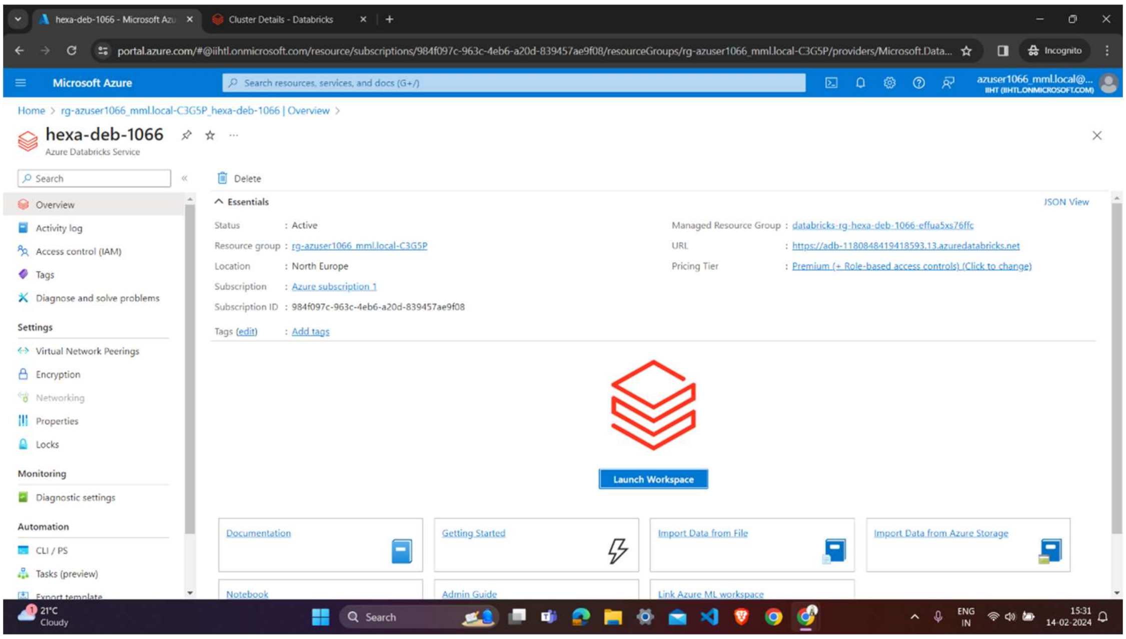
Name: Abhishek Kanoujia

DATA ENGINEERING BATCH 1

DAY 19 ASSIGNMENT

Introduction to Databricks Delta Lake:- first create cluster then perform the query


portal.azure.com/#@iihtl.onmicrosoft.com/resource/subscriptions/984f097c-963c-4eb6-a20d-839457ae9f08/resourceGroups/rg-azuser1066_mml.local-C3G5P/providers/Microsoft.Data...

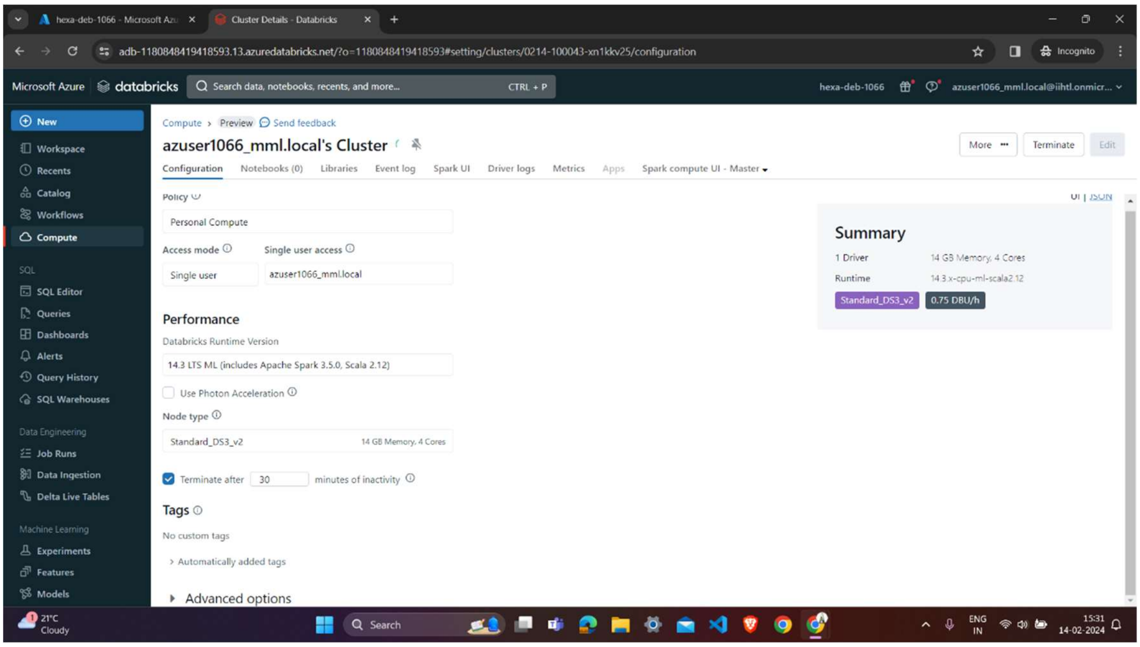
Home > rg-azuser1066_mml.local-C3G5P_hexa-deb-1066 | Overview >

hexa-deb-1066
Azure Databricks Service

Essentials

Status : Active
Resource group : rg-azuser1066_mml.local-C3G5P
Location : North Europe
Subscription : Azure subscription 1
Subscription ID : 984f097c-963c-4eb6-a20d-839457ae9f08
Tags (edit) : Add tags

Managed Resource Group : databricks-rg-hexa-deb-1066-effua5xs76ffc
URL : https://adb-1180848419418593.13.azuredatabricks.net
Pricing Tier : Premium (+ Role-based access controls) (Click to change)


adb-1180848419418593.13.azuredatabricks.net/?o=1180848419418593#setting/clusters/0214-100043-xn1kkv25/configuration

Compute > Preview Send feedback

azuser1066_mml.local's Cluster

Configuration

Policy
Personal Compute

Access mode: Single user
Single user access: azuser1066_mml.local

Performance

Databricks Runtime Version
14.3 LTS ML (includes Apache Spark 3.5.0, Scala 2.12)

☐ Use Photon Acceleration

Node type
Standard_DS3_v2 14 GB Memory, 4 Cores

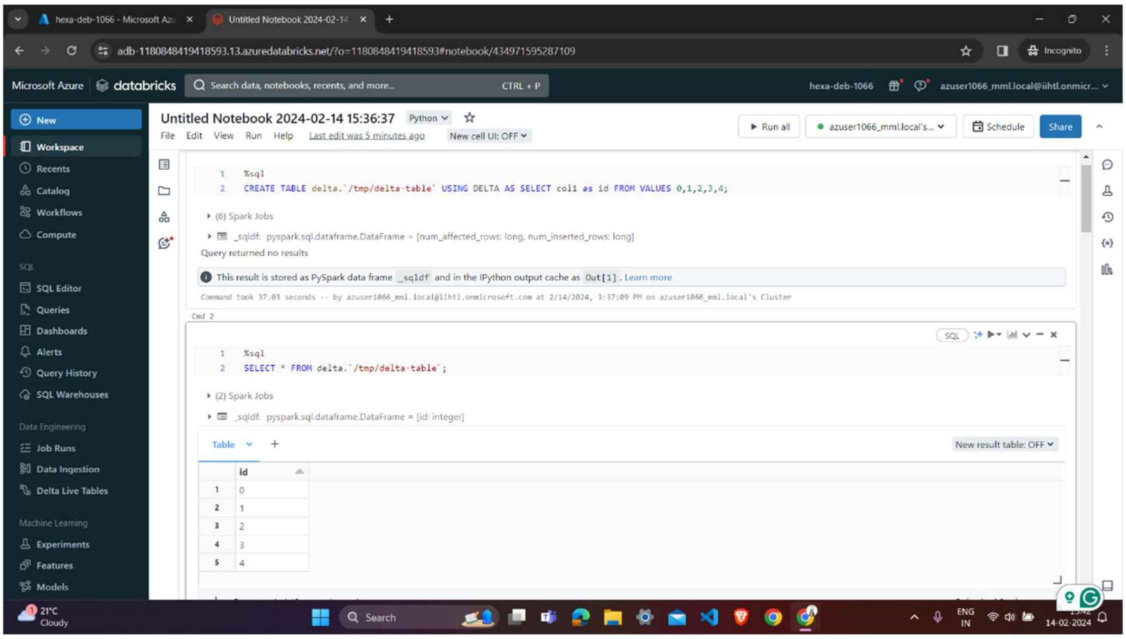
☑ Terminate after 30 minutes of inactivity

Tags
No custom tags
> Automatically added tags

Advanced options

Summary
1 Driver 14 GB Memory, 4 Cores
Runtime 14.3.x-cpu-ml-scala2.12
Standard_DS3_v2 0.75 DBU/h

Read table:-



Untitled Notebook 2024-02-14 15:36:37

```
%sql
CREATE TABLE delta.`/tmp/delta-table` USING DELTA AS SELECT col1 as id FROM VALUES 0,1,2,3,4;
```

```
(6) Spark Jobs
_sqldf:  pyspark.sql.dataframe.DataFrame = [num_affected_rows: long, num_inserted_rows: long]
Query returned no results
This result is stored as PySpark data frame _sqldf and in the IPython output cache as Out[1]. Learn more
Command took 37.03 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:37:09 PM on azuser1066_mml.local's Cluster
```

Cmd 2

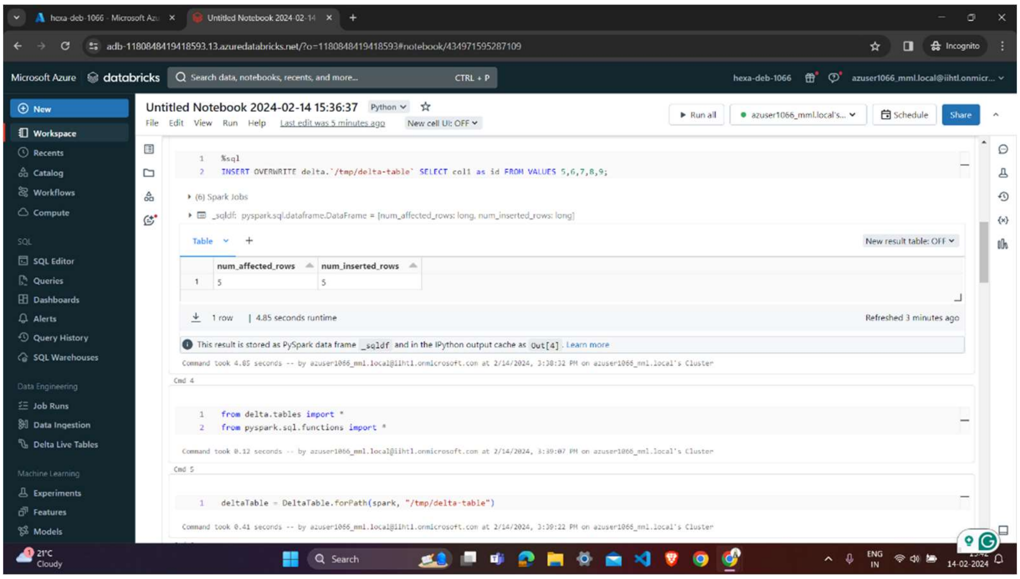
```
%sql
SELECT * FROM delta.`/tmp/delta-table`;
```

```
(2) Spark Jobs
_sqldf:  pyspark.sql.dataframe.DataFrame = [id: integer]
```

	id
1	0
2	1
3	2
4	3
5	4

Update table data:-

Conditional Update without overwrte:-



Untitled Notebook 2024-02-14 15:36:37

```
%sql
INSERT OVERWRITE delta.`/tmp/delta-table` SELECT col1 as id FROM VALUES 5,6,7,8,9;
```

```
(6) Spark Jobs
_sqldf:  pyspark.sql.dataframe.DataFrame = [num_affected_rows: long, num_inserted_rows: long]
```

	num_affected_rows	num_inserted_rows
1	5	5

```
1 row | 4.85 seconds runtime                                Refreshed 3 minutes ago
This result is stored as PySpark data frame _sqldf and in the IPython output cache as Out[4]. Learn more
Command took 4.85 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:38:32 PM on azuser1066_mml.local's Cluster
```

Cmd 4

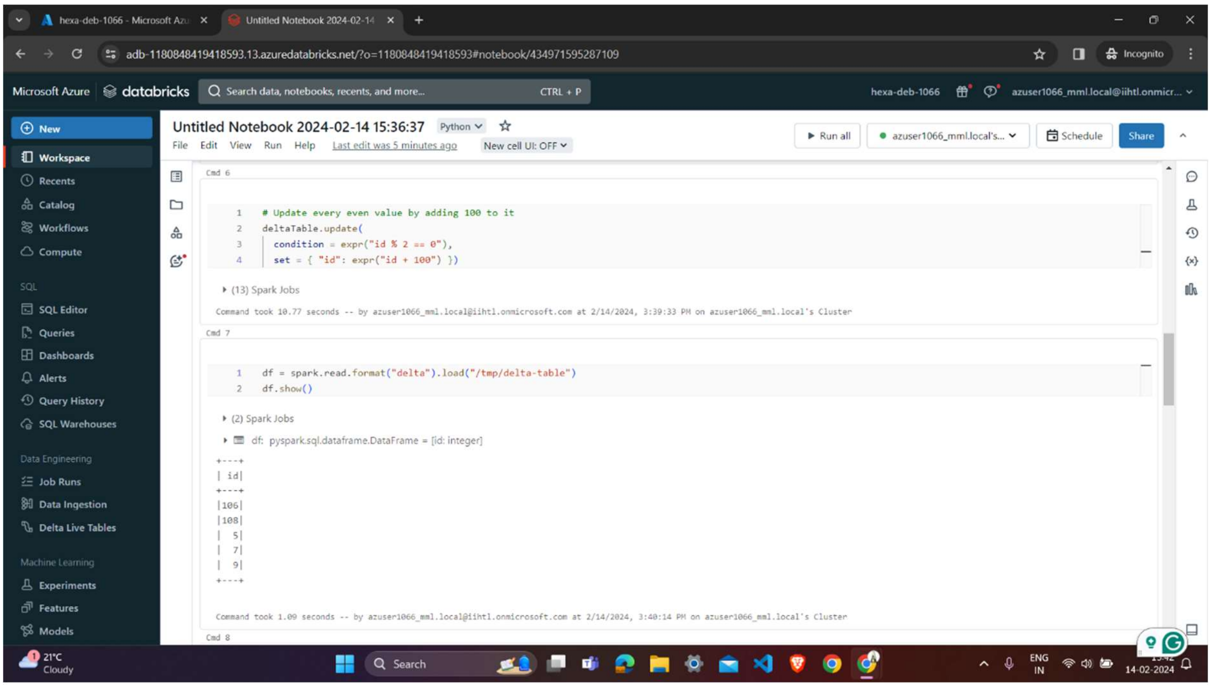
```
from delta.tables import *
from pyspark.sql.functions import *
```

```
Command took 0.12 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:39:07 PM on azuser1066_mml.local's Cluster
```

Cmd 5

```
deltaTable = DeltaTable.forPath(spark, "/tmp/delta-table")
```

```
Command took 0.41 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:39:22 PM on azuser1066_mml.local's Cluster
```



Cmd 6

```
# Update every even value by adding 100 to it
deltaTable.update(
  condition = expr("id % 2 == 0"),
  set = { "id": expr("id + 100") })
```

(13) Spark Jobs

```
Command took 10.77 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:39:33 PM on azuser1066_mml.local's Cluster
```

Cmd 7

```
df = spark.read.format("delta").load("/tmp/delta-table")
df.show()
```

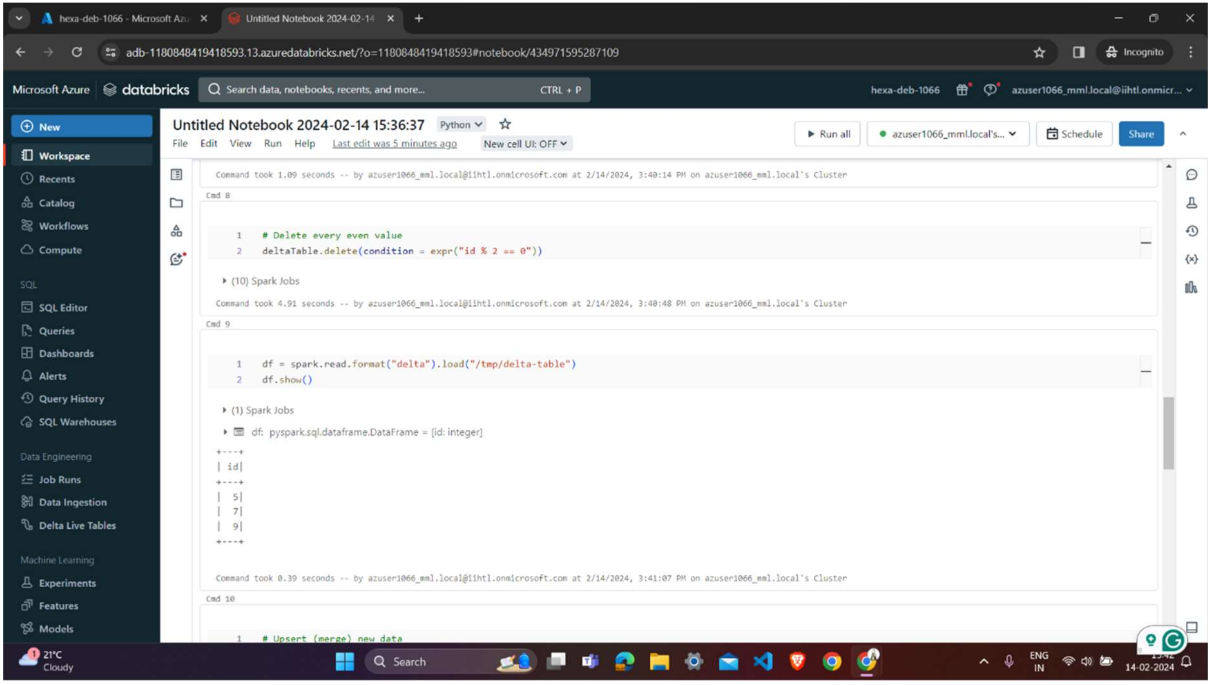
(2) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [id: integer]

```
+---+
| id|
+---+
|106|
|108|
|  5|
|  7|
|  9|
+---+
```

```
Command took 1.09 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:40:14 PM on azuser1066_mml.local's Cluster
```

Cmd 8



```
Command took 1.09 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:40:14 PM on azuser1066_mml.local's Cluster
```

Cmd 8

```
# Delete every even value
deltaTable.delete(condition = expr("id % 2 == 0"))
```

(10) Spark Jobs

```
Command took 4.91 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:40:48 PM on azuser1066_mml.local's Cluster
```

Cmd 9

```
df = spark.read.format("delta").load("/tmp/delta-table")
df.show()
```

(1) Spark Jobs

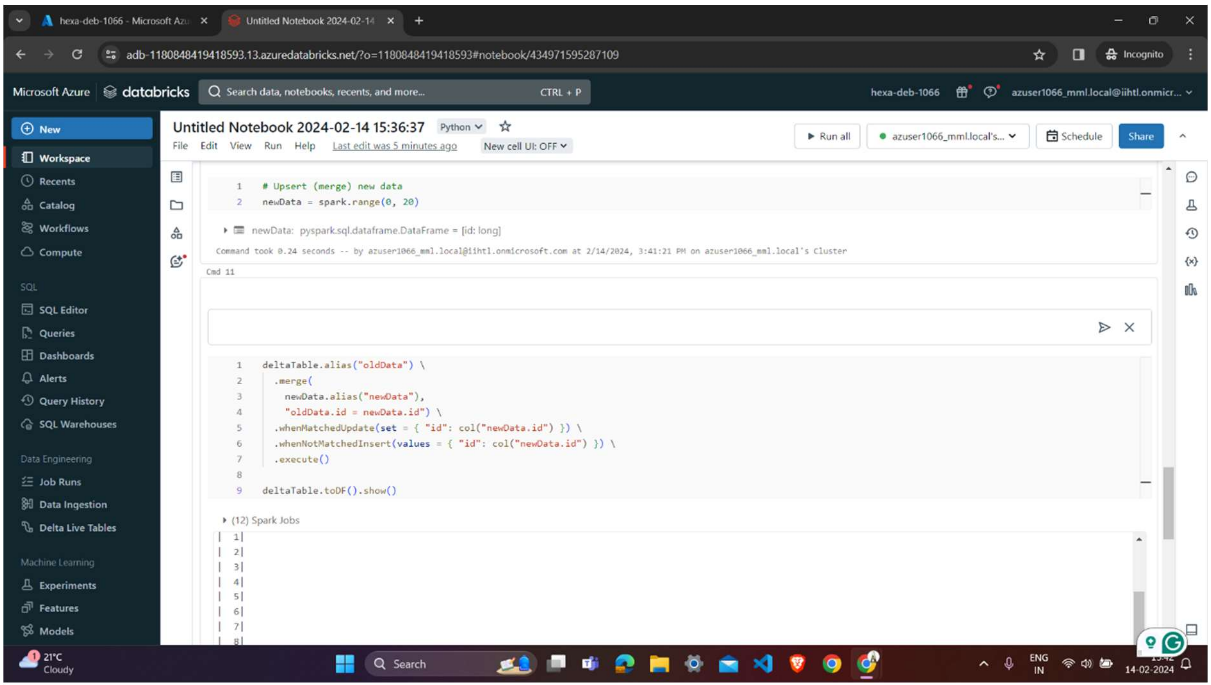
df: pyspark.sql.dataframe.DataFrame = [id: integer]

```
+---+
| id|
+---+
|  5|
|  7|
|  9|
+---+
```

```
Command took 0.39 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:41:07 PM on azuser1066_mml.local's Cluster
```

Cmd 10

```
# Upsert (merge) new data
```



Untitled Notebook 2024-02-14 15:36:37

```
# Upsert (merge) new data
newData = spark.range(0, 20)
```

```
newData:  pyspark.sql.dataframe.DataFrame = [id: long]
Command took 0.24 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:41:21 PM on azuser1066_mml.local's Cluster
```

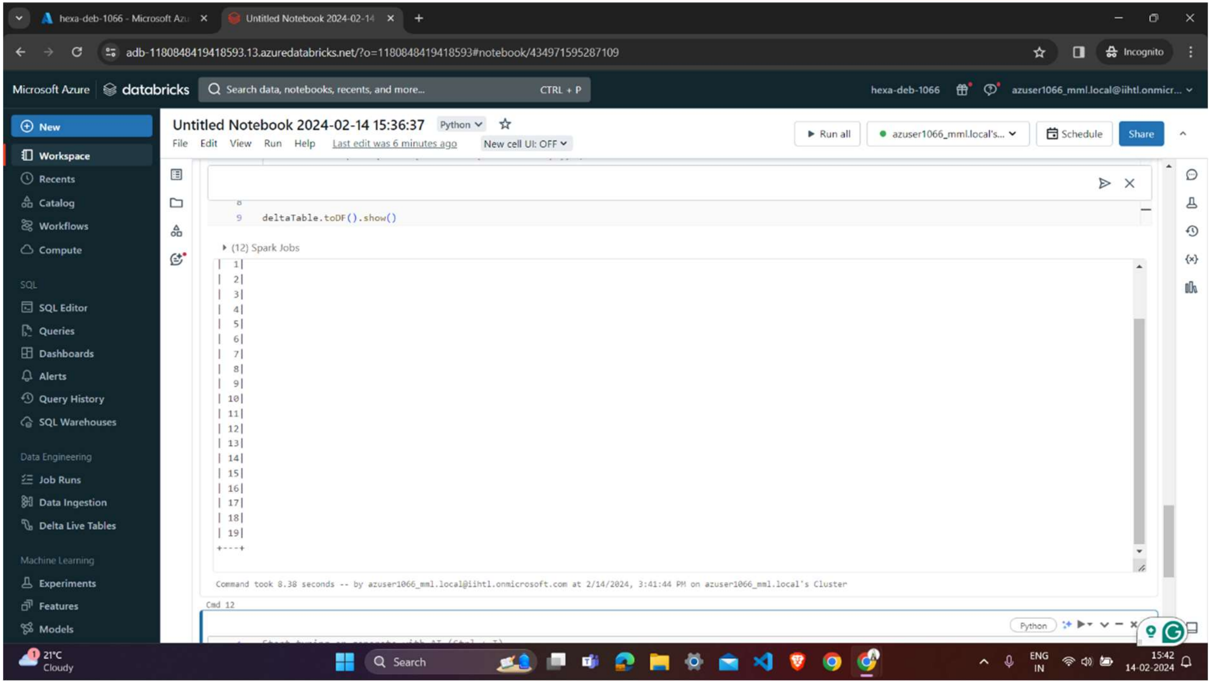
Cmd 11

```
deltaTable.alias("oldData") \
  .merge(
    newData.alias("newData"),
    "oldData.id = newData.id") \
  .whenMatchedUpdate(set = { "id": col("newData.id") }) \
  .whenNotMatchedInsert(values = { "id": col("newData.id") }) \
  .execute()

deltaTable.toDF().show()
```

(12) Spark Jobs

```
|  1|
|  2|
|  3|
|  4|
|  5|
|  6|
|  7|
|  8|
```



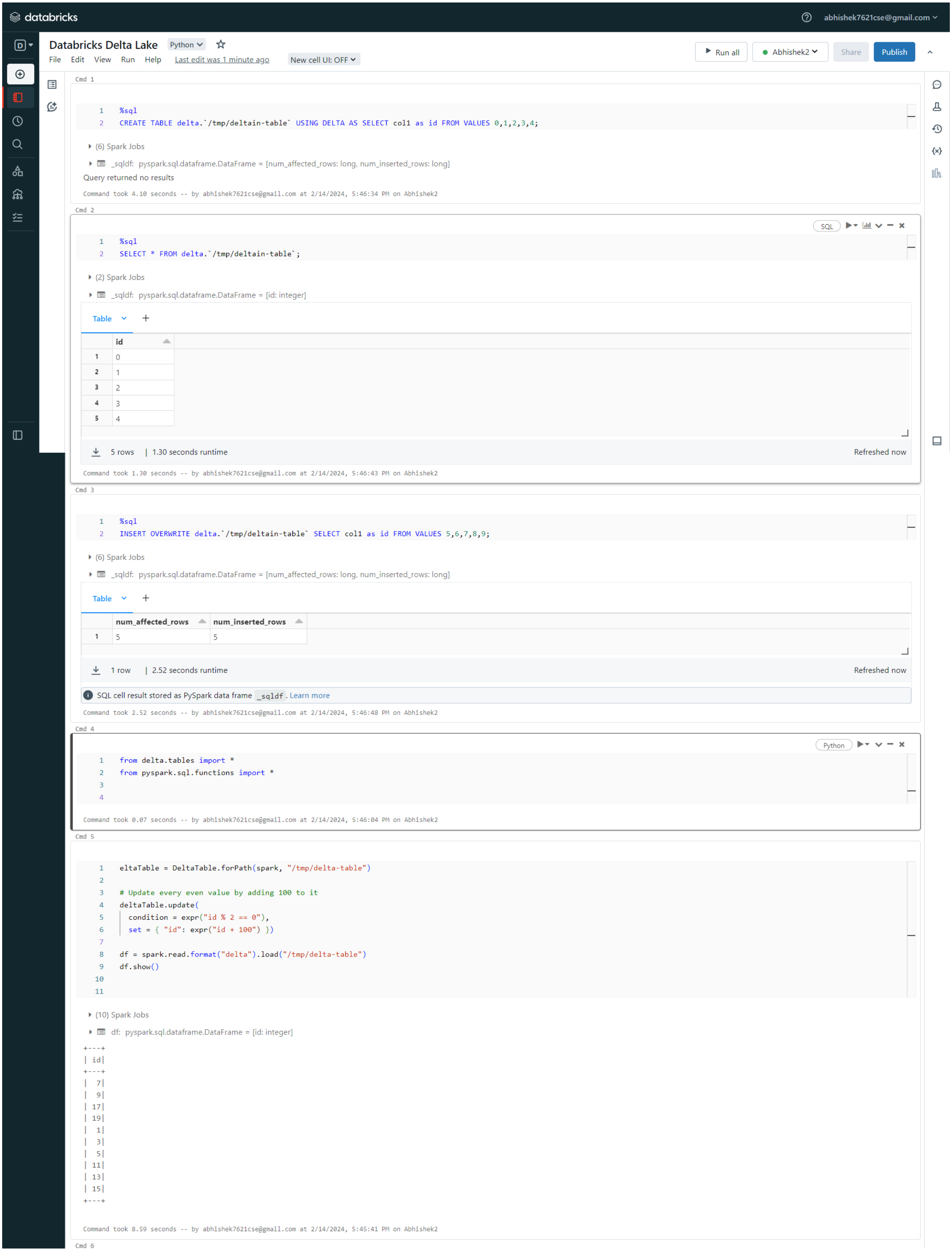
Untitled Notebook 2024-02-14 15:36:37

```
deltaTable.toDF().show()
```

(12) Spark Jobs

```
|  1|
|  2|
|  3|
|  4|
|  5|
|  6|
|  7|
|  8|
|  9|
| 10|
| 11|
| 12|
| 13|
| 14|
| 15|
| 16|
| 17|
| 18|
| 19|
+---+
Command took 8.38 seconds -- by azuser1066_mml.local@iihtl.onmicrosoft.com at 2/14/2024, 3:41:44 PM on azuser1066_mml.local's Cluster
```

Cmd 12



Cmd 1

```
%sql
CREATE TABLE delta.`/tmp/deltain-table` USING DELTA AS SELECT col1 as id FROM VALUES 0,1,2,3,4;
```

(6) Spark Jobs

_sqldf: pyspark.sql.dataframe.DataFrame = [num_affected_rows: long, num_inserted_rows: long]

Query returned no results

```
Command took 4.10 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:46:34 PM on Abhishek2
```

Cmd 2

```
%sql
SELECT * FROM delta.`/tmp/deltain-table`;
```

(2) Spark Jobs

_sqldf: pyspark.sql.dataframe.DataFrame = [id: integer]

	id
1	0
2	1
3	2
4	3
5	4

5 rows | 1.30 seconds runtime Refreshed now

```
Command took 1.30 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:46:43 PM on Abhishek2
```

Cmd 3

```
%sql
INSERT OVERWRITE delta.`/tmp/deltain-table` SELECT col1 as id FROM VALUES 5,6,7,8,9;
```

(6) Spark Jobs

_sqldf: pyspark.sql.dataframe.DataFrame = [num_affected_rows: long, num_inserted_rows: long]

	num_affected_rows	num_inserted_rows
1	5	5

1 row | 2.52 seconds runtime Refreshed now

SQL cell result stored as PySpark data frame `_sqldf`. Learn more

```
Command took 2.52 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:46:48 PM on Abhishek2
```

Cmd 4

```
from delta.tables import *
from pyspark.sql.functions import *


```

```
Command took 0.07 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:46:04 PM on Abhishek2
```

Cmd 5

```
eltaTable = DeltaTable.forPath(spark, "/tmp/delta-table")

# Update every even value by adding 100 to it
deltaTable.update(
  condition = expr("id % 2 == 0"),
  set = { "id": expr("id + 100") })

df = spark.read.format("delta").load("/tmp/delta-table")
df.show()


```

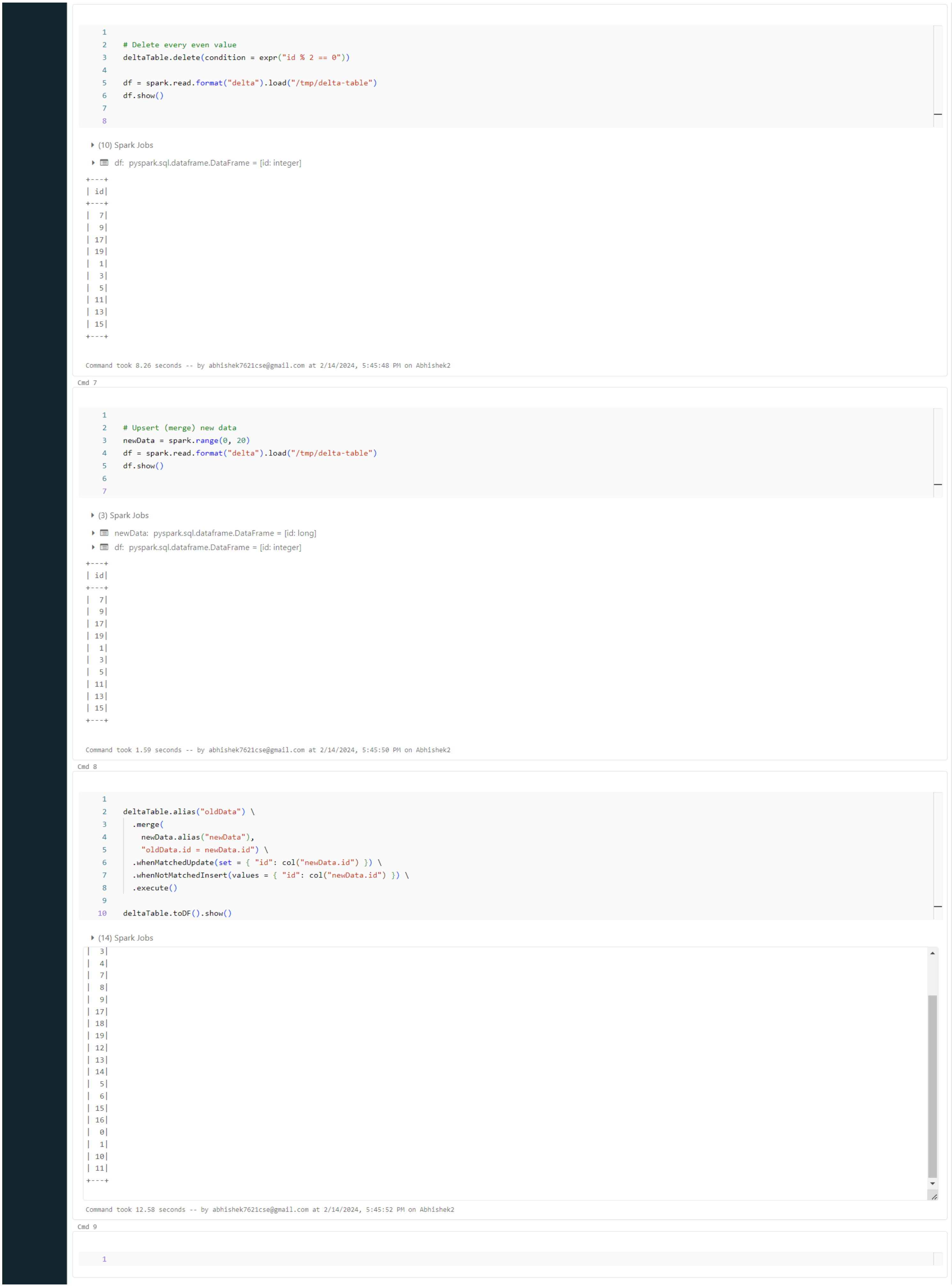
(10) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [id: integer]

```
+---+
| id|
+---+
|  7|
|  9|
| 17|
| 19|
|  1|
|  3|
|  5|
| 11|
| 13|
| 15|
+---+
```

```
Command took 8.59 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:45:41 PM on Abhishek2
```

Cmd 6



```
# Delete every even value
deltaTable.delete(condition = expr("id % 2 == 0"))

df = spark.read.format("delta").load("/tmp/delta-table")
df.show()
```

df: pyspark.sql.dataframe.DataFrame = [id: integer]

```
+---+
| id|
+---+
|  7|
|  9|
| 17|
| 19|
|  1|
|  3|
|  5|
| 11|
| 13|
| 15|
+---+
```

Command took 8.26 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:45:48 PM on Abhishek2

Cmd 7

```
# Upsert (merge) new data
newData = spark.range(0, 20)
df = spark.read.format("delta").load("/tmp/delta-table")
df.show()
```

newData: pyspark.sql.dataframe.DataFrame = [id: long]

df: pyspark.sql.dataframe.DataFrame = [id: integer]

```
+---+
| id|
+---+
|  7|
|  9|
| 17|
| 19|
|  1|
|  3|
|  5|
| 11|
| 13|
| 15|
+---+
```

Command took 1.59 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:45:50 PM on Abhishek2

Cmd 8

```
deltaTable.alias("oldData") \
  .merge(
    newData.alias("newData"),
    "oldData.id = newData.id") \
  .whenMatchedUpdate(set = { "id": col("newData.id") }) \
  .whenNotMatchedInsert(values = { "id": col("newData.id") }) \
  .execute()

deltaTable.toDF().show()
```

```
|  3|
|  4|
|  7|
|  8|
|  9|
| 17|
| 18|
| 19|
| 12|
| 13|
| 14|
|  5|
|  6|
| 15|
| 16|
|  0|
|  1|
| 10|
| 11|
+---+
```

Command took 12.58 seconds -- by abhishek7621cse@gmail.com at 2/14/2024, 5:45:52 PM on Abhishek2

Cmd 9

```
[Shift+Enter] to run
[Shift+Ctrl+Enter] to run selected text
```