplicates = df.groupBy(df.columns).count().filter(col("count") > 1).agg(
4     count("count")
5 )
6 duplicates.show()
```

09:07 IN 10-05-2025

academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A... Submit Lab

Project 3: Hybrid Customer Segmentation for Banking 05 : 39 : 06

Applications: ccfrad-synapse - Azure... Thunar Day1.ipynb - CapStoneP... Terminal - ubuntu@ip-1...

web.azuresynapse.net/en/authoring/analyze/notebooks/transactionNotebook?workspace=%2Fsubscriptions%2F81c97be5-d6a6-4431-89a7-d743a0bfc... New Chrome available

Microsoft Azure | Synapse Analytics | ccfrad-synapse user15@laptoptech001:~\$

Develop Validate all

Notebooks

- Credit_Card_All_datajoinNotebook
- CustomerNotebook
- MerchantNotebook
- MLModelTrain
- TestNewDataModel
- TransactionNotebook

MLModelTrain TestNewDataModel MerchantNotebook TransactionNotebook Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all Outline Attach to chfraudsparkpool Language PySpark (Python) Variables

Duplicate Check

+ Code + Markdown

```
1 from pyspark.sql.functions import col, count, when
2
3 df.select([count(when(col(c).isNotNull(), c)).alias(c) for c in df.columns]).show()
```

(1) ✓ - Command executed in 2 sec 53 ms on 3:37:11 AM, 5/09/25

trans_num	trans_date_trans_time	cc_num	amt	unix_time	is_fraud
0	0	0	0	0	0

Check the duplicate

Search 09:07

ENG IN 10-05-2025

academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A... Submit Lab

Project 3: Hybrid Customer Segmentation for Banking 05 : 38 : 46

Applications: ccfrad-synapse - Azure... Thunar Day1.ipynb - CapStoneP... Terminal - ubuntu@ip-1...

web.azuresynapse.net/en/authoring/analyze/notebooks/transactionNotebook?workspace=%2Fsubscriptions%2F81c97be5-d6a6-4431-89a7-d743a0bfc... New Chrome available

Microsoft Azure | Synapse Analytics | ccfrad-synapse user15@laptoptech001:~\$

Develop Validate all

Notebooks

- Credit_Card_All_datajoinNotebook
- CustomerNotebook
- MerchantNotebook
- MLModelTrain
- TestNewDataModel
- TransactionNotebook

MLModelTrain TestNewDataModel MerchantNotebook TransactionNotebook Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all Outline Attach to chfraudsparkpool Language PySpark (Python) Variables

14 .otherwise(col("amt"))
15)
16
(1) ✓ - Command executed in 3 sec 29 ms on 3:37:18 AM, 5/09/25

+ Code + Markdown

```
1
```

df.show()

(1) ✓ - Command executed in 2 sec 39 ms on 3:37:20 AM, 5/09/25

trans_num	trans_date_trans_time	cc_num	amt	unix_time	is_fraud	amt_capped
[2fb89733c7106995cd...]	2020-03-09 23:25:59]	3708774992126014	3.59	[1302871559]	0	8.99
[e6799372e5a5628...]	2020-05-30 07:42:05]	3502460573145241	0.50	[1369699725]	0	6.76
[b47399e7e7cb329e94...]	2019-12-25 16:48:48]	4581368699336675	6.94	[1255507329]	0	6.94
[2f18dc80d250d56e0b...]	2019-04-14 01:58:01]	620451534492	135.66	[1334568681]	0	115.66

Search 09:08

ENG IN 10-05-2025

academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A... Submit Lab

05 : 37 : 49

Applications : ccfraud-synapse - Azure... Thunar Day1.ipynb - CapStonePr... Terminal - ubuntu@p-1...

web.azuresynapse.net/en/authoring/analyze/notebooks/Credit_Card_All_dataJoinNotebook?workspace=%2F81c97be5-d6a6-4431-89a... New Chrome available

Microsoft Azure | Synapse Analytics | ccfraud-synapse

Develop | Validate all

Not started

Empty Markdown cell. Double click or press enter to add content.

```
1
1 cus_mer_tran df.schema
[SQL] ✓ Command executed in 265 ms on 11:58:40 AM, 5/9/25
--- StructType[StructField('trans_num', StringType(), True), StructField('gender', StringType(), True), StructField('dob', DateType(), True), StructField('job', StringType(), True), StructField('street', StringType(), True), StructField('city', StringType(), True), StructField('state', StringType(), True), StructField('zip', IntegerType(), True), StructField('lat', DoubleType(), True), StructField('long', DoubleType(), True), StructField('city_pop', IntegerType(), True), StructField('firstname', StringType(), True), StructField('lastname', StringType(), True), StructField('merchant', StringType(), True), StructField('merchant_id', StringType(), True), StructField('trans_date', StringType(), True), StructField('trans_time', TimestampType(), True), StructField('amt', DoubleType(), True), StructField('merch_zipcode', IntegerType(), True), StructField('merchant_clean', StringType(), True), StructField('is_fraud', IntegerType(), True), StructField('cc_num_masked', StringType(), True)]
```

Save the Data Into the delta table

09:09 IN 10-05-2025

Creating another container

academy.learnlytica.com/solve/67ba0900edc88a9abd04ab0c/6818f51340989aa5fe71c5d3/6818f89440989aa5fe71c698/https%3A... Submit Lab

02 : 18 : 23

Applications : ccfrauddatalake - Microsoft Edge Downloads - Thunar final.ipynb - CapStonePr... Terminal - ubuntu@p-1...

ccfraud | portal.azure.com/#@lucifertech800outlook.onmicrosoft.com/resource/subscriptions/81c97be5-d6a6-4431-89a7-d743a0bfcbb1/resourceGroups/c...

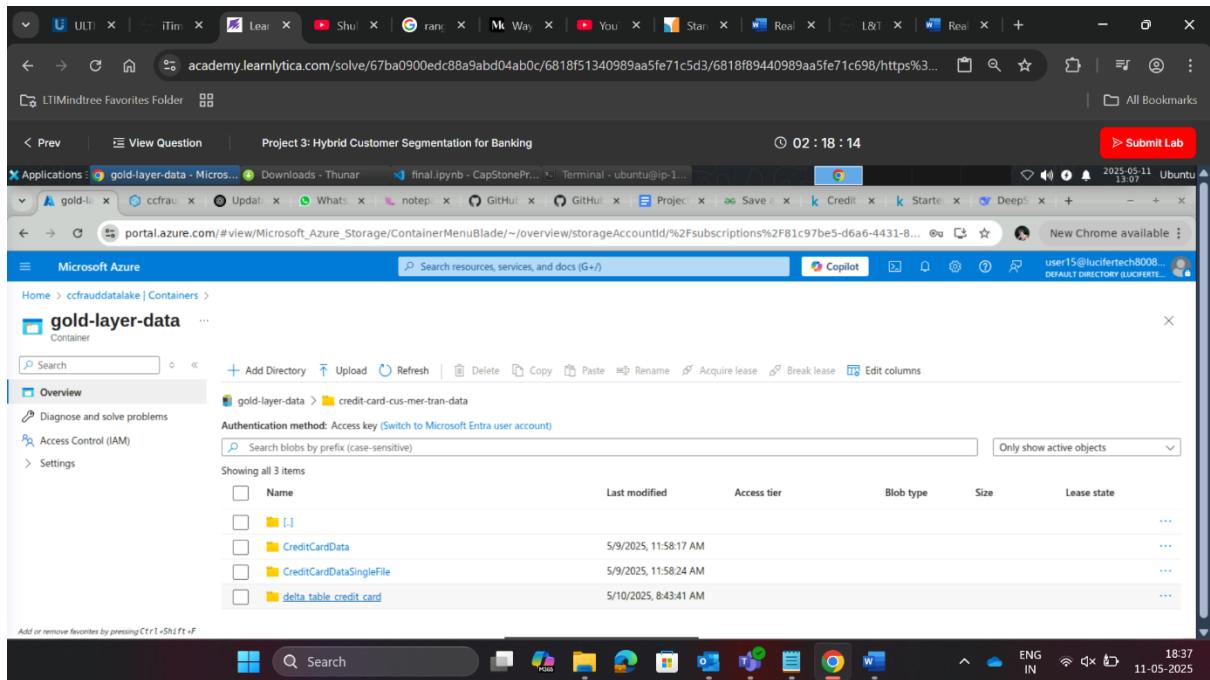
Microsoft Azure

ccfrauddatalake | Containers

Storage account

Name	Last modified	Anonymous access level	Lease state
Logs	5/8/2025, 9:04:19 AM	Private	Available
clean-data-creditcard	5/8/2025, 3:26:29 PM	Private	Available
eventhub-streaming-input	5/10/2025, 6:05:53 AM	Private	Available
fraud-detection-raw-synapse	5/8/2025, 9:19:27 AM	Private	Available
gold-layer-data	5/9/2025, 3:57:41 AM	Private	Available
model-file	5/9/2025, 9:00:35 AM	Private	Available
raw-fraud-database	5/8/2025, 9:06:16 AM	Private	Available

18:37 IN 11-05-2025



➤ The "gold-layer-data" container within the adls contains crucial datasets for the credit card fraud detection project.

➤ It has three folders

- CreditCardData
- CreditCardDataSingleFile
- delta_table_credit_card

The screenshot shows a Microsoft Azure Storage Container named 'gold-layer-data'. The container listing shows four blobs: '_delta_log', 'city-Achille', 'city=Acworth', and 'city=Adams'. The blobs were last modified on 5/10/2025 at 8:46:57 AM. The browser has multiple tabs open, including 'ULTI', 'iTIm', 'Learn', 'Shu', 'ran...', 'Way', 'You', 'Star', 'Real', 'L&T', 'Real', and a 'Submit Lab' button.

Code:

1. Customer EDA

```

1 customer_df = customer_df.drop('random_notes')
[254] ✓ - Command executed in 271 ms on 9:00:52 PM, 5/08/25

1 customer_df.show(2)
[257] ✓ - Command executed in 1 sec 332 ms on 9:01:01 PM, 5/08/25
+-----+
| trans_num|cc_num|gender|dob|job|street|city|state|zip|lat|long|city.pop|firstname|lastname|age|
+-----+
|43ec57a34399418b...|6511349151485438|M|1946-09-24|Interpreter|Garner Point|Ruth|NV|89319|39|3426|-114.8859|458|Robert|Nguyen|78|
|c88e5669c793267e9...|372569258176518|F|1985-06-18|Learning disabled...|Andrea Glen|Goodrich|MI|44438|42.9147|-83.4845|6551|Kristen|Hanson|39|
+-----+
only showing top 2 rows

1 customer_df.write.mode("overwrite").parquet("abfss://clean-data@ccfrauddatalake.dfs.core.windows.net/credit-card/cleaned_data")
[258] ✓ - Command executed in 1 sec 332 ms on 9:01:01 PM, 5/08/25

1 # Define the correct base path
2 base_path = "abfss://clean-data-creditcard@ccfrauddatalake.dfs.core.windows.net/cleansed_files/"
[259] ✓ - Command executed in 1 sec 332 ms on 9:01:01 PM, 5/08/25

```

The screenshot shows a Microsoft Azure Synapse Analytics workspace titled 'Customer_EDA'. It displays a PySpark notebook with the following code:

- Reading a CSV file into a DataFrame: `customer_df = customer_df.drop('random_notes')`
- Displaying the first two rows of the DataFrame: `customer_df.show(2)`
- Writing the DataFrame back to a Parquet file: `customer_df.write.mode("overwrite").parquet("abfss://clean-data@ccfrauddatalake.dfs.core.windows.net/credit-card/cleaned_data")`
- Defining a base path for cleaning data: `base_path = "abfss://clean-data-creditcard@ccfrauddatalake.dfs.core.windows.net/cleansed_files/"`

ULTIM/ Shubh Mk Ways T Learnly pandas what a New To histogr Credit ccl Credit New To +

LTIMindtree Favorites Folder

Microsoft Azure | Synapse Analytics > cfraud-synapse

We use optional cookies to provide a better experience. Learn more

Synapse live Validate all Publish all

Develop Filter resources by name

Notebooks BusinessQueryCreditCard CreditCardFraudData_EDA Customer_EDA Merchant_EDA MLModelTrain Partition_credit_card_data TestNewDataModel Transaction_EDA

BusinessQueryCredit... CreditCardFraudData... Customer_EDA

Run all Undo Publish Outline Attach to fraudsparkpool Language PySpark (Python) Variables

Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Not started

```
1 from pyspark.sql.functions import col, sum
2
3 null_counts = customer_df.select([sum(col(c).isNull().cast("int")).alias(c) for c in customer_df.columns])
4 null_counts.show()
5
```

[216] ✓ - Command executed in 31 sec 606 ms on 8:47:27 PM, 5/08/25

```
6
+-----+
|trans_num|cc_num|gender|dob| job|street| city|state| zip|lat|long|city_pop|full_name|random_notes|
+-----+
|      0|     0|     0| 0|13039| 0|13035| 0|13021| 0|     0|     0|     0|     0|
```

Finding total number of duplicate rows

```
1 from pyspark.sql.functions import col
```

Search ENG IN 09:25 12-05-2025

ULTIM/ Shubh Mk Ways T Learnly pandas what a New To histogr Credit ccl Credit New To +

LTIMindtree Favorites Folder

Microsoft Azure | Synapse Analytics > cfraud-synapse

We use optional cookies to provide a better experience. Learn more

Synapse live Validate all Publish all

Develop Filter resources by name

Notebooks BusinessQueryCreditCard CreditCardFraudData_EDA Customer_EDA Merchant_EDA MLModelTrain Partition_credit_card_data TestNewDataModel Transaction_EDA

BusinessQueryCredit... CreditCardFraudData... Customer_EDA

Run all Undo Publish Outline Attach to fraudsparkpool Language PySpark (Python) Variables

Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Not started

```
3 duplicates = customer_df.groupBy(customer_df.columns).count().filter(col("count") > 1).agg(
4   count("*")
5 )
6 duplicates.show()
7
```

[217] ✓ - Command executed in 4 sec 143 ms on 8:47:31 PM, 5/08/25

```
8
+-----+
|count(1)|
+-----+
| 1623|
```

FINDING THE OUTLIERS

NO OUTLIERS WERE FOUND (ALL VALUES WITHIN THE RANGE OF -90 TO 90)

```
1 customer_df.select(min(col('lat'))).show()
```

[218] ✓ - Command executed in 1 sec 263 ms on 8:47:32 PM, 5/08/25

```
2
+-----+
|min(lat)|
+-----+
| 28.0271|
```

Search ENG IN 09:25 12-05-2025

Customer_EDA

Dropping the fullname column

```
1 customer_df = customer_df.drop("full_name")
```

[237] ✓ - Command executed in 265 ms on 8:50:12 PM, 5/08/25

```
1 customer_df.show(2)
```

[238] ✓ - Command executed in 1 sec 999 ms on 8:50:15 PM, 5/08/25

trans_num	cc_hun[gender]	dob	job	street	city state	zip	lat	long city_pop[random_notes firstname lastname]
43e5c7a34709e1bb... 6511349151495438	M 1946-08-24	Interpreter 74835 Garner Point	Ruth	IV 89319, 0 39.3426 -114.8859	458 unknown Robert Nguyen			
c80e586f79320768... 37258925817618	F 1985-06-18 Learning disabiliti...	26544 Andrea Glen Goodrich	M 48438, 0 42.9147 -81.4845	6951 n/a Kristen Hanson				

only showing top 2 rows

Counting the total number of unique Credit Card numbers

```
1 # Unique card number of this customerinfo table inside
```

[239] ?

2. Merchant EDA

The screenshot shows a Microsoft Azure Synapse Analytics workspace titled 'Merchant_EDA'. The left sidebar lists notebooks: BusinessQueryCreditCard, CreditCardFraudData_EDA, Customer_EDA, Merchant_EDA (selected), MUModelTrain, Partition_credit_card_data, TestNewDataModel, and Transaction_EDA. The main area displays code snippets and their execution results. The first snippet converts a float column to an integer:

```
1 merchant_df = merchant_df.withColumn("merch_zipcode", col("merch_zipcode").cast(IntegerType()))
```

The second snippet shows the resulting DataFrame:

```
1 merchant_df.show(2)
```

trans_num	merchant	category	merch_lat	merch_long	merch_zipcode
3faef354d73ecf0800... fraud_Turner and ... shopping_pos 38.09 -93.78 64871	01cc0d3f7fb1ca0e... fraud_Zilme, Bode... gas_transport 35.64 -90.04 72370				

only showing top 2 rows

At the bottom, a search bar and a taskbar with various icons are visible.

Finding the Missing or Null value

```
1 from pyspark.sql.functions import col, sum
2
3 null_counts = merchant_df.select([sum(col(c).isNull().cast("int")).alias(c) for c in merchant_df.columns])
4
5 null_counts.show()
```

[8] ✓ - Command executed in 29 sec 444 ms on 9:06:12 AM, 5/10/25

	merch_id	merch_long	merch_zipcode
1	0	0	13031
2	0	0	200883

Adding a new column called fraud_merchant

```
1 from pyspark.sql.functions import regexp_extract
2
3 merchant_df = merchant_df.withColumn("fraud_merchant", regexp_extract(col("merchant"), r"(fraud_[A-Za-z]+)", 1))
```

[9] ✓ - Command executed in 2 sec 25 ms on 9:06:36 AM, 5/10/25

Drop the null values

```
1 merchant_df = merchant_df.filter(col("merch_zipcode").isNotNull())
2
```

[10] ✓ - Command executed in 256 ms on 9:06:37 AM, 5/10/25

```

1 merchant_df.collect
[1] ✓ - Command executed in 258 ms on 9:06:38 AM, 5/10/25
<bound method DataFrame.collect of DataFrame[trans_num: string, category: string, merch_lat: double, merch_long: double, merch_zipcode: int, merchant_clean: string]>

1 merchant_df.show(2)
[1] ✓ - Command executed in 652 ms on 9:06:40 AM, 5/10/25
+-----+-----+-----+-----+
| trans_num | category|merch_lat|merch_long|merch_zipcode|merchant_clean|
+-----+-----+-----+-----+
| 3fa135a73ecf080... | shopping_pos | 38.99 | -93.78 | 64071 | fraud_Turner |
| 01cc00d3f71b1ca0... | gas_transport | 35.64 | -98.04 | 72370 | fraud_Zieme |
+-----+-----+-----+-----+
only showing top 2 rows

Saving the cleaned data into adls

```

3.Transaction EDA

```

1 transaction_df.show()
[1] ✓ - Command executed in 2 sec 39 ms on 9:07:20 AM, 5/10/25
+-----+-----+-----+-----+-----+-----+-----+
| trans_num | trans_date_trans_time | cc_num | amt | unix_tx_time | is_fraud | amt_capped |
+-----+-----+-----+-----+-----+-----+-----+
| 2fbfa0733c71d695cd... | 2028-03-09 23:25:59 | 3788774952112014 | 8.59 | 1362871599 | 0 | 8.59 |
| e1d39372e52aa5a562b... | 2028-05-30 07:42:05 | 3506246577314524 | 6.76 | 1360899725 | 0 | 6.76 |
| b67722932881bd4bd... | 2019-12-11 11:27:57 | 4342532437704183 | 67.58 | 1355225277 | 0 | 67.58 |
| 5d159e9e7cb329e94... | 2019-12-15 18:48:49 | 4561368699336875 | 6.94 | 1355597329 | 0 | 6.94 |
| 2a1010a0a0a0a0a0a0... | 2019-06-05 05:15:45 | 4659875464594515... | 115.45 | 1338697181 | 0 | 115.45 |
| 64f7fobase... | 2019-06-05 05:15:42 | 4659875464594516... | 115.45 | 1338697221 | 0 | 67.00 |
| 6398ffcc1b7e517922... | 2019-12-13 02:06:16 | 365606864640617 | [115.86] | 1355364576 | 0 | 115.86 |
| ad411dd444ba0be158091... | 2019-02-13 08:18:08 | 40960847418182 | [6.02] | 1339122288 | 0 | 56.92 |
| 7ee25e7d6ab4719bc... | 2019-01-13 02:15:15 | 4155921259183879 | 87.71 | 1326423315 | 0 | 87.71 |
| c71f979c20916ab3b... | 2019-04-22 22:39:11 | 1418183325558613886 | 8.33 | 1335134531 | 0 | 8.33 |
| ab669da2883119773... | 2019-05-07 01:27:43 | 43024752164408909 | 3.71 | 1336354063 | 0 | 3.71 |
| 78c337fd6d2b6fe... | 2019-08-25 22:45:00 | 3548075240003197 | [131.34] | 1345934790 | 0 | 131.34 |
| 13b5107980f05b4... | 2019-07-29 21:43:54 | 30235438713380 | 3.02 | 1343598234 | 0 | 3.02 |
| 0dd94a0817845587... | 2019-02-05 12:21:10 | 4092452671390169678 | 59.25 | 1328444470 | 0 | 59.25 |
| e9cd5d40970681127... | 2019-08-02 15:27:13 | 630451534402 | [110.15] | 1343921233 | 0 | 110.15 |
| e5a91519dc6181fe69d... | 2019-10-06 04:39:43 | 501802953650 | [174.37] | 1340498383 | 0 | 174.37 |
| f7e84adeefeb7134... | 2019-10-01 07:41:58 | 372246459334925 | [108.55] | 1340877438 | 0 | 108.55 |
| 1ae91bc9e9897095fb... | 2019-10-28 20:16:33 | 45956282999005111019 | 7.24 | 1351455393 | 0 | 7.24 |
| 5ff974801a4d6894... | 2020-06-11 23:00:37 | 3459339645607467 | 2.63 | 1370991997 | 0 | 2.63 |
| cce3a59bb0fa2f8a7... | 2019-12-07 23:01:30 | 180048185037117 | 42.46 | 1354921298 | 0 | 42.46 |

```

Microsoft Azure | Synapse Analytics > ccfraud-synapse

We use optional cookies to provide a better experience. Learn more ↗

Synapse live Validate all Publish all

Develop Filter resources by name

Notebooks BusinessQueryCreditCard CreditCardFraudData_EDA Customer_EDA Merchant_EDA Transaction_EDA

BusinessQueryCredit... CreditCardFraudData... Customer_EDA Merchant_EDA Transaction_EDA

Run all Undo Publish Outline Attach to /fraudsparkpool Language PySpark (Python) Variables

Not started

```
1 from pyspark.sql.types import *
2 from pyspark.sql.functions import *
3 from pyspark.sql import SparkSession
4 from pyspark.sql.functions import col, count
5
6
7 [1] ✓ - Apache Spark session started in 45 sec: 301 ms. Command executed in 301 ms on 9:07:00 AM, 5/09/25
```

Reading the csv file from adls

```
1 transaction_df = spark.read.csv("abfss://raw-fraud-database@ccfrauddatalake.dfs.core.windows.net/TransactionData.csv", inferSchema=True , header=True ,sep="\t")
2 transaction_df.show()
```

[1] ✓ - Command executed in 9 sec 9 ms on 9:07:09 AM, 5/09/25

trans_num	trans_date_trans_time	cc_num	merchant	amt	unix_time	is_fraud
1d8bdcbc194ddc696...	2019-07-27 21:00:29	577588686219	Fraud_Kassulke PLC	110.39	1343422829	0
0315af3a79861431...	2019-02-27 13:25:28	6011366578560244	fraud_Mueller, Ge...	10.87	1330349128	0
b2a2a9066bf0f1818...	2019-12-13 17:14:45	213112662687660	fraud_Little-Gle...	3.15	1355418885	0
fb3ba5f056815bce3...	2020-06-01 03:00:47	2131191402330821	fraud_Friesen-Stam...	97.26	1370055647	0

only showing top 4 rows

ENG IN 09:30 12-05-2025

Microsoft Azure | Synapse Analytics > ccfraud-synapse

We use optional cookies to provide a better experience. Learn more ↗

Synapse live Validate all Publish all

Develop Filter resources by name

Notebooks BusinessQueryCreditCard CreditCardFraudData_EDA Customer_EDA Merchant_EDA Transaction_EDA

BusinessQueryCredit... CreditCardFraudData... Customer_EDA Merchant_EDA Transaction_EDA

Run all Undo Publish Outline Attach to /fraudsparkpool Language PySpark (Python) Variables

Not started

Counting null values per column

```
1 from pyspark.sql.functions import col, count, when
2
3 transaction_df.select([count(when(col(c).isNull(), c)).alias(c) for c in transaction_df.columns]).show()
```

[1] ✓ - Command executed in 2 sec 53 ms on 9:07:11 AM, 5/09/25

trans_num	trans_date_trans_time	cc_num	merchant	amt	unix_time	is_fraud
0	0	0	0	0	0	0

Counting the number of duplicates

```
1 duplicate = transaction_df.groupBy(transaction_df.columns).count().filter(col("count") > 1).count()
2 print(f"Number of duplicate rows: {duplicate}")
3
```

[1] ✓ - Command executed in 2 sec 995 ms on 9:07:14 AM, 5/09/25

Number of duplicate rows: 6466

ENG IN 09:30 12-05-2025

```

1 from pyspark.sql.functions import col, expr, percentile_approx
2
3 # Step 1: Compute Q1, Q3 and IQR
4 q1, q3 = df.approxQuantile("amt", [0.25, 0.75], 0.01)
5 iqr = q3 - q1
6 lower_bound = q1 - 1.5 * iqr
7 upper_bound = q3 + 1.5 * iqr
8
9 #Fix the outlier cap
10
11 df = df.withColumn("amt_capped",
12     when(col("amt") < lower_bound, lower_bound)
13     .when(col("amt") > upper_bound, upper_bound)
14     .otherwise(col("amt"))
15 )
16

```

✓ - Command executed in 3 sec 29 ms on 9:07:18 AM, 5/09/25

```

1 transaction_df.show()

```

✓ - Command executed in 2 sec 39 ms on 9:07:20 AM, 5/09/25

trans_num	trans_time	cc_num	amt	unix_time	is_fraud	amt_capped
1	2025-05-09T09:07:18Z	1234567890123456	100.0	1625540438	0	100.0
2	2025-05-09T09:07:20Z	1234567890123456	150.0	1625540440	0	150.0

Optimization:

```

1 from pyspark.sql.functions import concat, lit, col
2
3 cus_mer_tran_df = cus_mer_tran_df.withColumn(
4     "cc_num_masked",
5     concat(lit("*****"), col("cc_num").substr(-4, 4))
6 )
7

```

✓ - Command executed in 287 ms on 5:20:23 PM, 5/09/25

...
Code Markdown

Dropping the previous column named cc_num

```

1 cus_mer_tran_df = cus_mer_tran_df.drop("cc_num")
2 cus_mer_tran_df = cus_mer_tran_df.drop("amt_corrected")
3 cus_mer_tran_df.schema

```

09:33 12-05-2025

Removing the duplicate column credit card number mentioned as cc_num

```
1 tran_df = tran_df.drop("cc_num")  
[210] ✓ - Command executed in 253 ms on 5:19:20 PM, 5/09/25
```

Joined previously obtained dataframe with transactions on trans_num

```
1 cus_mer_tran_df = cust_merch_df.join(tran_df , on="trans_num" , how="inner")  
[211] ✓ - Command executed in 258 ms on 5:19:20 PM, 5/09/25
```

Count of records after joining

Removing the duplicate column credit card number mentioned as cc_num

```
1 tran_df = tran_df.drop("cc_num")  
[210] ✓ - Command executed in 253 ms on 5:19:20 PM, 5/09/25
```

Joined previously obtained dataframe with transactions on trans_num

```
1 cus_mer_tran_df = cust_merch_df.join(tran_df , on="trans_num" , how="inner")  
[211] ✓ - Command executed in 258 ms on 5:19:20 PM, 5/09/25
```

Count of records after joining

```

1 cus_mer_tran_df.schema
2
[21] ✓ - Command executed in 316 ms on 5:19:26 PM, 5/09/25
StructType([StructField('trans_num', StringType(), True), StructField('cc_num', StringType(), True), StructField('gender', StringType(), True), StructField('dob', DateType(), True), StructField('job', StringType(), True), StructField('street', StringType(), True), StructField('city', StringType(), True), StructField('state', StringType(), True), StructField('zip', IntegerType(), True), StructField('lat', DoubleType(), True), StructField('city_pop', IntegerType(), True), StructField('firstname', StringType(), True), StructField('lastname', StringType(), True), StructField('merch_lat', DoubleType(), True), StructField('merch_long', DoubleType(), True), StructField('merch_zipcode', IntegerType(), True), StructField('merchant_clean', StringType(), True), StructField('amt', DoubleType(), True), StructField('unix_time', LongType(), True), StructField('is_fraud', IntegerType(), True), StructField('amt_corrected', DoubleType(), True)])

```

Count of total number of columns

```

1 # cus_mer_tran_df = cus_mer_tran_df.drop("cc_num")
2 print("This is Final of Column:",len(cus_mer_tran_df.schema))
3
[214] ✓ - Command executed in 276 ms on 5:19:26 PM, 5/09/25
This is Final of Column: 17

```

Partition And delta table optimization:

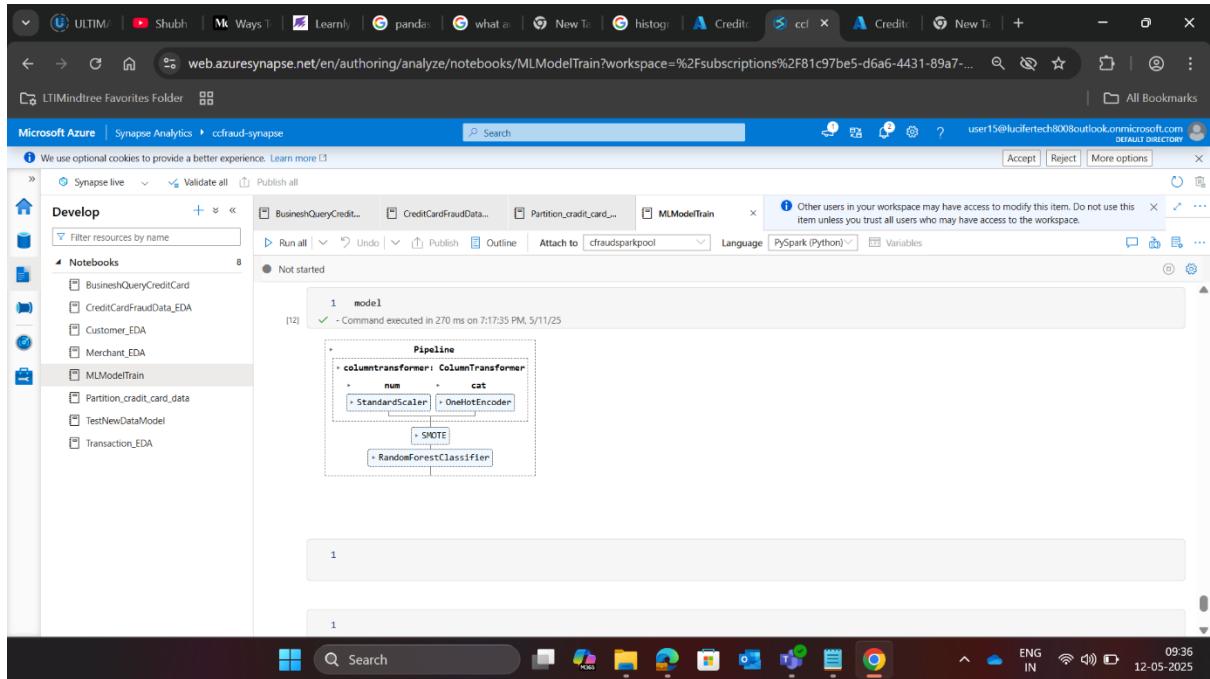
```

1 # credit_card_df = spark.read.csv("", inferSchema=True , schema=True)
2 credit_card_df = spark.read.csv("abfss://gold-layer-data@ccfrauddatalake.dfs.core.windows.net/credit-card-cus-mer-tran-data/CreditCardDataSingleFile/part-00000-095856e8-d777-4333-8...
3
[1] ✓ - Command executed in 2 sec 16 ms on 2:10:10 PM, 5/10/25
...
1 credit_card_df
2
[11] ✓ - Command executed in 275 ms on 2:15:42 PM, 5/10/25
DataFrame(trans_num: string, gender: string, dob: date, job: string, street: string, city: string, state: string, zip: int, lat: double, long: double, city_pop: int, firstname: string, lastname: string, age: int, category: string, merch_lat: double, merch_long: double, merch_zipcode: int, merchant_clean: string, trans_date_trans_time: timestamp, amt: double, unix_time: int, is_fraud: int, cc_num_masked: string)

1 credit_card_df.write.format("delta") \
2 .mode("overwrite") \
3 .partitionBy("city") \
4 .save("abfss://gold-layer-data@ccfrauddatalake.dfs.core.windows.net/credit-card-cus-mer-tran-data/delta_table_credit_card/")
5
[14] ✓ - Command executed in 19 sec 258 ms on 2:17:05 PM, 5/10/25

```

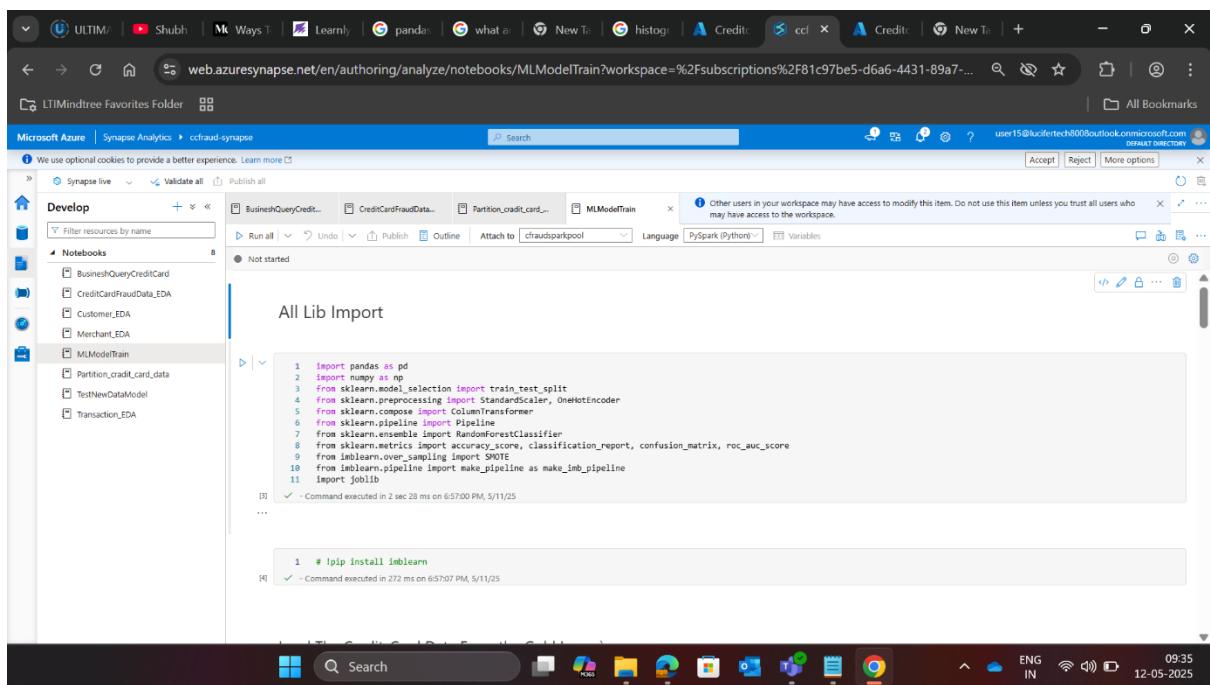
Machine Learning to detect Fraudsters:



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar lists notebooks under the 'Develop' category, including 'BusinessQueryCreditCard', 'CreditCardFraudData_EDA', 'Customer_EDA', 'Merchant_EDA', 'MLModelTrain', 'Partition_credit_card_data', 'TestNewDataModel', and 'Transaction_EDA'. The main area displays a notebook titled 'MLModelTrain' with a pipeline diagram. The pipeline consists of a 'ColumnTransformer' step containing a 'StandardScaler' and an 'OneHotEncoder', followed by a 'SMOTE' step and a 'RandomForestClassifier' step.

```
1 mode1
[1] ✓ - Command executed in 270 ms on 7:17:35 PM, 5/11/25

Pipeline
+ columntransformer: ColumnTransformer
  + num
    + cat
      + StandardScaler
      + OneHotEncoder
    + SMOTE
  + RandomForestClassifier
```



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar lists notebooks under the 'Develop' category, including 'BusinessQueryCreditCard', 'CreditCardFraudData_EDA', 'Customer_EDA', 'Merchant_EDA', 'MLModelTrain', 'Partition_credit_card_data', 'TestNewDataModel', and 'Transaction_EDA'. The main area displays a notebook titled 'MLModelTrain' with code for library imports and an pip install command. The code includes imports for pandas, numpy, train_test_split, StandardScaler, OneHotEncoder, ColumnTransformer, Pipeline, RandomForestClassifier, accuracy_score, classification_report, confusion_matrix, roc_auc_score, SMOTE, and make_pipeline. It also includes a comment for # pip install imblearn.

```
1 import pandas as pd
2 import numpy as np
3 from sklearn.model_selection import train_test_split
4 from sklearn.preprocessing import StandardScaler, OneHotEncoder
5 from sklearn.compose import ColumnTransformer
6 from sklearn.pipeline import Pipeline
7 from sklearn.ensemble import RandomForestClassifier
8 from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, roc_auc_score
9 from imblearn.over_sampling import SMOTE
10 from imblearn.pipeline import make_pipeline as make_imb_pipeline
11 import joblib

[1] ✓ - Command executed in 2 sec 28 ms on 6:57:00 PM, 5/11/25

1 # pip install imblearn
[4] ✓ - Command executed in 272 ms on 6:57:07 PM, 5/11/25
```

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar lists notebooks: BusinessQueryCreditCard, CreditCardFraudData_EDA, Customer_EDA, Merchant_EDA, MLModelTrain (selected), Partition_credit_card_data, TestNewDataModel, and Transaction_EDA. The main area displays a Python code snippet for feature engineering:

```
2 def feature_engineering(df):
3     df = df.copy()
4
5     # Time Features
6     df['trans_hour'] = df['trans_date_trans_time'].dt.hour
7     df['trans_day'] = df['trans_date_trans_time'].dt.day
8     df['trans_month'] = df['trans_date_trans_time'].dt.month
9     df['trans_dayofweek'] = df['trans_date_trans_time'].dt.dayofweek
10
11    # Geographic features
12    df['distance'] = np.sqrt(
13        (df['lat'] - df['merch_lat'])**2 +
14        (df['long'] - df['merch_long'])**2
15    )
16
17    # Age grouping
18    age_bins = [0, 20, 30, 40, 50, 60, 100]
19    df['age_group'] = pd.cut(
20        df['age'],
21        bins=age_bins,
22        labels=['0-20', '20-30', '30-40', '40-50', '50-60', '60+'],
23        right=False
24    )
25
26    # Transaction categories
27    amt_bins = [0, 10, 50, 100, 500, 1000, float('inf')]
28    df['amt_category'] = pd.cut(
29        df['amt'],
30        bins=amt_bins,
31        labels=['0-10', '10-50', '50-100', '100-500', '500-1000', '1000+'],
32        right=False
33    )
34
```

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar lists the same notebooks as the previous screenshot. The main area displays a Python code snippet for dropping columns:

```
1
2
3 # Prepare features and target
4 X = df.drop(['cc_num', 'cc_num_masked',
5             'trans_date_trans_time', 'friptime',
6             'LastName', 'street', 'dob', 'merchant_clean'],
7             1, axis=1)
```

Below the code editor, there is a section titled "Extract the Label Column" with the note "1 Means Fraud, 0 Means No Fraud". A code snippet for extracting the label column is shown:

```
1 y = df['is_fraud']
```

At the bottom, there is a section titled "Split the Data into two part".

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar lists notebooks: BusinessQueryCreditCard, CreditCardFraudData_EDA, Customer_EDA, Merchant_EDA, and MLModelTrain (which is selected). The main area displays the code for the MLModelTrain notebook:

```
1 # Reset Indices
2 X_train = X_train.reset_index(drop=True)
3 X_test = X_test.reset_index(drop=True)
4 y_train = y_train.reset_index(drop=True)
5 y_test = y_test.reset_index(drop=True)
6
7 # Define preprocessing
8 numeric_features = [
9     'zip', 'lat', 'long', 'city_pop', 'age',
10    'merch_lat', 'merch_long', 'merch_zipcode',
11    'amt', 'unix_time', 'trans_hour', 'trans_day',
12    'trans_month', 'trans_dayofweek', 'distance'
13 ]
14
15 categorical_features = [
16     'gender', 'job', 'city', 'state',
17     'category', 'age_group', 'amt_category'
18 ]
19
20 preprocessor = ColumnTransformer(
21     transformers=[
22         ('num', StandardScaler(), numeric_features),
23         ('cat', OneHotEncoder(handle_unknown='ignore'), categorical_features)
24     ],
25     remainder='drop'
26 )
```

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar lists notebooks: BusinessQueryCreditCard, CreditCardFraudData_EDA, Customer_EDA, Merchant_EDA, and MLModelTrain (which is selected). The main area displays the code for the MLModelTrain notebook:

```
1
2
3 # Build optimized pipeline
4 model = make_imb_pipeline(
5     preprocessor,
6     SVC(solver='auto', random_state=42, k_neighbors=5),
7     Random ForestClassifier(
8         n_estimators=200,
9         class_weight='balanced',
10        random_state=42,
11        max_depth=8,
12        n_jobs=-1
13    )
14 )
15
16 # Train model
17 print("Training model...")
18 model.fit(X_train, y_train)
19 print("Training completed.\n")
```

Below the code, there is a section titled "Model Evaluate" with the sub-instruction "find out the Accuracy".

ULTIM/ | Shubh | Mk Ways | Learnly | pandas | what | New To | histogram | Credit | ccl | Credit | New To | +

web.azuresynapse.net/en/authoring/analyze/notebooks/MLModelTrain?workspace=%2Fsubscriptions%2F81c97be5-d6a6-4431-89a7-... | Search | All Bookmarks

LTMindtree Favorites Folder

Microsoft Azure | Synapse Analytics > cfraud-synapse

We use optional cookies to provide a better experience. Learn more | Accept | Reject | More options

Synapse live | Validate all | Publish all

Develop | Filter resources by name | Notebooks | BusinessQueryCreditCard | CreditCardFraudData_EDA | Partition_credit_card_... | MLModelTrain | Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all | Undo | Publish | Outline | Attach to cfraudsparkpool | Language PySpark (Python) | Variables

MLModelTrain

Not started

```
35 ## Check the Column Data Type
36 for col in numeric_features:
37     dummy_df[col] = pd.to_numeric(dummy_df[col])
38
39
```

Command executed in 4 min 22 sec 338 ms on 7/02/2021, 5/11/2025

Training model...
Training completed.

Training Evaluation:

Accuracy: 0.9364
ROC AUC: 0.9515

Classification Report:

	precision	recall	f1-score	support
Non-Fraud	1.00	0.94	0.97	849332
Fraud	0.87	0.76	0.82	4919
accuracy			0.94	854251
macro avg	0.53	0.85	0.54	854251
weighted avg	0.99	0.94	0.96	854251

Confusion Matrix:

```
[[198985 13348]
 [ 289  941]]
```

09:35 | ENG IN | 12-05-2025

ULTIM/ | Shubh | Mk Ways | Learnly | pandas | what | New To | histogram | Credit | ccl | Credit | New To | +

web.azuresynapse.net/en/authoring/analyze/notebooks/MLModelTrain?workspace=%2Fsubscriptions%2F81c97be5-d6a6-4431-89a7-... | Search | All Bookmarks

LTMindtree Favorites Folder

Microsoft Azure | Synapse Analytics > cfraud-synapse

We use optional cookies to provide a better experience. Learn more | Accept | Reject | More options

Synapse live | Validate all | Publish all

Develop | Filter resources by name | Notebooks | BusinessQueryCreditCard | CreditCardFraudData_EDA | Partition_credit_card_... | MLModelTrain | Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all | Undo | Publish | Outline | Attach to cfraudsparkpool | Language PySpark (Python) | Variables

MLModelTrain

Not started

```
... [[198985 13348]
 [ 289  941]]
```

Dummy Transaction Prediction: Fraud
Fraud Probability: 59.56%

Top 10 Features:

Feature	Value
amt	0.195373
amt_category_100-500	0.116776
trans_hour	0.099139
amt_category_50-100	0.093747
amt_category_0-10	0.055143
category_shopping_net	0.042399
category_grocery_pos	0.035371
category_misc_net	0.033098
category_food_dining	0.026089
category_home	0.025482

dtype: float64

09:36 | ENG IN | 12-05-2025

Azure Key Vault:

The screenshot shows the Azure Key Vault interface. On the left, there's a sidebar with options like 'Create', 'Group by none', and a note about viewing a new version of the Azure experience. The main area is titled 'adlskeyvaultcredit | Secrets'. It lists one secret named 'ads-key-vault' with the type 'Access control (IAM)', status 'Enabled', and no expiration date. There are tabs for 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Access policies', 'Resource visualizer', 'Events', 'Objects', 'Keys', 'Secrets', and 'Certificates'. A search bar at the top is empty. At the bottom, there's a message to add or remove resources by pressing Ctrl+Shift+F.

The screenshot shows the Azure Synapse Analytics interface. The left sidebar includes 'Synapse live', 'Validate all', 'Publish all', 'Analytics pools', 'SQL pools', 'Apache Spark pools', 'Data Explorer pools (prev...)', 'External connections', 'Linked services', and 'Microsoft Purview'. The 'Linked services' section is currently selected. It shows two entries: 'creditcardfraudworkspace-WorkspaceDefaultSqlServer' (Type: Azure Synapse) and 'creditcardfraudworkspace-WorkspaceDefaultStorage' (Type: Azure Data). On the right, there's a configuration panel for a linked service. It has sections for 'Connect via integration runtime' (set to 'AutoResolveIntegrationRuntime'), 'Authentication type' (set to 'Account key'), 'Account selection method' (set to 'Enter manually'), 'URL' (set to 'https://<accountname>.dfs.core.windows.net'), 'Storage account key' (set to 'Azure Key Vault'), and 'AKV linked service' (with a dropdown menu). The status bar at the bottom indicates '09:42 IN 12-05-2025'.

Step 7:



1. Total Records and Fraud Cases

- Total Records: 1M
- Total Fraud Cases: 6K

2. Fraudsters per State per City

- A bar chart showing fraud counts for different cities (Achille, Acworth, Adams, Afton, Akron) across various states (TX, PA, NY, FL).

3. Fraud Cases Greater Than 500 by Month

- May: 722
- June: 544
- January: 692
- February: 686
- December: 616
- April: 558
- Total Fraud: 4540

4. Amount of Fraud Segregated by Gender

- A bar chart displaying the sum of fraud amounts for genders Female (F) and Male (M).

5. Fraud Count per Credit Card

- A bar chart showing fraud counts for different credit card numbers.

6. Detecting Seasonal Trends in Fraud Activity

- A line graph illustrating trends in various categories (entertainment, food_dining, gas_transport) over months from January to December.

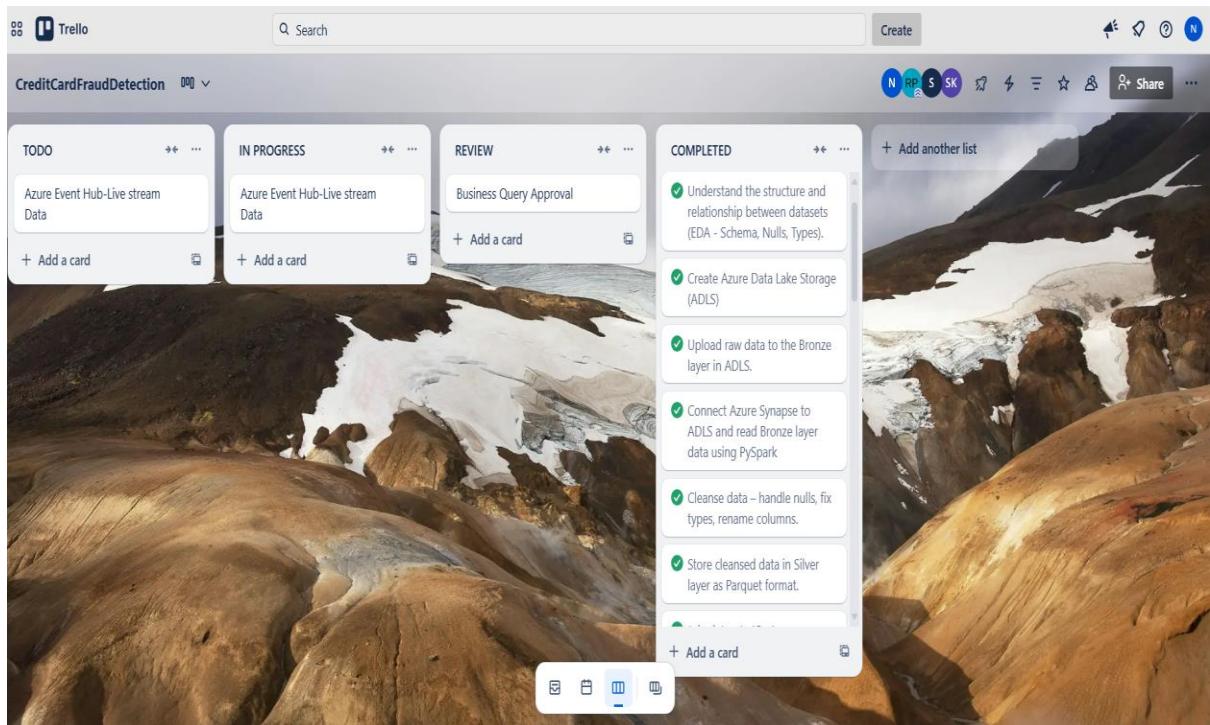
Trello

In our project, we utilized Trello for effective project management and collaboration. Trello helped us organize tasks, track progress, and ensure timely completion of project milestones.

Project Completion Status

- **Review:** 8% of the project is currently under review. This includes tasks that have been completed but are awaiting final approval and quality checks.
- **Completed:** 90% of the project tasks have been successfully completed. This includes all major deliverables and milestones.
- **Future Work:** 2% of the project is planned for future work. These tasks are identified as potential improvements or additional features that can be implemented in subsequent phases.

Trello:



FUTURE ENHANCEMENT:

- Live Stream Data: Implement live stream data processing using Azure Event Hub.
- Monitor user behavior to spot subtle fraud patterns.
- Integrate additional data sources for more context.
- Implement automatic actions for detected fraud.
- Optimize for handling more data efficiently.
- Improve tools for detailed insights and trends.