CS5 - COMPREHENSIVE WELL EVALUATION: A DEEP DIVE INTO WELL ANALYSIS CASE STUDY

Batch - 2

SRAVANI MEKA

PRIYADARSINI M S

YASH ROHILLA

AYUSH SOOD

NABIHA KHAN

Problem Statement :

The focal point of this case study is the strategic utilization of the ShaleWellCube database, a comprehensive resource provided by Rystad Energy, housing an extensive repository of data encompassing more than 1.5 million horizontal, fracked vertical, and conventional wells spanning across the regions of the US, Canada, and Argentina.

The core objective of this endeavour is to orchestrate an in-depth exploration into the reservoir of well-related information, meticulously examining historical monthly production records and an array of intricate well attributes.

Augmenting this analysis, the study endeavours to enrich the insights through the integration of projected production figures, inferred parameters, meticulous cost appraisals, breakeven pricing evaluations, NPV and IRR computations, and inclusive coverage of activities related to flaring and venting.

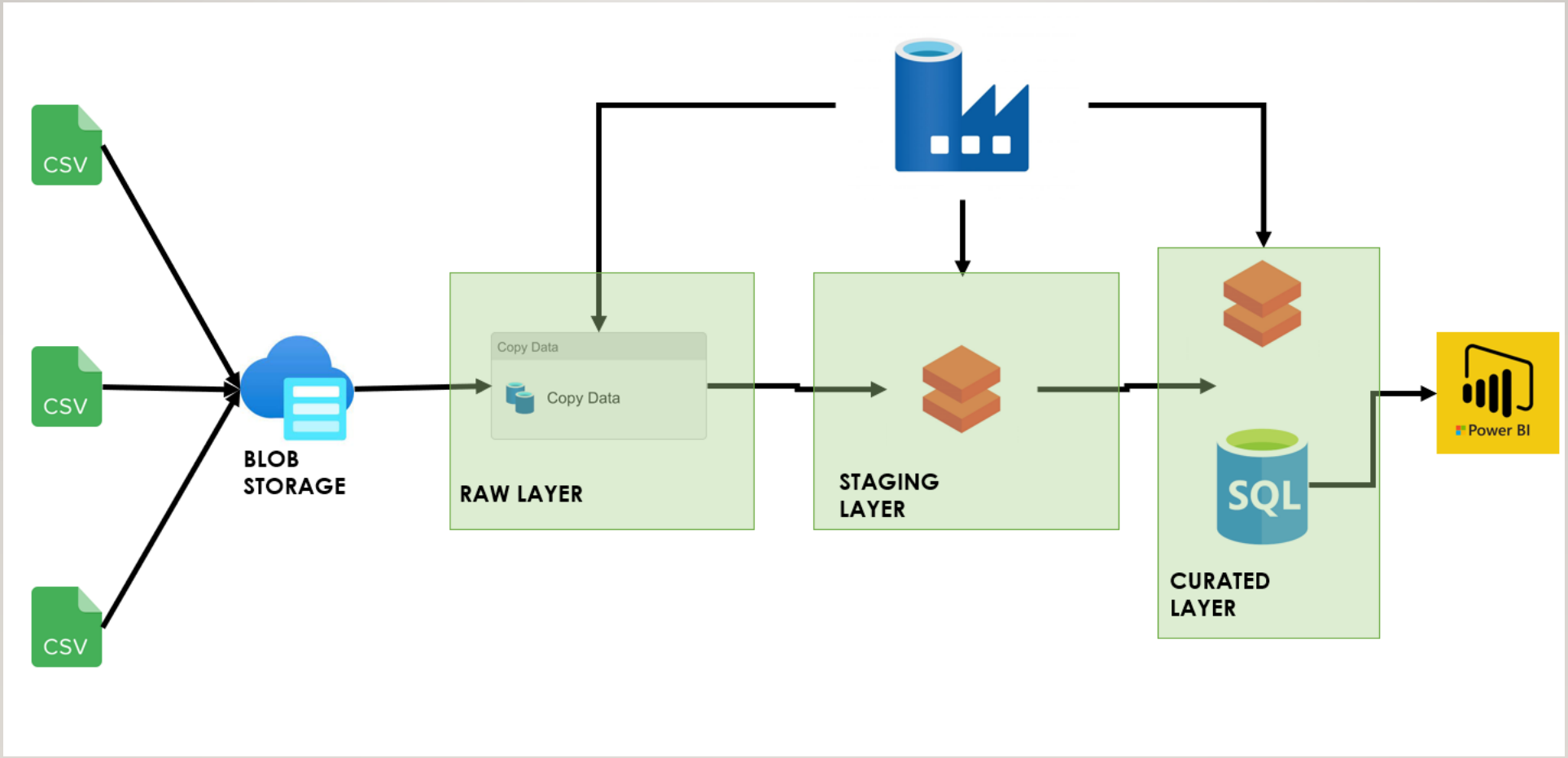
Challenges Faced:

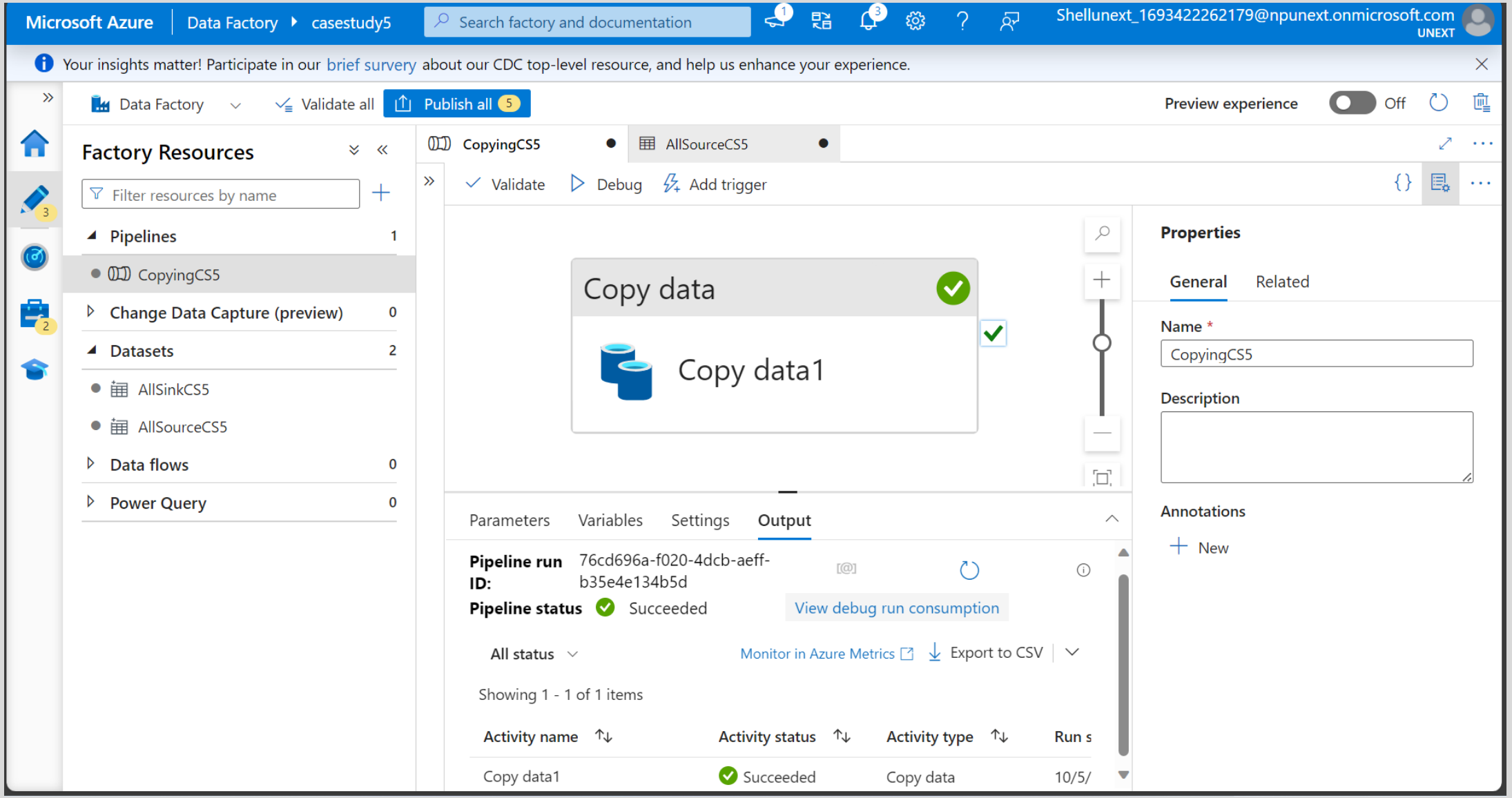
1. Data processing task was challenging because we had 23 csv tables with each table having many columns.
2. Finding relationship among tables was challenging because many tables had no common column.

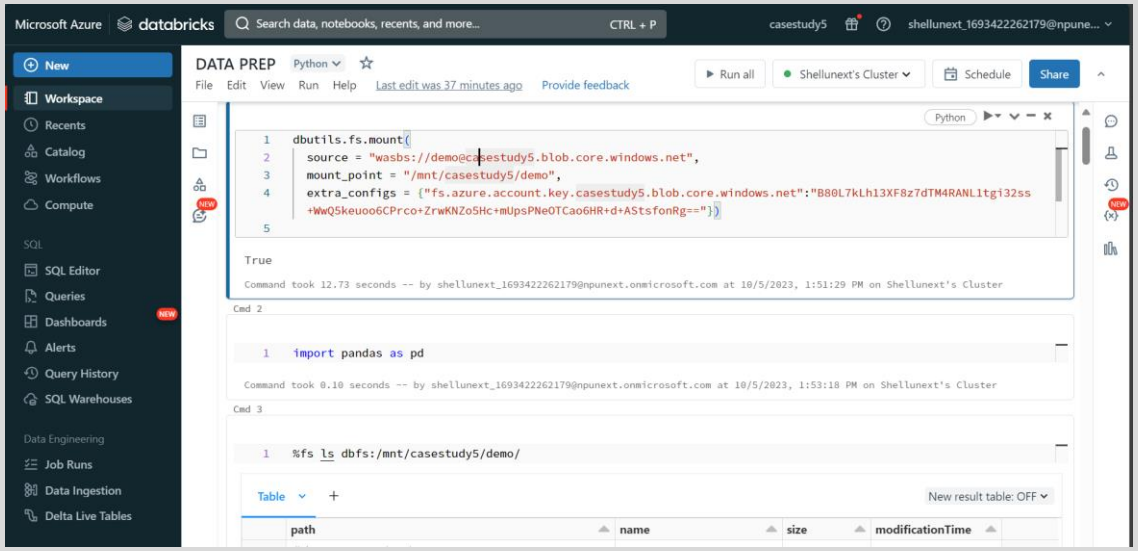
Questions:

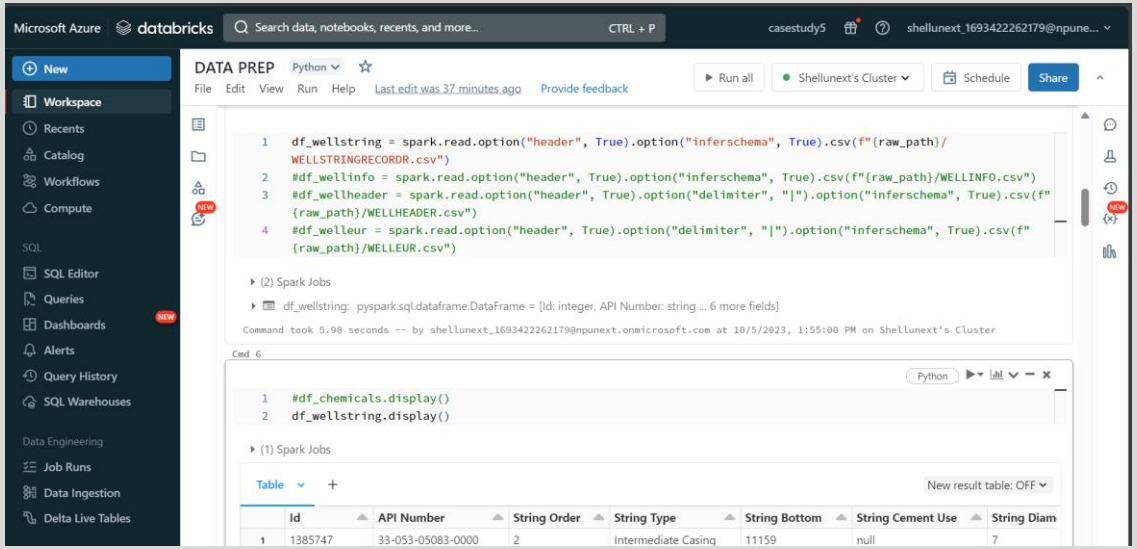
1. Historical and Forecast Production?
2. Operators With Highest Well Count?
3. Who are Flaring More?
4. Which shale play has the highest monthly production of oil and gas combined?
5. What is the average well length across all wells in the dataset?
6. How does the monthly production of Ethane compare to that of dry gas for a specific operator?

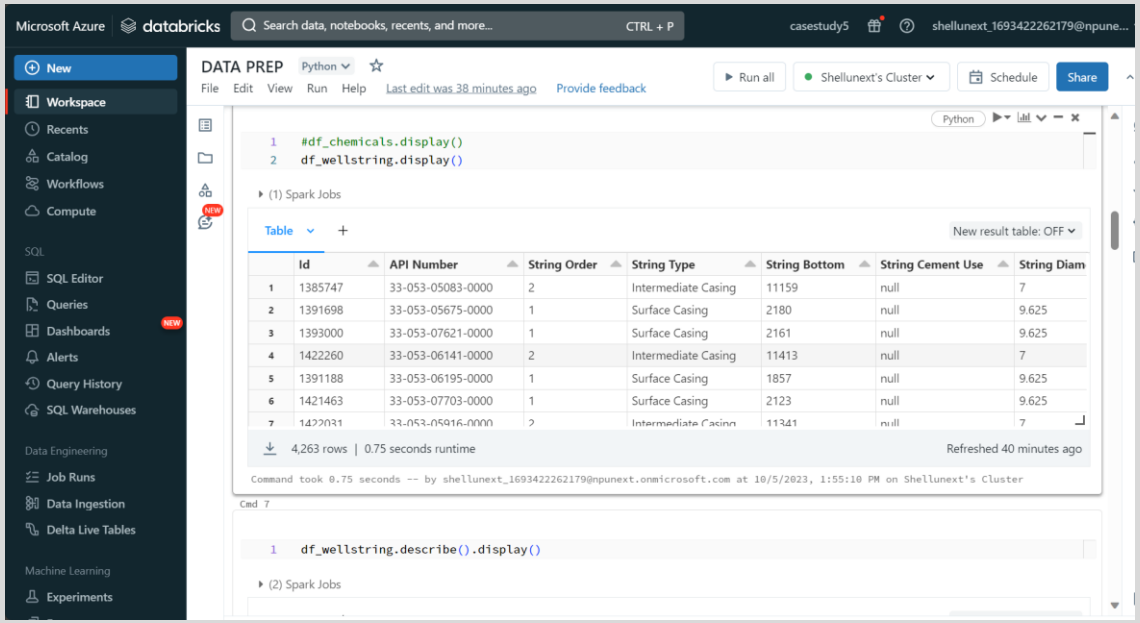
ARCHITECTURE

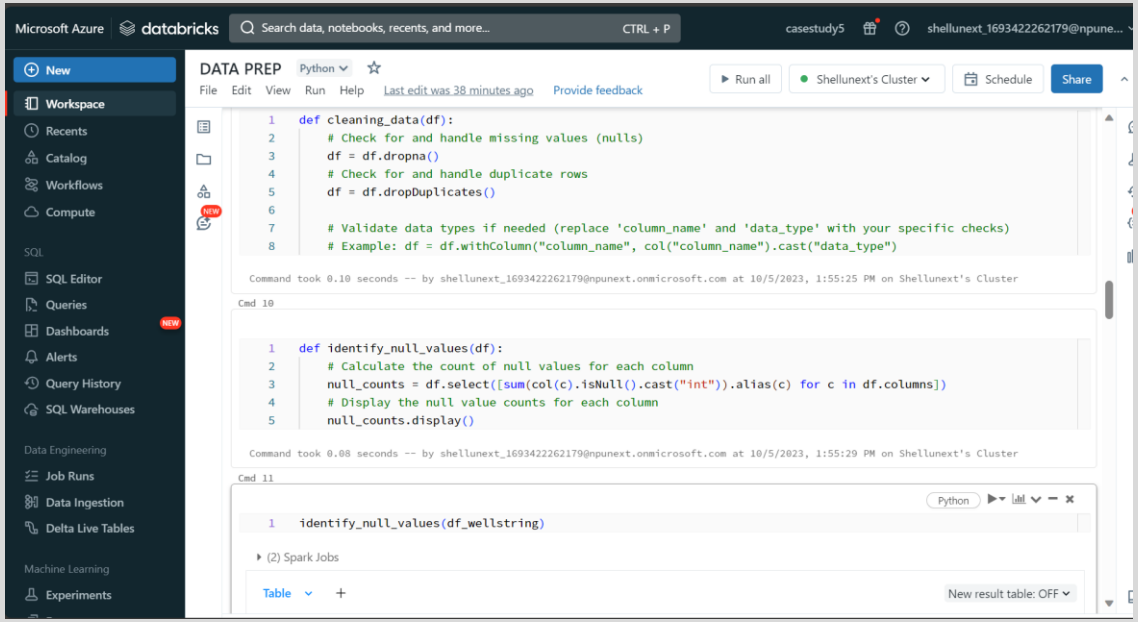


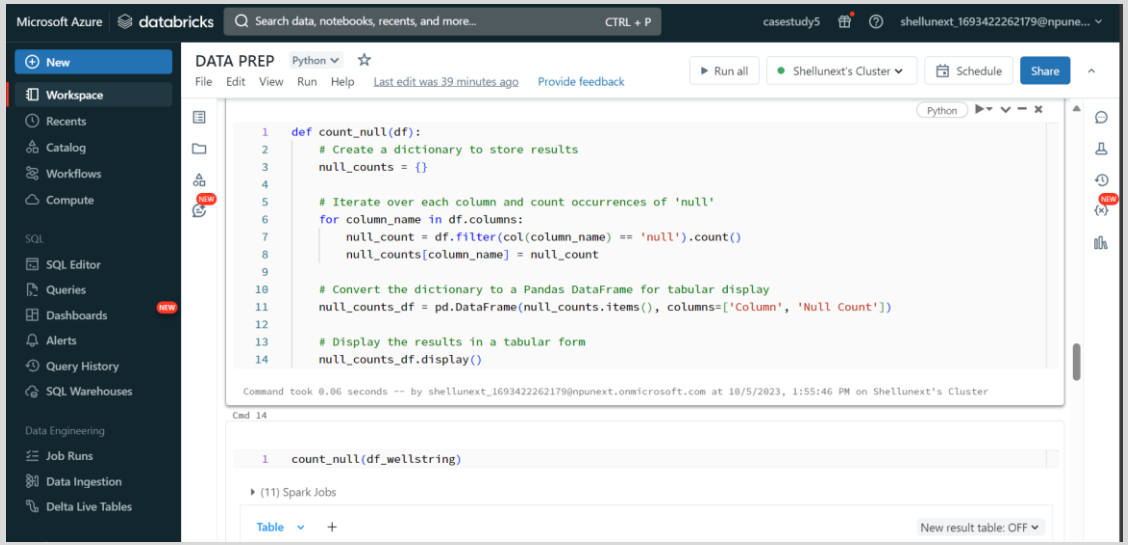

Microsoft Azure
Data Factory
casestudy5
Search factory and documentation
Shellunext_1693422262179@npunext.onmicrosoft.com
UNEXT
Your insights matter! Participate in our brief survery about our CDC top-level resource, and help us enhance your experience.
Data Factory
Validate all
Publish all 5
Preview experience
Off
Factory Resources
Filter resources by name
Pipelines 1
CopyingCS5
Change Data Capture (preview) 0
Datasets 2
AllSinkCS5
AllSourceCS5
Data flows 0
Power Query 0
CopyingCS5
AllSourceCS5
Validate
Debug
Add trigger
Copy data
Copy data1
Properties
General
Related
Name *
CopyingCS5
Description
Annotations
New
Parameters
Variables
Settings
Output
Pipeline run ID:
76cd696a-f020-4dcb-aeff-b35e4e134b5d
Pipeline status
Succeeded
View debug run consumption
All status
Monitor in Azure Metrics
Export to CSV
Showing 1 - 1 of 1 items
Activity name
Activity status
Activity type
Run s
Copy data1
Succeeded
Copy data
10/5/

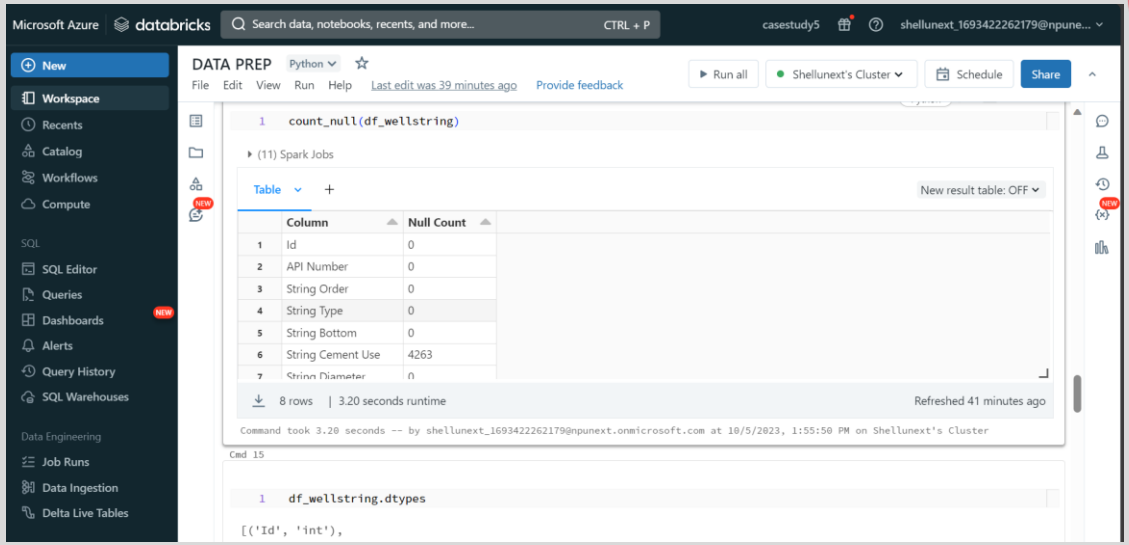

Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
DATA PREP Python
File Edit View Run Help Last edit was 37 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
dbutils.fs.mount(
source = "wasbs://demo@casestudy5.blob.core.windows.net",
mount_point = "/mnt/casestudy5/demo",
extra_configs = {"fs.azure.account.key.casestudy5.blob.core.windows.net":"B80L7kLh13XF8z7dTM4RANL1tgi32ss+WwQ5keuoo6CPrco+ZrwKNZo5Hc+mUpsPNeOTCao6HR+d+AStsfonRg=="})
True
Command took 12.73 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:51:29 PM on Shellunext's Cluster
Cmd 2
import pandas as pd
Command took 0.10 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:53:18 PM on Shellunext's Cluster
Cmd 3
%fs ls dbfs:/mnt/casestudy5/demo/
Table New result table: OFF
path name size modificationTime
New Workspace Recents Catalog Workflows Compute SQL SQL Editor Queries Dashboards Alerts Query History SQL Warehouses Data Engineering Job Runs Data Ingestion Delta Live Tables

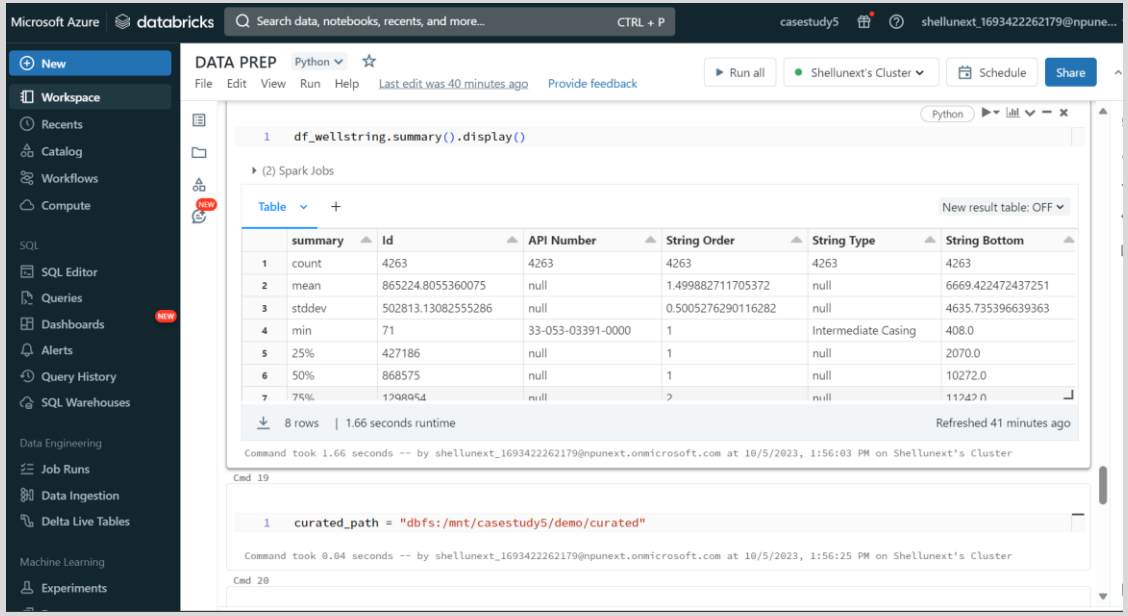

Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
DATA PREP Python
File Edit View Run Help Last edit was 37 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
df_wellstring = spark.read.option("header", True).option("inferschema", True).csv(f"{raw_path}/WELLSTRINGRECORDR.csv")
#df_wellinfo = spark.read.option("header", True).option("inferschema", True).csv(f"{raw_path}/WELLINFO.csv")
#df_wellheader = spark.read.option("header", True).option("delimiter", "|").option("inferschema", True).csv(f"{raw_path}/WELLHEADER.csv")
#df_welleur = spark.read.option("header", True).option("delimiter", "|").option("inferschema", True).csv(f"{raw_path}/WELLEUR.csv")
(2) Spark Jobs
df_wellstring: pyspark.sql.dataframe.DataFrame = [Id: integer, API Number: string ... 6 more fields]
Command took 5.98 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:55:00 PM on Shellunext's Cluster
Cmd 6
#df_chemicals.display()
df_wellstring.display()
(1) Spark Jobs
Table New result table: OFF
Id API Number String Order String Type String Bottom String Cement Use String Diam
1 1385747 33-053-05083-0000 2 Intermediate Casing 11159 null 7

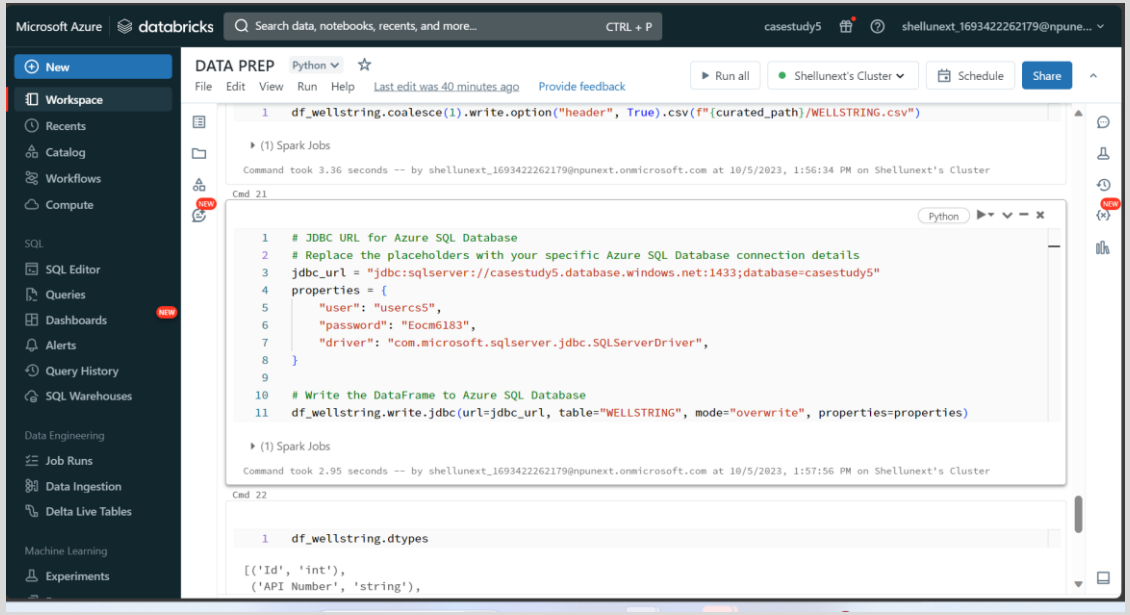

Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
DATA PREP Python
File Edit View Run Help Last edit was 38 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
#df_chemicals.display()
df_wellstring.display()
(1) Spark Jobs
Table New result table: OFF
Id API Number String Order String Type String Bottom String Cement Use String Diam
1 1385747 33-053-05083-0000 2 Intermediate Casing 11159 null 7
2 1391698 33-053-05675-0000 1 Surface Casing 2180 null 9.625
3 1393000 33-053-07621-0000 1 Surface Casing 2161 null 9.625
4 1422260 33-053-06141-0000 2 Intermediate Casing 11413 null 7
5 1391188 33-053-06195-0000 1 Surface Casing 1857 null 9.625
6 1421463 33-053-07703-0000 1 Surface Casing 2123 null 9.625
4,263 rows | 0.75 seconds runtime Refreshed 40 minutes ago
Command took 0.75 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:55:10 PM on Shellunext's Cluster
Cmd 7
df_wellstring.describe().display()
(2) Spark Jobs

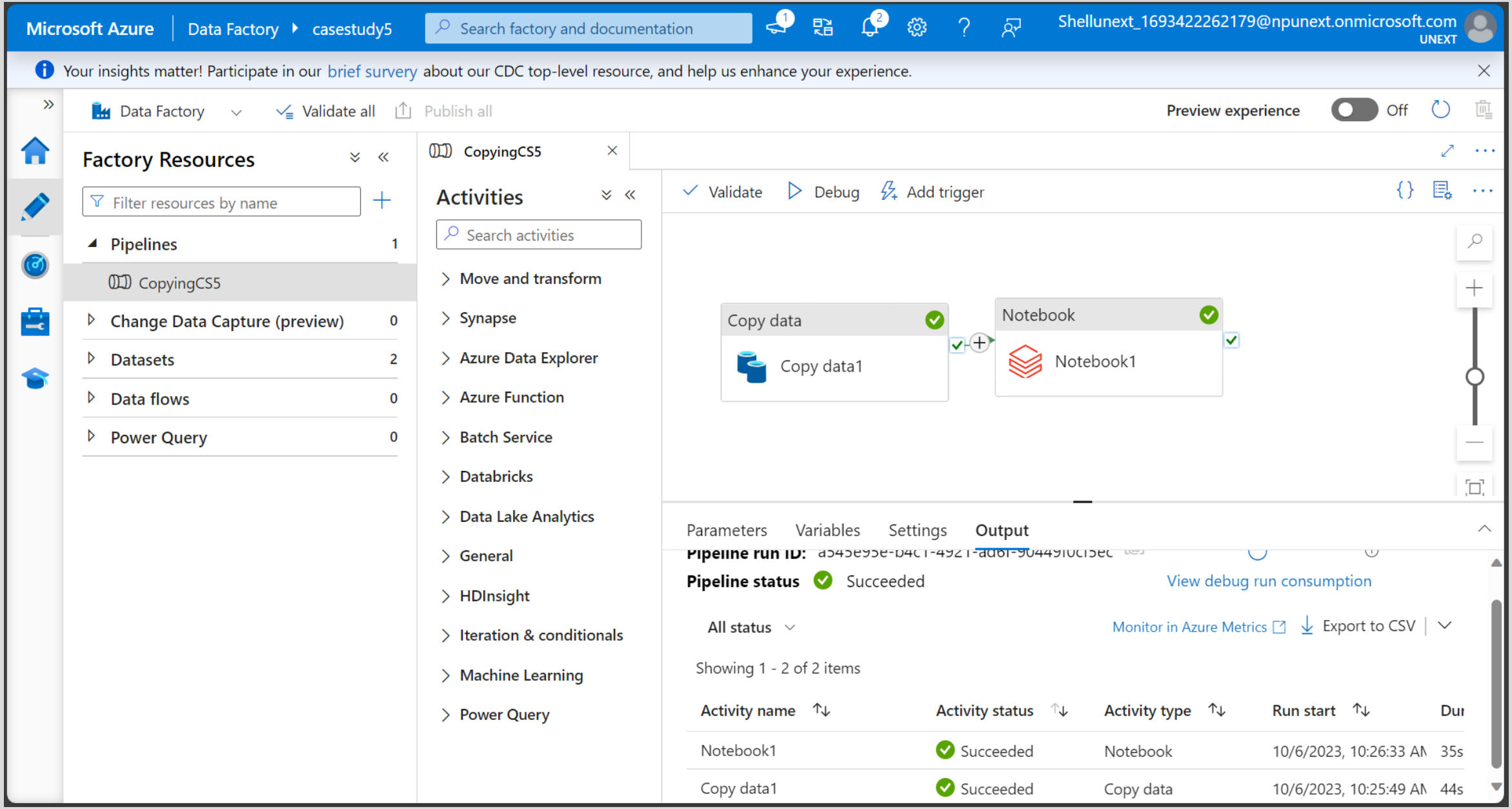

Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
DATA PREP Python
File Edit View Run Help Last edit was 38 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
def cleaning_data(df):
# Check for and handle missing values (nulls)
df = df.dropna()
# Check for and handle duplicate rows
df = df.dropDuplicates()
# Validate data types if needed (replace 'column_name' and 'data_type' with your specific checks)
# Example: df = df.withColumn("column_name", col("column_name").cast("data_type")
Command took 0.10 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:55:25 PM on Shellunext's Cluster
Cmd 10
def identify_null_values(df):
# Calculate the count of null values for each column
null_counts = df.select([sum(col(c).isNull().cast("int")).alias(c) for c in df.columns])
# Display the null value counts for each column
null_counts.display()
Command took 0.08 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:55:29 PM on Shellunext's Cluster
Cmd 11
identify_null_values(df_wellstring)
(2) Spark Jobs
Table New result table: OFF


Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
New
Workspace
Recents
Catalog
Workflows
Compute
SQL
SQL Editor
Queries
Dashboards
Alerts
Query History
SQL Warehouses
Data Engineering
Job Runs
Data Ingestion
Delta Live Tables
DATA PREP Python
File Edit View Run Help Last edit was 39 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
def count_null(df):
# Create a dictionary to store results
null_counts = {}
# Iterate over each column and count occurrences of 'null'
for column_name in df.columns:
null_count = df.filter(col(column_name) == 'null').count()
null_counts[column_name] = null_count
# Convert the dictionary to a Pandas DataFrame for tabular display
null_counts_df = pd.DataFrame(null_counts.items(), columns=['Column', 'Null Count'])
# Display the results in a tabular form
null_counts_df.display()
Command took 0.06 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:55:46 PM on Shellunext's Cluster
Cmd 14
count_null(df_wellstring)
(11) Spark Jobs
Table
New result table: OFF

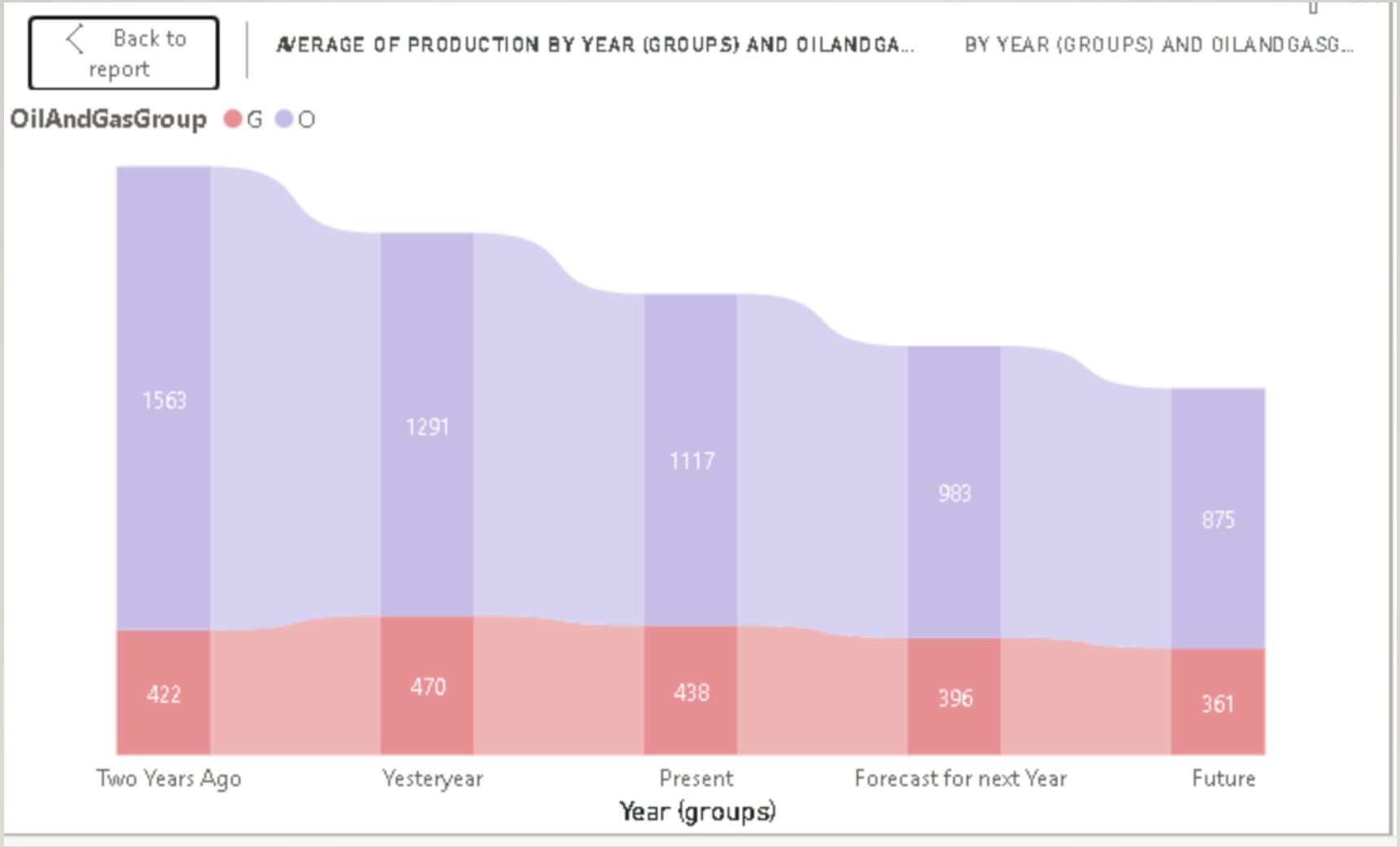

Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npunext...
DATA PREP Python
File Edit View Run Help Last edit was 39 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
count_null(df_wellstring)
(11) Spark Jobs
Table
New result table: OFF
Column Null Count
1 Id 0
2 API Number 0
3 String Order 0
4 String Type 0
5 String Bottom 0
6 String Cement Use 4263
7 String Diameter 0
8 rows | 3.20 seconds runtime
Refreshed 41 minutes ago
Command took 3.20 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:55:50 PM on Shellunext's Cluster
Cmd 15
df_wellstring.dtypes
[('Id', 'int'),


Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
DATA PREP Python
File Edit View Run Help Last edit was 40 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
df_wellstring.summary().display()
(2) Spark Jobs
Table
New result table: OFF
summary Id API Number String Order String Type String Bottom
1 count 4263 4263 4263 4263 4263
2 mean 865224.8055360075 null 1.499882711705372 null 6669.422472437251
3 stddev 502813.13082555286 null 0.5005276290116282 null 4635.735396639363
4 min 71 33-053-03391-0000 1 Intermediate Casing 408.0
5 25% 427186 null 1 null 2070.0
6 50% 868575 null 1 null 10272.0
8 rows | 1.66 seconds runtime
Refreshed 41 minutes ago
Command took 1.66 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:56:03 PM on Shellunext's Cluster
Cmd 19
curated_path = "dbfs:/mnt/casestudy5/demo/curated"
Command took 0.04 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:56:25 PM on Shellunext's Cluster
Cmd 20

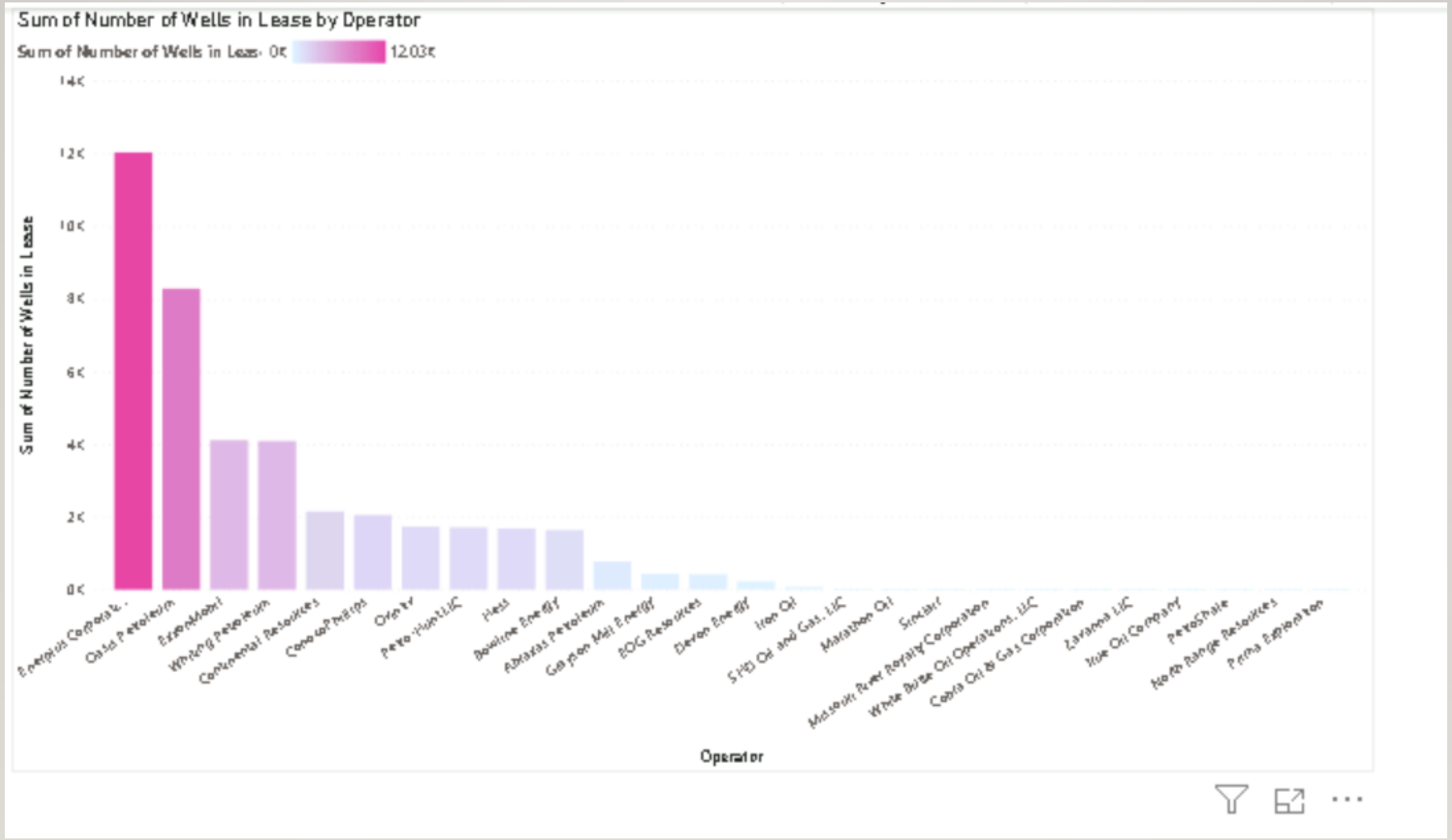

Microsoft Azure databricks Search data, notebooks, recents, and more... CTRL + P casestudy5 shellunext_1693422262179@npune...
DATA PREP Python
File Edit View Run Help Last edit was 40 minutes ago Provide feedback
Run all Shellunext's Cluster Schedule Share
df_wellstring.coalesce(1).write.option("header", True).csv(f"{curated_path}/WELLSTRING.csv")
(1) Spark Jobs
Command took 3.36 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:56:34 PM on Shellunext's Cluster
Cmd 21
# JDBC URL for Azure SQL Database
# Replace the placeholders with your specific Azure SQL Database connection details
jdbc_url = "jdbc:sqlserver://casestudy5.database.windows.net:1433;database=casestudy5"
properties = {
"user": "usercs5",
"password": "Eocm6183",
"driver": "com.microsoft.sqlserver.jdbc.SQLServerDriver",
}
# Write the DataFrame to Azure SQL Database
df_wellstring.write.jdbc(url=jdbc_url, table="WELLSTRING", mode="overwrite", properties=properties)
(1) Spark Jobs
Command took 2.95 seconds -- by shellunext_1693422262179@npunext.onmicrosoft.com at 10/5/2023, 1:57:56 PM on Shellunext's Cluster
Cmd 22
df_wellstring.dtypes
[('Id', 'int'),
('API Number', 'string'),


Microsoft Azure
Data Factory
casestudy5
Search factory and documentation
Shellunext_1693422262179@npunext.onmicrosoft.com
UNEXT
Your insights matter! Participate in our brief survery about our CDC top-level resource, and help us enhance your experience.
Data Factory
Validate all
Publish all
Preview experience
Off
Factory Resources
Filter resources by name
Pipelines 1
CopyingCS5
Change Data Capture (preview) 0
Datasets 2
Data flows 0
Power Query 0
CopyingCS5
Activities
Search activities
Move and transform
Synapse
Azure Data Explorer
Azure Function
Batch Service
Databricks
Data Lake Analytics
General
HDInsight
Iteration & conditionals
Machine Learning
Power Query
Validate
Debug
Add trigger
Copy data
Copy data1
Notebook
Notebook1
Parameters
Variables
Settings
Output
Pipeline run ID:
Pipeline status
Succeeded
View debug run consumption
All status
Monitor in Azure Metrics
Export to CSV
Showing 1 - 2 of 2 items
Activity name
Activity status
Activity type
Run start
Dur
Notebook1
Succeeded
Notebook
10/6/2023, 10:26:33 AM
35s
Copy data1
Succeeded
Copy data
10/6/2023, 10:25:49 AM
44s

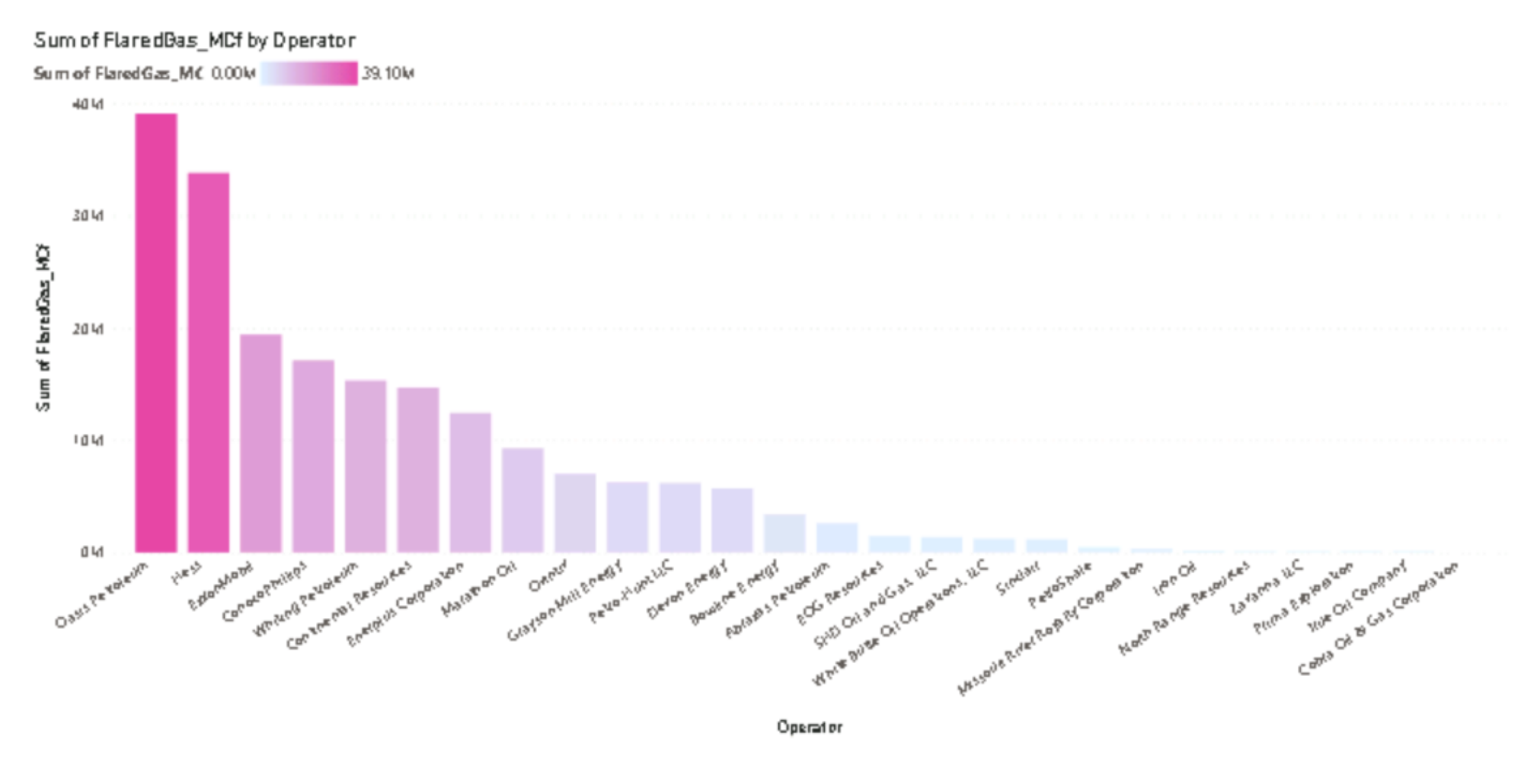
Historical and Forecast Production?



Operators With Highest Well Count?



Who are Flaring More?

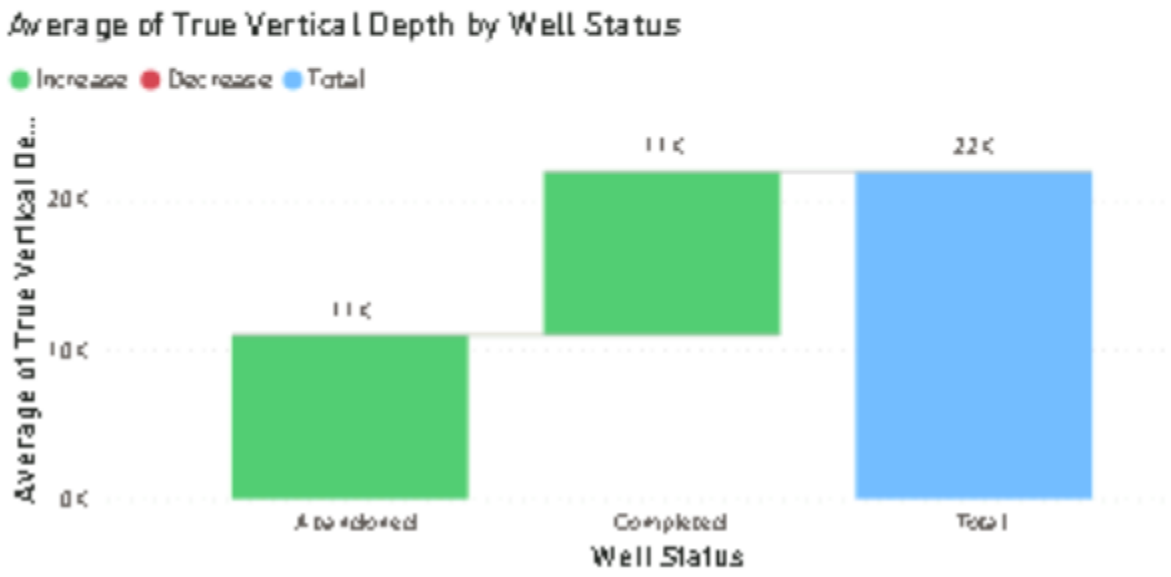


Which shale play has the highest monthly production of oil and gas combined?

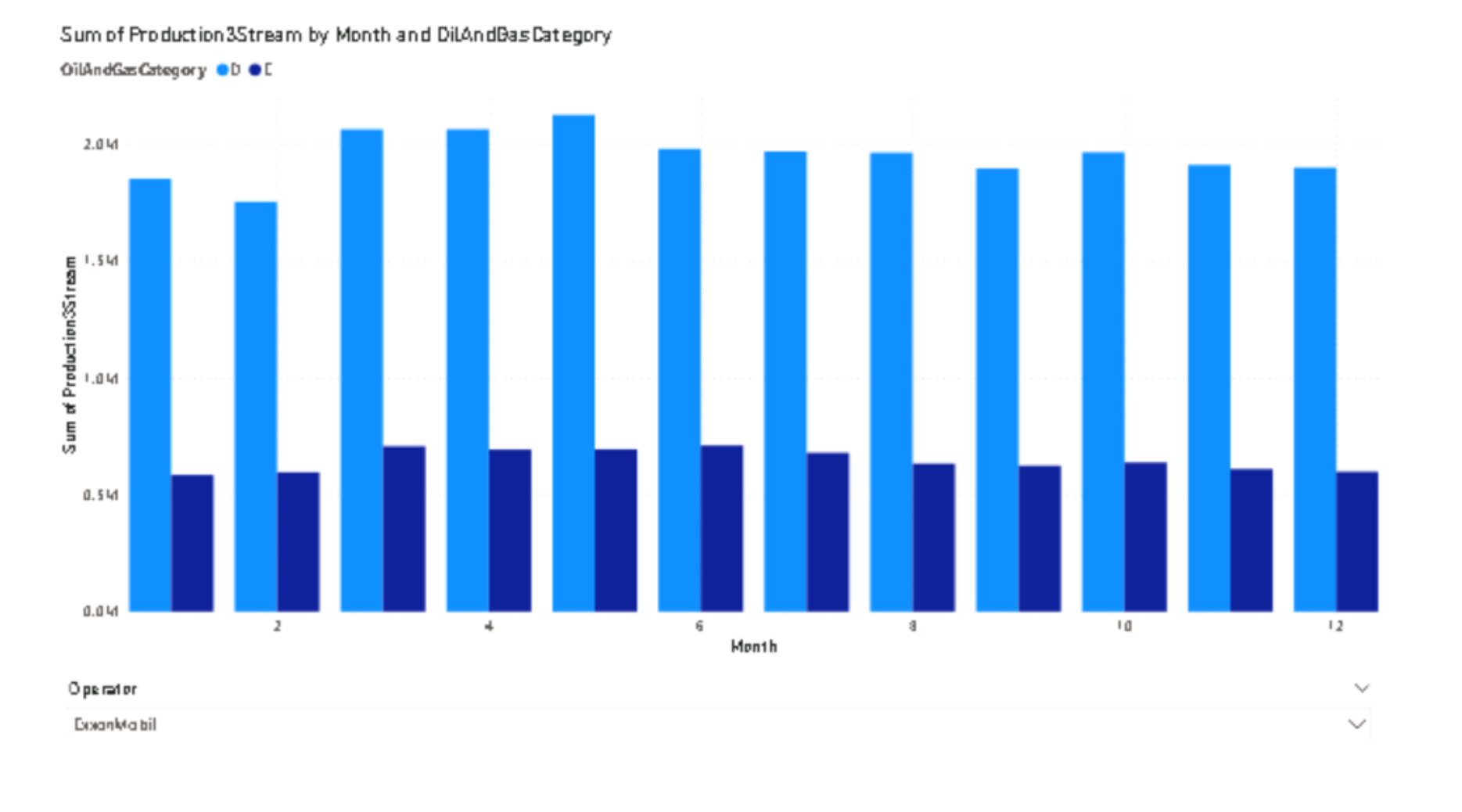
What is the average well length across all wells in the dataset?

10.93K

Average of True Vertical Depth



How does the monthly production of Ethane compare to that of dry gas for a specific operator?



Github Link:

https://github.com/nabihaakhann/WellAnalysis.git