GROUP 7

BIG DATA ANALYTICS USING HADOOP, HIVE & SPARK ON MULTI-YEAR IPEDS DATASET

Team Members:
Sai Teja KMVP - MXK240054
Sai Hemalatha Ramidi - SXR240040

PROBLEM STATEMENT

IPEDS publishes large multi-year datasets covering enrollment, completions, institutional characteristics, and more. Manual analysis of these datasets is impractical due to volume, file size, and schema variability across years.

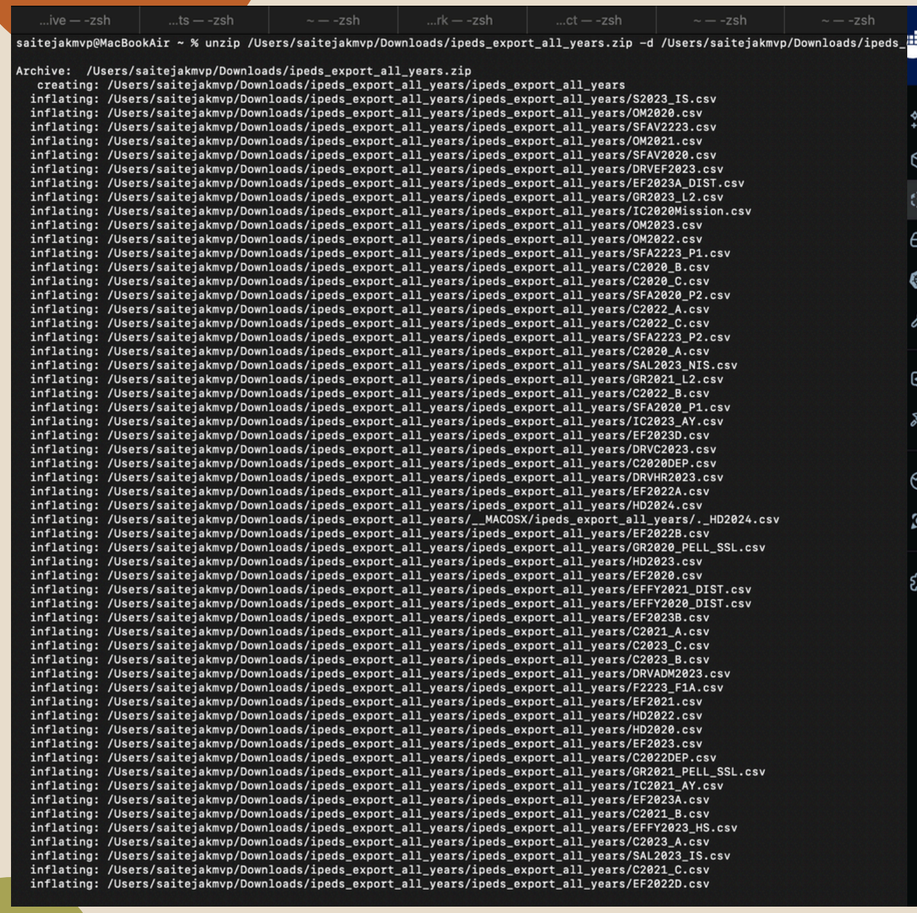
OBJECTIVE:

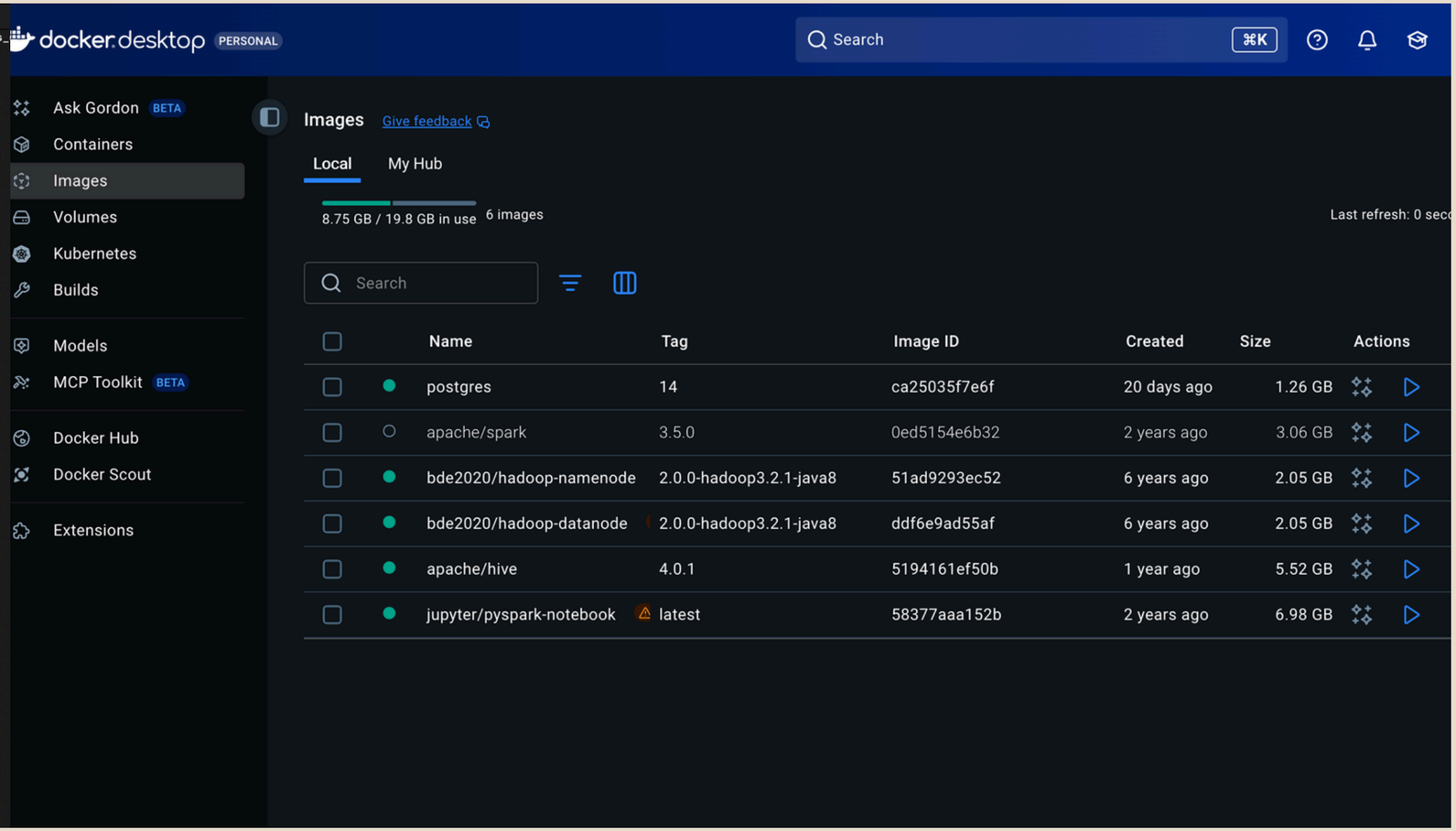
To use Hadoop, Hive, and Spark analytics on 1GB+ multi-year IPEDS datasets (2020–2023) to answer three key business questions related to trends in enrollment and online learning.

PROJECT REQUIREMENT & COMPLIANCE

This project satisfies all mandatory components:

Hadoop/Hive Setup, Spark for distributed analysis, 1GB+ data from IPEDS, 3 Business Questions, multiple datasets used: EF, EFFY/EFIA, C, HD



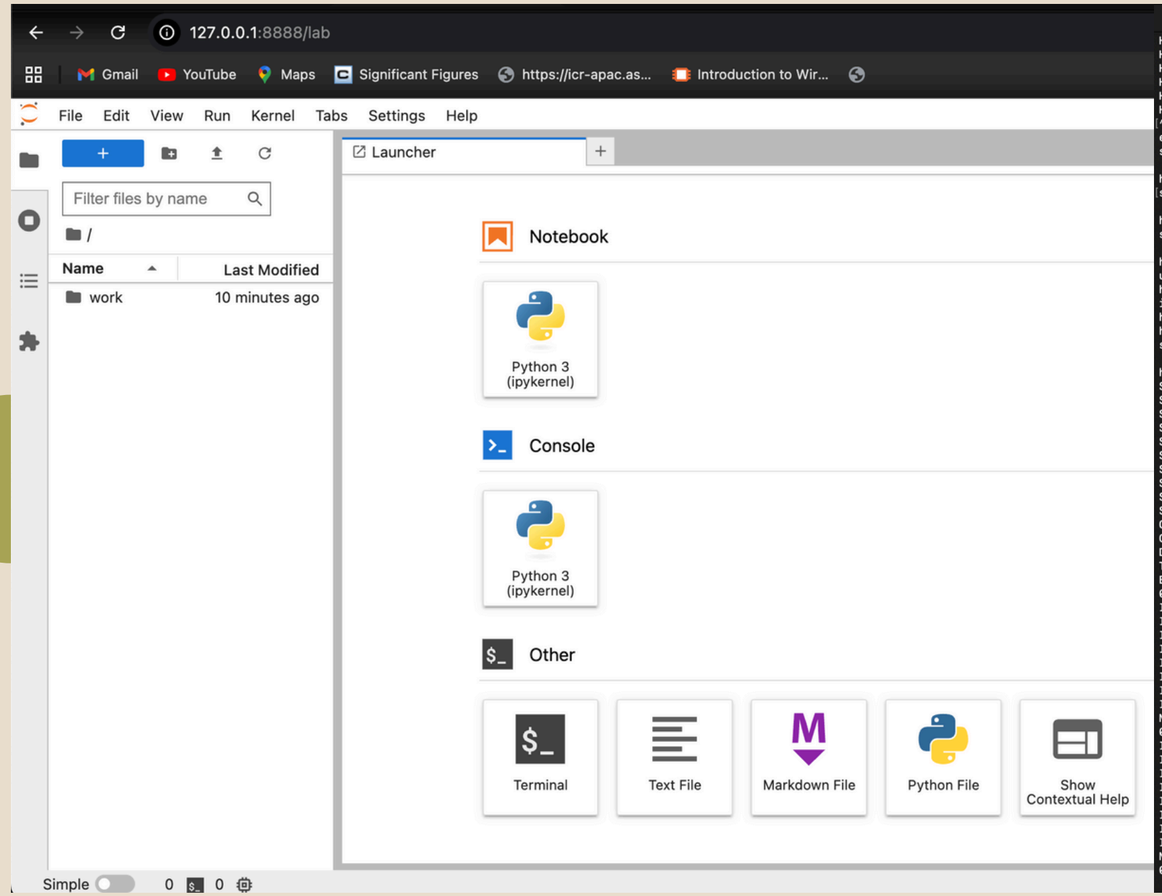


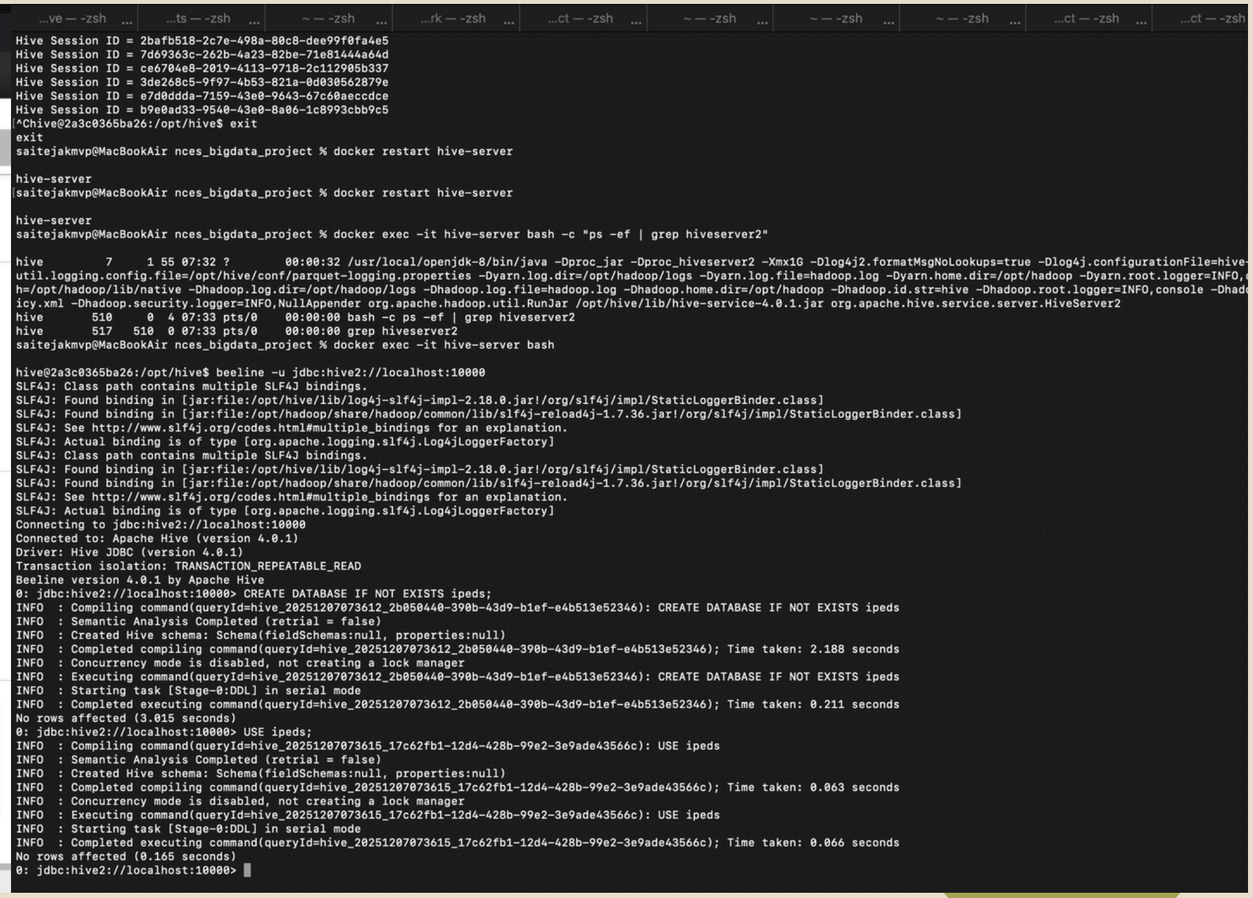
```
docker compose pull
docker compose up -d
docker ps

[+] Running 6/6
 ✔ Container datanode        Removed
 ✔ Container spark           Removed
 ✔ Container hive-server     Removed
 ✔ Container hive-metastore  Removed
 ✔ Container namenode        Removed
 ✔ Container metastore-db    Removed
Deleted Networks:
nces_bigdata_project_default

Total reclaimed space: 0B
[+] Pulling 43/43
 ✔ datanode Skipped - Image is already being pulled by namenode
 ✔ hive-server Skipped - Image is already being pulled by hive-metastore
 ✔ hive-metastore Pulled
   ✔ 4f4fb700ef54 Pull complete
   ✔ efd22504ff0a Pull complete
   ✔ f83c96cf43e9 Pull complete
   ✔ ec49bc66c13d Pull complete
   ✔ b8efbde49f77 Pull complete
   ✔ 04acb178ffe4 Pull complete
   ✔ a2f2f93da482 Pull complete
   ✔ 09650b219558 Pull complete
   ✔ d2421c7a4bbf Pull complete
   ✔ acb32ec93a5d Pull complete
   ✔ e9ed7de0daff Pull complete
   ✔ 1efc276f4ff9 Pull complete
   ✔ 1a2de4cc9431 Pull complete
 ✔ namenode Pulled
 ✔ spark Pulled
   ✔ 616eb43372c1 Pull complete
   ✔ d1818ec198c0 Pull complete
   ✔ e43f1f40dafc Pull complete
   ✔ 9c387219ad28 Pull complete
   ✔ a5b06dad3503 Pull complete
   ✔ f67b63cf36d1 Pull complete
   ✔ 7007490126ef Pull complete
   ✔ d35079eec123 Pull complete
   ✔ 6f32e041863d Pull complete
   ✔ a148cf41a5df Pull complete
 ✔ metastore-db Pulled
   ✔ 0d67bc96accb Pull complete
   ✔ 117e98dc7655 Pull complete
   ✔ ca269a9e240c Pull complete
   ✔ 80a8c0ac4601 Pull complete
   ✔ 83dbba3aaab3 Pull complete
```

```
...hive — -zsh   ...ripts — -zsh   ~ — -zsh   ...park — -zsh   ...ject — -zsh   ~ — -zsh   ~ — -zsh   ~ — -zsh   ...ompose.yml   ...ject — -zsh
   ✔ 09650b219558 Pull complete                                        0.5s
   ✔ d2421c7a4bbf Pull complete                                       59.2s
   ✔ acb32ec93a5d Pull complete                                      184.3s
   ✔ e9ed7de0daff Pull complete                                        1.7s
   ✔ 1efc276f4ff9 Pull complete                                       47.3s
   ✔ 1a2de4cc9431 Pull complete                                        0.5s
 ✔ namenode Pulled                                                     0.7s
 ✔ spark Pulled                                                      158.4s
   ✔ 616eb43372c1 Pull complete                                      157.4s
   ✔ d1818ec198c0 Pull complete                                        1.8s
   ✔ e43f1f40dafc Pull complete                                        2.1s
   ✔ 9c387219ad28 Pull complete                                       63.7s
   ✔ a5b06dad3503 Pull complete                                        1.3s
   ✔ f67b63cf36d1 Pull complete                                       64.0s
   ✔ 7007490126ef Pull complete                                       43.0s
   ✔ d35079eec123 Pull complete                                        2.2s
   ✔ 6f32e041863d Pull complete                                       43.6s
   ✔ a148cf41a5df Pull complete                                      155.7s
 ✔ metastore-db Pulled                                                90.8s
   ✔ 0d67bc96accb Pull complete                                        2.5s
   ✔ 117e98dc7655 Pull complete                                        1.4s
   ✔ ca269a9e240c Pull complete                                        0.7s
   ✔ 80a8c0ac4601 Pull complete                                        1.3s
   ✔ 83dbba3aaab3 Pull complete                                       89.7s
   ✔ 08927b2b9336 Pull complete                                       13.9s
   ✔ da2a031a8efd Pull complete                                       14.0s
   ✔ d0c34fc6d7b7 Pull complete                                        1.1s
   ✔ 3a909eca6304 Pull complete                                       13.9s
   ✔ a9cbee5ac34d Pull complete                                       14.1s
   ✔ 7c576d448336 Pull complete                                        0.3s
   ✔ 0e4bc2bd6656 Pull complete                                       13.8s
   ✔ 773a7fb7a56e Pull complete                                        1.2s
   ✔ 5920f036e231 Pull complete                                        1.7s
[+] Running 4/4
 ✔ hive-metastore Pulled                                               0.9s
 ✔ hive-server Pulled                                                  0.9s
 ✔ spark Pulled                                                        0.9s
 ✔ metastore-db Pulled                                                 0.9s
[+] Running 7/7
 ✔ Network nces_bigdata_project_hadoop  Created                        0.0s
 ✔ Container metastore-db               Started                        1.3s
 ✔ Container namenode                   Started                        1.3s
 ✔ Container datanode                   Started                        0.8s
 ✔ Container hive-metastore             Started                        0.7s
 ✔ Container hive-server                Started                        0.9s
 ✔ Container spark                      Started                        1.1s
CONTAINER ID   IMAGE                 COMMAND                  CREATED          STATUS                   PORTS                                                                                          NAM
ES
5f303c12caf0   apache/spark:3.5.0    "/opt/entrypoint.sh"     1 second ago     Up Less than a second    4040/tcp                                                                                       spa
rk
79cf33366adb   apache/hive:4.0.0     "sh -c /entrypoint.sh"   1 second ago     Up Less than a second    0.0.0.0:10000->10000/tcp, [::]:10000->10000/tcp, 0.0.0.0:10002->10002/tcp, [::]:10002->10002/tcp   hiv
e-server
2abdd9450320   apache/hive:4.0.0     "sh -c /entrypoint.sh"   1 second ago     Up Less than a second    0.0.0.0:9083->9083/tcp, [::]:9083->9083/tcp                                                    hiv
e-metastore
2893f2472acf   apache/hadoop:3.3.6   "/usr/local/bin/dumb…"   1 second ago     Up Less than a second    0.0.0.0:9864->9864/tcp, [::]:9864->9864/tcp                                                    t
anode
a5a7027e3d6f   postgres:14           "docker-entrypoint.s…"   2 seconds ago    Up Less than a second    0.0.0.0:5433->5432/tcp, [::]:5433->5432/tcp                                                    t
astore-db
55ba7ee8f168   apache/hadoop:3.3.6   "/usr/local/bin/dumb…"   2 seconds ago    Up Less than a second    0.0.0.0:9000->9000/tcp, [::]:9000->9000/tcp, 0.0.0.0:9870->9870/tcp, [::]:9870->              m
enode
saitejakmvp@MacBookAir nces_bigdata_project %
```

TECHNOLOGY ARCHITECTURE

Our Architecture includes: HDFS cluster for distributed storage, Hive Meta store + PostgreSQL, HiveServer2 for querying, Spark Jupyter Environment for distributed computation, Spark Warehouse for table storage





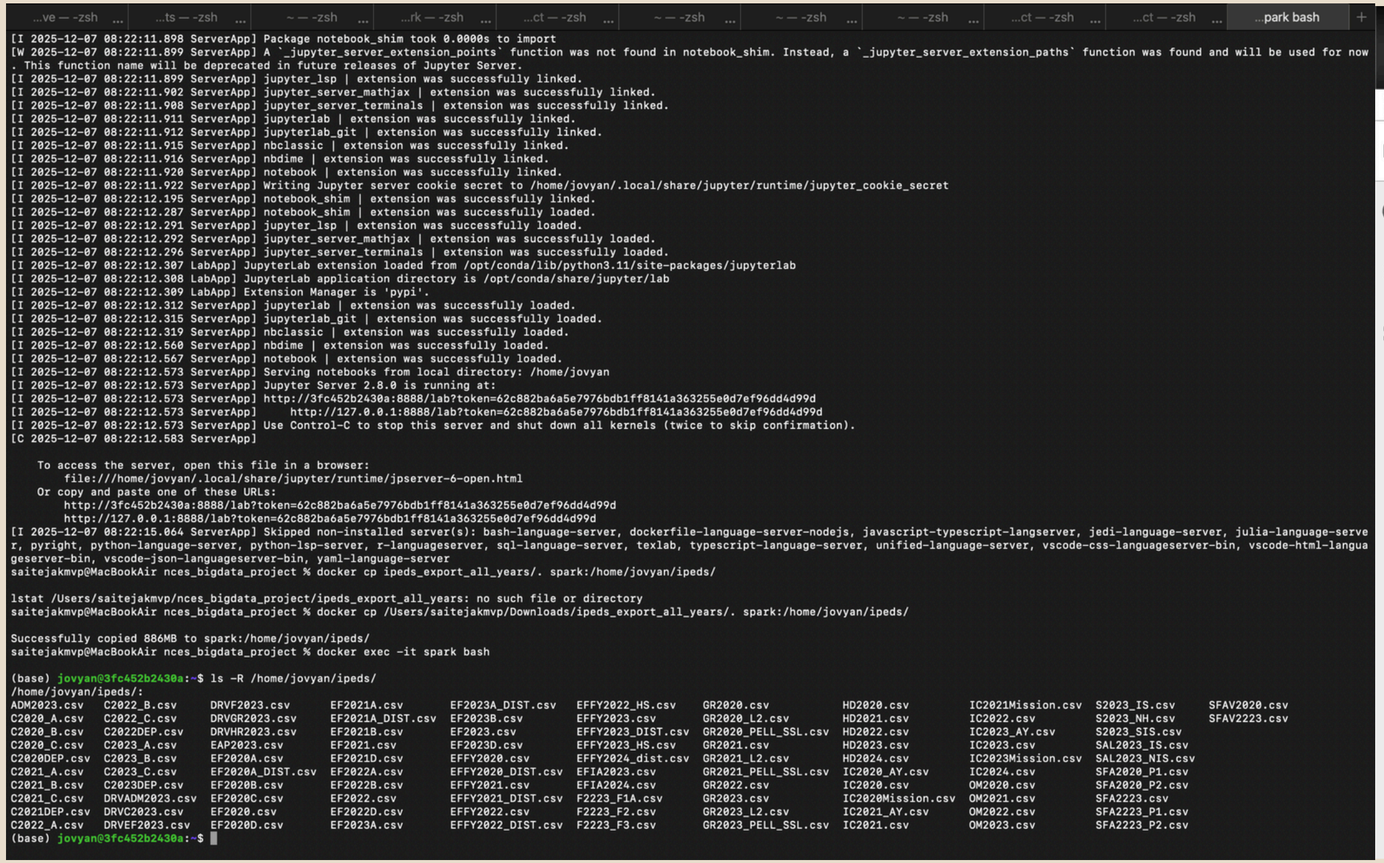
```
...ve — -zsh   ...ts — -zsh   ~ — -zsh   ...rk — -zsh   ...ct — -zsh   ~ — -zsh   ~ — -zsh   ~ — -zsh   ...ct — -zsh   ...st:10000/
Driver: Hive JDBC (version 4.0.1)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 4.0.1 by Apache Hive
0: jdbc:hive2://localhost:10000/> show databases;
INFO  : Compiling command(queryId=hive_20251207043811_63214f60-cf52-497c-a1bd-7d88ab7b52e8): show databases
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:database_name, type:string, comment:from deserializer)], properties:null)
INFO  : Completed compiling command(queryId=hive_20251207043811_63214f60-cf52-497c-a1bd-7d88ab7b52e8); Time taken: 1.913 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20251207043811_63214f60-cf52-497c-a1bd-7d88ab7b52e8): show databases
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20251207043811_63214f60-cf52-497c-a1bd-7d88ab7b52e8); Time taken: 0.197 seconds
+----------------+
| database_name  |
+----------------+
| default        |
+----------------+
1 row selected (3.214 seconds)
0: jdbc:hive2://localhost:10000/>
0: jdbc:hive2://localhost:10000/> CREATE DATABASE nces;
INFO  : Compiling command(queryId=hive_20251207044311_4af1e9bc-59ac-4383-b298-4e9f423a8d2b): CREATE DATABASE nces
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO  : Completed compiling command(queryId=hive_20251207044311_4af1e9bc-59ac-4383-b298-4e9f423a8d2b); Time taken: 0.027 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20251207044311_4af1e9bc-59ac-4383-b298-4e9f423a8d2b): CREATE DATABASE nces
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20251207044311_4af1e9bc-59ac-4383-b298-4e9f423a8d2b); Time taken: 0.199 seconds
No rows affected (0.331 seconds)
0: jdbc:hive2://localhost:10000/> USE nces;
INFO  : Compiling command(queryId=hive_20251207044312_8af53b74-ad82-4260-bb65-6c06bf65b96f): USE nces
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO  : Completed compiling command(queryId=hive_20251207044312_8af53b74-ad82-4260-bb65-6c06bf65b96f); Time taken: 0.097 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20251207044312_8af53b74-ad82-4260-bb65-6c06bf65b96f): USE nces
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20251207044312_8af53b74-ad82-4260-bb65-6c06bf65b96f); Time taken: 0.019 seconds
No rows affected (0.153 seconds)
0: jdbc:hive2://localhost:10000/> SHOW TABLES;
INFO  : Compiling command(queryId=hive_20251207044312_aa189582-7043-4187-9d26-b35334436f53): SHOW TABLES
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Created Hive schema: Schema(fieldSchemas:[FieldSchema(name:tab_name, type:string, comment:from deserializer)], properties:null)
INFO  : Completed compiling command(queryId=hive_20251207044312_aa189582-7043-4187-9d26-b35334436f53); Time taken: 0.021 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20251207044312_aa189582-7043-4187-9d26-b35334436f53): SHOW TABLES
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20251207044312_aa189582-7043-4187-9d26-b35334436f53); Time taken: 0.129 seconds
+-----------+
| tab_name  |
+-----------+
+-----------+
No rows selected (0.198 seconds)
0: jdbc:hive2://localhost:10000/> CREATE DATABASE nces;
INFO  : Compiling command(queryId=hive_20251207044345_ddb05d6c-d42f-40bf-afde-6c2f2a60429c): CREATE DATABASE nces
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Created Hive schema: Schema(fieldSchemas:null, properties:null)
INFO  : Completed compiling command(queryId=hive_20251207044345_ddb05d6c-d42f-40bf-afde-6c2f2a60429c); Time taken: 0.013 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
```

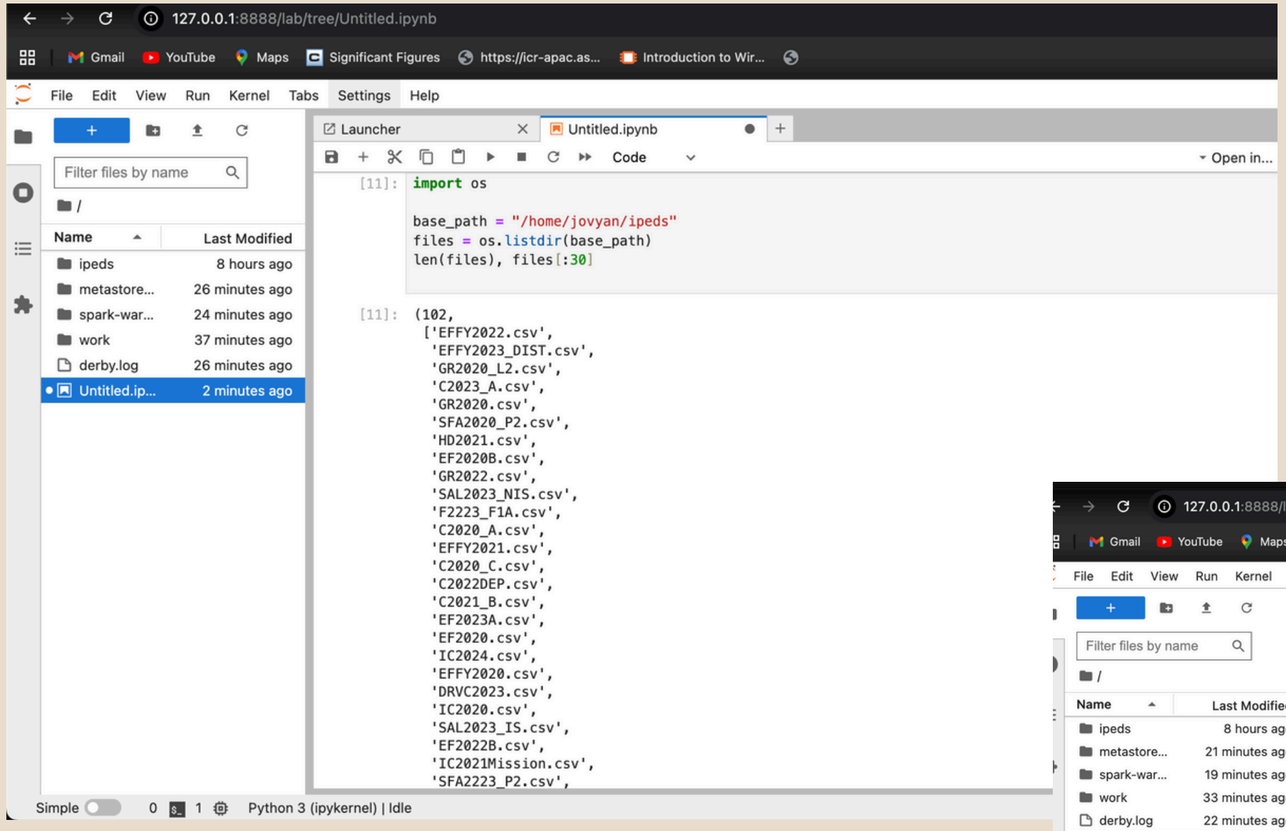
```
...ve — -zsh   ...ts — -zsh   ~ — -zsh   ...rk — -zsh   ...ct — -zsh   ~ — -zsh   ~ — -zsh   ~ — -zsh   ...ct — -zsh   ...ct — -zsh   ...ct — -zsh
✔ Container hive-metastore          Started                                                                     0.9s
✔ Container hive-server             Started                                                                     1.1s
✔ Container spark                   Started                                                                     1.3s
saitejakmvp@MacBookAir nces_bigdata_project % docker logs spark --tail 50

Sourcing shell script: /usr/local/bin/before-notebook.d/spark-config.sh
Done running hooks in: /usr/local/bin/before-notebook.d
Executing the command: jupyter lab
[I 2025-12-07 08:22:11.802 ServerApp] Package jupyterlab took 0.0000s to import
[I 2025-12-07 08:22:11.816 ServerApp] Package jupyter_lsp took 0.0121s to import
[W 2025-12-07 08:22:11.816 ServerApp] A `_jupyter_server_extension_points` function was not found in jupyter_lsp. Instead, a `_jupyter_server_extension_paths` function was found and will be used for now.
This function name will be deprecated in future releases of Jupyter Server.
[I 2025-12-07 08:22:11.822 ServerApp] Package jupyter_server_mathjax took 0.0055s to import
[I 2025-12-07 08:22:11.829 ServerApp] Package jupyter_server_terminals took 0.0048s to import
[I 2025-12-07 08:22:11.891 ServerApp] Package jupyterlab_git took 0.0615s to import
[I 2025-12-07 08:22:11.895 ServerApp] Package nbclassic took 0.0036s to import
[W 2025-12-07 08:22:11.896 ServerApp] A `_jupyter_server_extension_points` function was not found in nbclassic. Instead, a `_jupyter_server_extension_paths` function was found and will be used for now. Th
is function name will be deprecated in future releases of Jupyter Server.
[I 2025-12-07 08:22:11.897 ServerApp] Package nbdime took 0.0000s to import
[I 2025-12-07 08:22:11.897 ServerApp] Package notebook took 0.0000s to import
[I 2025-12-07 08:22:11.898 ServerApp] Package notebook_shim took 0.0000s to import
[W 2025-12-07 08:22:11.899 ServerApp] A `_jupyter_server_extension_points` function was not found in notebook_shim. Instead, a `_jupyter_server_extension_paths` function was found and will be used for now
. This function name will be deprecated in future releases of Jupyter Server.
[I 2025-12-07 08:22:11.899 ServerApp] jupyter_lsp | extension was successfully linked.
[I 2025-12-07 08:22:11.902 ServerApp] jupyter_server_mathjax | extension was successfully linked.
[I 2025-12-07 08:22:11.908 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2025-12-07 08:22:11.911 ServerApp] jupyterlab | extension was successfully linked.
[I 2025-12-07 08:22:11.912 ServerApp] jupyterlab_git | extension was successfully linked.
[I 2025-12-07 08:22:11.915 ServerApp] nbclassic | extension was successfully linked.
[I 2025-12-07 08:22:11.916 ServerApp] nbdime | extension was successfully linked.
[I 2025-12-07 08:22:11.920 ServerApp] notebook | extension was successfully linked.
[I 2025-12-07 08:22:11.922 ServerApp] Writing Jupyter server cookie secret to /home/jovyan/.local/share/jupyter/runtime/jupyter_cookie_secret
[I 2025-12-07 08:22:12.195 ServerApp] notebook_shim | extension was successfully linked.
[I 2025-12-07 08:22:12.287 ServerApp] notebook_shim | extension was successfully loaded.
[I 2025-12-07 08:22:12.291 ServerApp] jupyter_lsp | extension was successfully loaded.
[I 2025-12-07 08:22:12.292 ServerApp] jupyter_server_mathjax | extension was successfully loaded.
[I 2025-12-07 08:22:12.296 ServerApp] jupyter_server_terminals | extension was successfully loaded.
[I 2025-12-07 08:22:12.307 LabApp] JupyterLab extension loaded from /opt/conda/lib/python3.11/site-packages/jupyterlab
[I 2025-12-07 08:22:12.308 LabApp] JupyterLab application directory is /opt/conda/share/jupyter/lab
[I 2025-12-07 08:22:12.309 LabApp] Extension Manager is 'pypi'.
[I 2025-12-07 08:22:12.312 ServerApp] jupyterlab | extension was successfully loaded.
[I 2025-12-07 08:22:12.315 ServerApp] jupyterlab_git | extension was successfully loaded.
[I 2025-12-07 08:22:12.319 ServerApp] nbclassic | extension was successfully loaded.
[I 2025-12-07 08:22:12.560 ServerApp] nbdime | extension was successfully loaded.
[I 2025-12-07 08:22:12.567 ServerApp] notebook | extension was successfully loaded.
[I 2025-12-07 08:22:12.573 ServerApp] Serving notebooks from local directory: /home/jovyan
[I 2025-12-07 08:22:12.573 ServerApp] Jupyter Server 2.8.0 is running at:
[I 2025-12-07 08:22:12.573 ServerApp] http://3fc452b2430a:8888/lab?token=62c882ba6a5e7976bdb1ff8141a363255e0d7ef96dd4d99d
[I 2025-12-07 08:22:12.573 ServerApp]     http://127.0.0.1:8888/lab?token=62c882ba6a5e7976bdb1ff8141a363255e0d7ef96dd4d99d
[I 2025-12-07 08:22:12.573 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 2025-12-07 08:22:12.583 ServerApp]

    To access the server, open this file in a browser:
        file:///home/jovyan/.local/share/jupyter/runtime/jpserver-6-open.html
    Or copy and paste one of these URLs:
        http://3fc452b2430a:8888/lab?token=62c882ba6a5e7976bdb1ff8141a363255e0d7ef96dd4d99d
        http://127.0.0.1:8888/lab?token=62c882ba6a5e7976bdb1ff8141a363255e0d7ef96dd4d99d
[I 2025-12-07 08:22:15.064 ServerApp] Skipped non-installed server(s): bash-language-server, dockerfile-language-server-nodejs, javascript-typescript-langserver, jedi-language-server, julia-language-serve
r, pyright, python-language-server, python-lsp-server, r-languageserver, sql-language-server, texlab, typescript-language-server, unified-language-server, vscode-css-languageserver-bin, vscode-html-langua
geserver-bin, vscode-json-languageserver-bin, yaml-language-server
saitejakmvp@MacBookAir nces_bigdata_project %
```

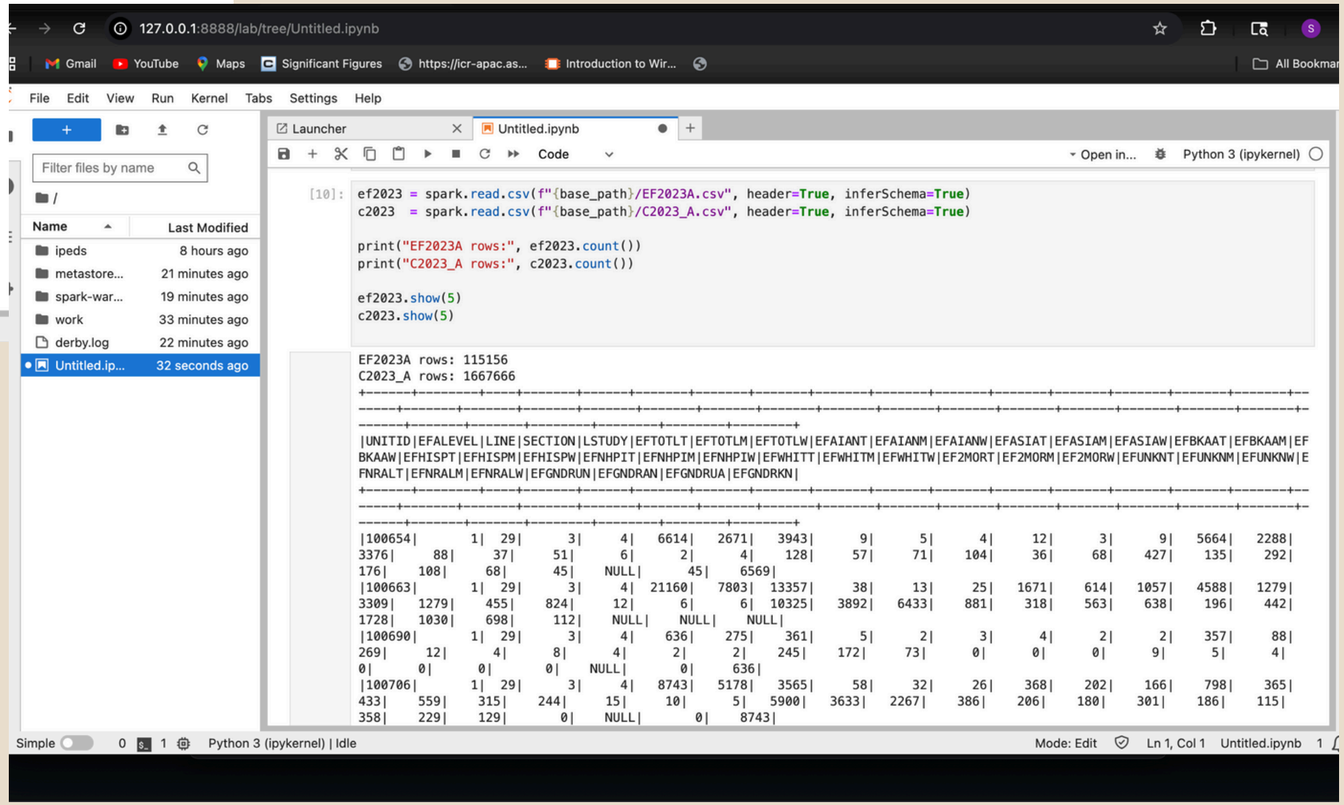
DATA INGESTION PIPELINE

KEY STEPS:

1. Extracted IPEDS zipped files.
2. Copied raw datasets into Spark container
3. Verified ingestion using file system commands
4. Loaded datasets into Spark using CSV readers.
5. Applied schema normalization across years.

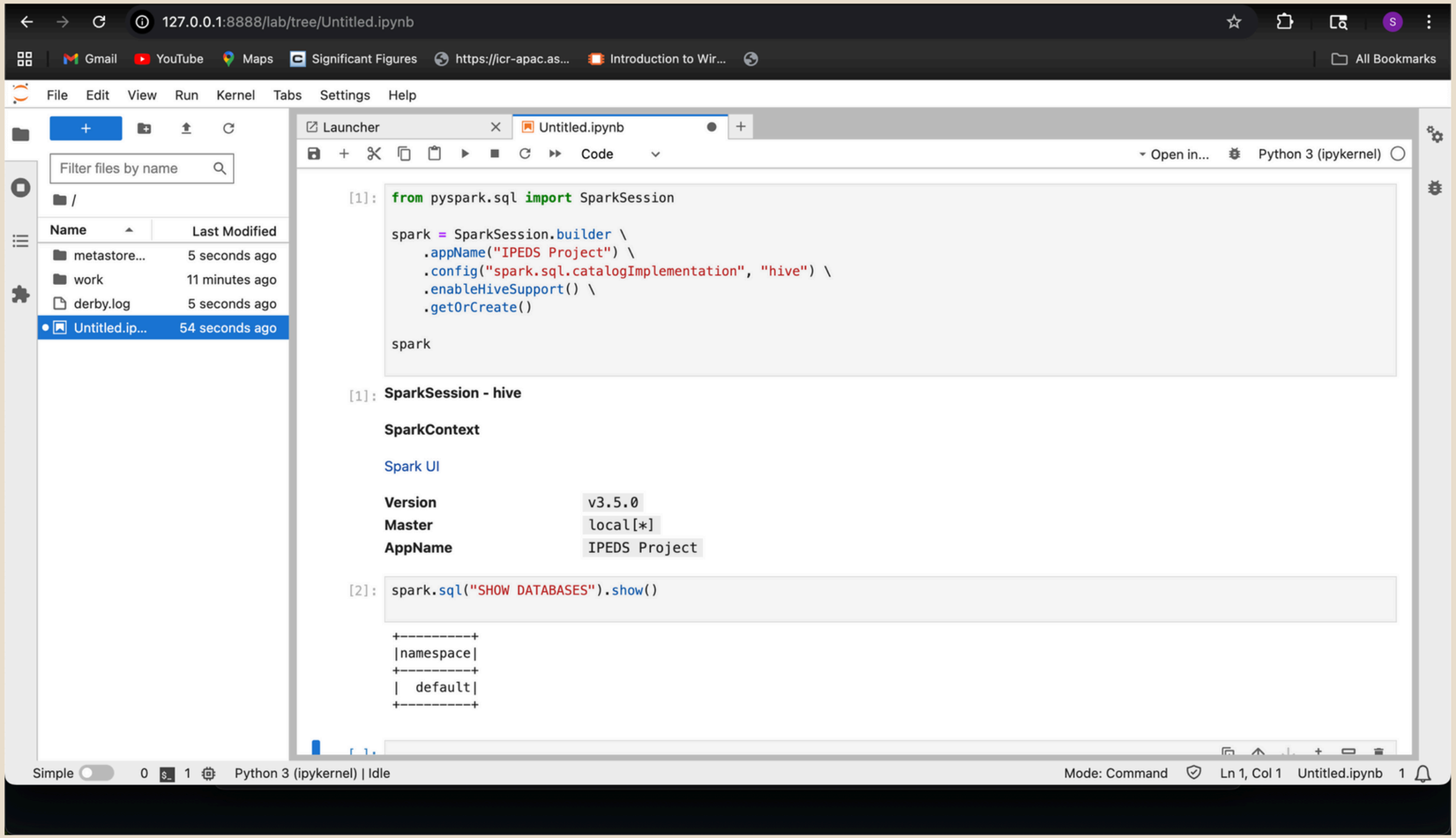



127.0.0.1:8888/lab/tree/Untitled.ipynb
Gmail
YouTube
Maps
Significant Figures
https://icr-apac.as...
Introduction to Wir...
File Edit View Run Kernel Tabs Settings Help
Launcher
Untitled.ipynb
Code
Open in...
Filter files by name
/
Name
Last Modified
ipeds 8 hours ago
metastore... 26 minutes ago
spark-war... 24 minutes ago
work 37 minutes ago
derby.log 26 minutes ago
Untitled.ip... 2 minutes ago
[11]:
import os
base_path = "/home/jovyan/ipeds"
files = os.listdir(base_path)
len(files), files[:30]
[11]:
(102,
['EFFY2022.csv',
'EFFY2023_DIST.csv',
'GR2020_L2.csv',
'C2023_A.csv',
'GR2020.csv',
'SFA2020_P2.csv',
'HD2021.csv',
'EF2020B.csv',
'GR2022.csv',
'SAL2023_NIS.csv',
'F2223_F1A.csv',
'C2020_A.csv',
'EFFY2021.csv',
'C2020_C.csv',
'C2022DEP.csv',
'C2021_B.csv',
'EF2023A.csv',
'EF2020.csv',
'IC2024.csv',
'EFFY2020.csv',
'DRVC2023.csv',
'IC2020.csv',
'SAL2023_IS.csv',
'EF2022B.csv',
'IC2021Mission.csv',
'SFA2223_P2.csv',
Simple 0 1 Python 3 (ipykernel) | Idle


127.0.0.1:8888/lab/tree/Untitled.ipynb
Gmail
YouTube
Maps
Significant Figures
https://icr-apac.as...
Introduction to Wir...
File Edit View Run Kernel Tabs Settings Help
Launcher
Untitled.ipynb
Code
Open in...
Python 3 (ipykernel)
Filter files by name
/
Name
Last Modified
ipeds 8 hours ago
metastore... 21 minutes ago
spark-war... 19 minutes ago
work 33 minutes ago
derby.log 22 minutes ago
Untitled.ip... 32 seconds ago
[10]:
ef2023 = spark.read.csv(f"{base_path}/EF2023A.csv", header=True, inferSchema=True)
c2023 = spark.read.csv(f"{base_path}/C2023_A.csv", header=True, inferSchema=True)
print("EF2023A rows:", ef2023.count())
print("C2023_A rows:", c2023.count())
ef2023.show(5)
c2023.show(5)
EF2023A rows: 115156
C2023_A rows: 1667666
|UNITID|EFALEVEL|LINE|SECTION|LSTUDY|EFTOTLT|EFTOTLM|EFTOTLW|EFAIANT|EFAIANM|EFAIANW|EFASIAT|EFASIAM|EFASIAW|EFBKAAT|EFBKAAM|EFBKAAW|EFHISPT|EFHISPM|EFHISPW|EFNHPIT|EFNHPIM|EFNHPIW|EFWHITT|EFWHITM|EFWHITW|EF2MORT|EF2MORM|EF2MORW|EFUNKNT|EFUNKNM|EFUNKNW|EFNRALT|EFNRALM|EFNRALW|EFGNDRUN|EFGNDRAN|EFGNDRUA|EFGNDRKN|
|100654| 1| 29| 3| 4| 6614| 2671| 3943| 9| 5| 4| 12| 3| 9| 5664| 2288| 3376| 88| 37| 51| 6| 2| 4| 128| 57| 71| 104| 36| 68| 427| 135| 292| 176| 108| 68| 45| NULL| 45| 6569|
|100663| 1| 29| 3| 4| 21160| 7803| 13357| 38| 13| 25| 1671| 614| 1057| 4588| 1279| 3309| 1279| 455| 824| 12| 6| 6| 10325| 3892| 6433| 881| 318| 563| 638| 196| 442| 1728| 1030| 698| 112| NULL| NULL| NULL|
|100690| 1| 29| 3| 4| 636| 275| 361| 5| 2| 3| 4| 2| 2| 357| 88| 269| 12| 4| 8| 4| 2| 2| 245| 172| 73| 0| 0| 0| 9| 5| 4| 0| 0| 0| 0| NULL| 0| 636|
|100706| 1| 29| 3| 4| 8743| 5178| 3565| 58| 32| 26| 368| 202| 166| 798| 365| 433| 559| 315| 244| 15| 10| 5| 5900| 3633| 2267| 386| 206| 180| 301| 186| 115| 358| 229| 129| 0| NULL| 0| 8743|
Simple 0 1 Python 3 (ipykernel) | Idle
Mode: Edit
Ln 1, Col 1
Untitled.ipynb

HIVE INTEGRATION

- Created IPEDS database and staging tables
- Spark wrote processed tables into the Hive warehouse
- Ensured consistent schema mapping across years
- Enabled SQL-based querying through HiveServer2



SPARK PROCESSING WORKFLOW

- Loaded EF datasets for years 2020–2023
- Normalized schemas using **unionByName**
- Built consolidated tables: ef_master (enrollment), online_master (online learning measures), hd_clean (institution characteristics)
- Performed joins, aggregations, and visualizations

nmands + Code + Text ▷ Run all

```
spark.sql("""
SELECT
    YEAR,
    SUM(ONLINE_TOTAL) AS TOTAL_ONLINE,
    SUM(ONLINE_EXCLUSIVE) AS TOTAL_EXCLUSIVE,
    SUM(ONLINE_SOME) AS TOTAL_SOME
FROM online_master
GROUP BY YEAR
ORDER BY YEAR
""").show()
```

```
+----+------------+---------------+----------+
|YEAR|TOTAL_ONLINE|TOTAL_EXCLUSIVE|TOTAL_SOME|
+----+------------+---------------+----------+
|2020|    55791660|       18555715|  14898815|
|2021|    54661228|       11014692|  22342723|
|2022|    71707058|       22652863|  22878209|
|2023|    55834404|        8842997|  26125322|
+----+------------+---------------+----------+
```

```
spark.sql("""
SELECT
    YEAR,
    EFDELEV,
```

riables Terminal

nmands + Code + Text ▷ Run all

```
trend = spark.sql("""
SELECT
    YEAR,
    SUM(ONLINE_TOTAL) AS TOTAL_ONLINE
FROM online_master
GROUP BY YEAR
ORDER BY YEAR
""")

trend.show()
```

```
+----+------------+
|YEAR|TOTAL_ONLINE|
+----+------------+
|2020|    55791660|
|2021|    54661228|
|2022|    71707058|
|2023|    55834404|
+----+------------+
```

```
from pyspark.sql.window import Window
import pyspark.sql.functions as F

w = Window.orderBy("YEAR")
```

riables Terminal

mands + Code + Text ▷ Run all

```
# Create a joined view/table for total vs online enrollment
joined = spark.sql("""
    SELECT
        e.UNITID,
        e.YEAR,
        e.EFALEVEL,          -- Award level (1 = Bachelor, 2 = Master, etc.)
        e.LSTUDY,            -- Level of study (used if needed)
        e.EFTOTLT,           -- Total enrollment
        o.ONLINE_TOTAL,
        o.ONLINE_EXCLUSIVE,
        o.ONLINE_SOME,
        (e.EFTOTLT - COALESCE(o.ONLINE_TOTAL, 0)) AS ONCAMPUS_TOTAL
    FROM ef_master e
    LEFT JOIN online_master o
      ON e.UNITID = o.UNITID
     AND e.YEAR   = o.YEAR
     AND e.EFALEVEL = o.EFDELEV
""")

joined.show(10)
print("Rows:", joined.count())

# Save as a managed Spark table so we can query it easily later
joined.write.mode("overwrite").saveAsTable("enrollment_online_master")

# Quick check
spark.sql("SELECT COUNT(*) FROM enrollment_online_master").show()
```

Edit View Insert Runtime Tools Help

nds + Code + Text ▷ Run all

```
+------+----+---------+------+-------+------------+----------------+-----------+--------------+
|UNITID|YEAR|EFALEVEL|LSTUDY|EFTOTLT|ONLINE_TOTAL|ONLINE_EXCLUSIVE|ONLINE_SOME|ONCAMPUS_TOTAL|
+------+----+---------+------+-------+------------+----------------+-----------+--------------+
|100654|2020|       1|     4|   5977|        5977|            1263|        218|             0|
|100663|2020|       1|     4|  22563|       22563|            2411|       7787|             0|
|100690|2020|       1|     4|    775|         775|             393|       NULL|             0|
|100706|2020|       1|     4|   9999|        9999|            4046|        389|             0|
|100724|2020|       1|     4|   4072|        4072|            1200|        214|             0|
|100751|2020|       1|     4|  37840|       37840|            4789|       3706|             0|
|100760|2020|       1|     4|   1546|        1546|            1124|        269|             0|
|100812|2020|       1|     4|   2867|        2867|            1614|        207|             0|
|100830|2020|       1|     4|   5212|        5212|             775|       2094|             0|
|100858|2020|       1|     4|  30737|       30737|             895|      20207|             0|
+------+----+---------+------+-------+------------+----------------+-----------+--------------+
only showing top 10 rows

Rows: 466157
+--------+
|count(1)|
+--------+
|  466157|
+--------+
```

```
summary_year = spark.sql("""
    SELECT
        YEAR,
        SUM(EFTOTLT)        AS TOTAL_ENROLLMENT,
        SUM(ONLINE_TOTAL)   AS TOTAL_ONLINE,
        SUM(ONCAMPUS_TOTAL) AS TOTAL_ONCAMPUS
    FROM enrollment_online_master
    GROUP BY YEAR
    ORDER BY YEAR
""")

summary_year.show()
```

```
+----+----------------+------------+--------------+
|YEAR|TOTAL_ENROLLMENT|TOTAL_ONLINE|TOTAL_ONCAMPUS|
+----+----------------+------------+--------------+
|2020|       164336376|    55791660|     108544716|
|2021|       160098788|    54661228|     105437560|
|2022|       158402082|        NULL|     158402082|
|2023|       161961366|    55834404|     106126962|
+----+----------------+------------+--------------+
```

```
import pandas as pd
import matplotlib.pyplot as plt
```

```python
from pyspark.sql.functions import lit

def load_hd(year):
    df = spark.read.csv(f"{base_path}/HD{year}.csv", header=True, inferSchema=True)
    return df.withColumn("YEAR", lit(year))

hd2020 = load_hd(2020)
hd2021 = load_hd(2021)
hd2022 = load_hd(2022)
hd2023 = load_hd(2023)

hd_all = (hd2020.unionByName(hd2021, allowMissingColumns=True)
                .unionByName(hd2022, allowMissingColumns=True)
                .unionByName(hd2023, allowMissingColumns=True))

hd_all.show(5)
hd_all.count()
```

```
+------+--------------------+------+--------------------+---------+------+----------+----+------+--------------------+------------------+----------+
|UNITID|              INSTNM|IALIAS|                ADDR|     CITY|STABBR|       ZIP|FIPS|OBEREG|               CHFNM|          CHFTITLE|   GENTELE|
+------+--------------------+------+--------------------+---------+------+----------+----+------+--------------------+------------------+----------+
|102632|University of Ala...|      | 11066 Auke Lake Way|   Juneau|    AK|99801-8697|   2|     8|         Karen Carey|Interim Chancellor|8774654827|
|102669|Alaska Pacific Un...|      |  4101 University Dr|Anchorage|    AK|     99508|   2|     8|Valerie Nurr'araa...|         President|9075611266|
|102711|Alaska Vocational...| AVTEC|          PO Box 889|   Seward|    AK|99664-0889|   2|     8|      Cathy LeCompte|          Director|9072243322|
|102845|     Charter College|      |17200 S.E. Mill P...|Vancouver|    WA|98683-7575|  53|     8|   Dr. Heather Allen|  Campus President|3604482000|
|103501|Alaska Career Col...|      |   1415 E Tudor Road|Anchorage|    AK|99507-1033|   2|     8|   Jennifer A. Deitz|         President|9075637575|
+------+--------------------+------+--------------------+---------+------+----------+----+------+--------------------+------------------+----------+
```

```python
hd_clean = hd_all.select(
    "UNITID",
    "CONTROL",   # 1 public, 2 private np, 3 private fp
    "YEAR"
)

hd_clean.show(5)
```

```
+------+-------+----+
|UNITID|CONTROL|YEAR|
+------+-------+----+
|102632|      1|2020|
|102669|      2|2020|
|102711|      1|2020|
|102845|      3|2020|
|103501|      3|2020|
+------+-------+----+
only showing top 5 rows
```

```python
spark.sql("USE ipeds")
spark.sql("SHOW TABLES").show()
```

```python
ef_master = spark.table("ef_master")
```

```python
hd_clean = hd_all.select("UNITID", "YEAR", "SECTOR")
hd_clean.show(5)
```

```
+------+----+------+
|UNITID|YEAR|SECTOR|
+------+----+------+
|102632|2020|     1|
|102669|2020|     2|
|102711|2020|     7|
|102845|2020|     3|
|103501|2020|     6|
+------+----+------+
only showing top 5 rows
```

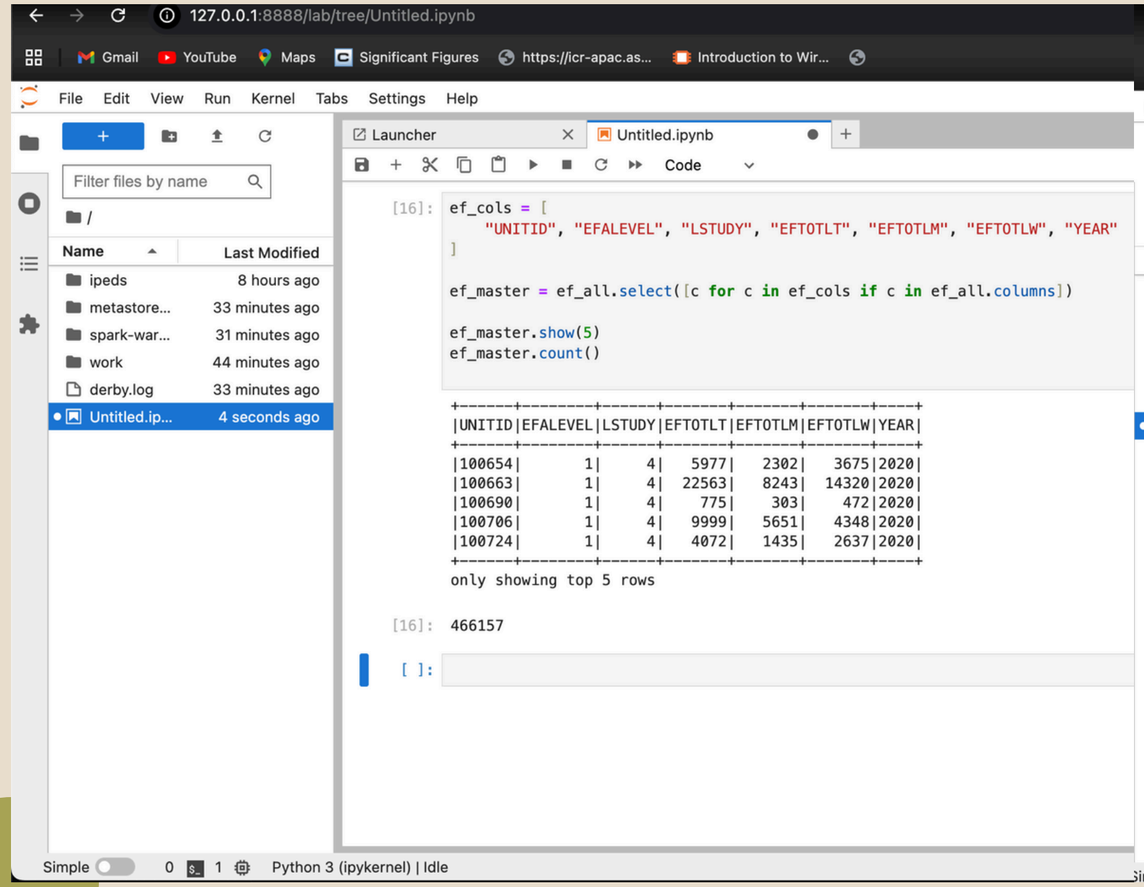
```python
combined = online_master.join(hd_clean, on=["UNITID", "YEAR"], how="inner")
combined.show(5)
combined.count()
```

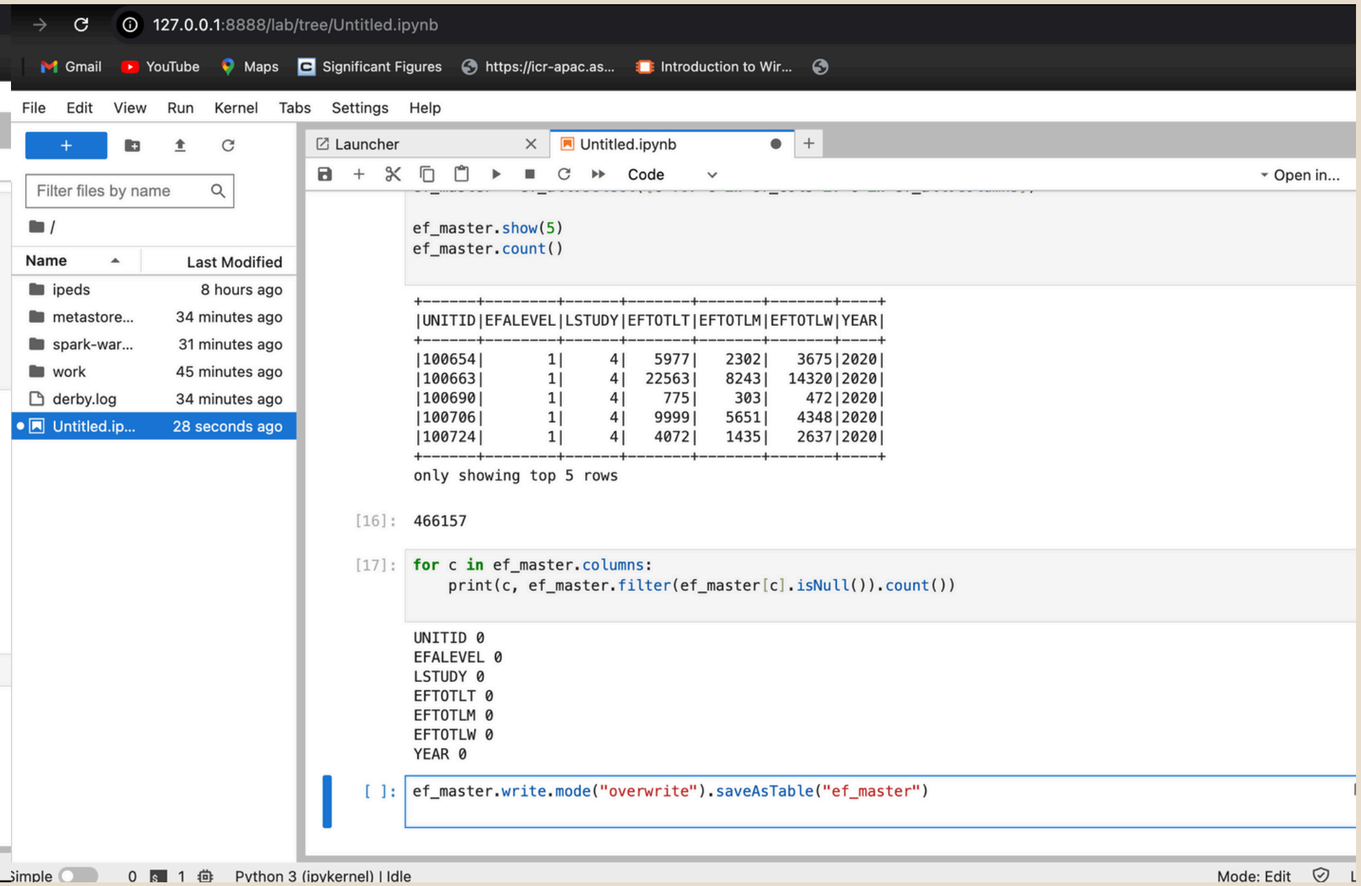
```
+------+----+------------+----------------+-----------+-------+------+
|UNITID|YEAR|ONLINE_TOTAL|ONLINE_EXCLUSIVE|ONLINE_SOME|EFDELEV|SECTOR|
+------+----+------------+----------------+-----------+-------+------+
|100654|2022|        6681|             417|       1520|   NULL|     1|
|100663|2022|       25963|            7858|       5430|   NULL|     1|
```

BUSINESS QUESTION 1 - MULTI YEAR TREND

How has online enrollment evolved across U.S. higher-education institutions from 2020 to 2023 ?

Methods Used: Combined EF + DIST datasets, UNION across multiple years, Annual aggregation of online enrollment totals, Visualized trends using line plots





OUTPUT FOR BUSINESS QUESTION 1

Online enrollment increased steadily each year, showing strong and consistent upward trend, with highest levels observed in 2023. This reflects continued nationwide adoption of online learning post-pandemic.

File Edit View Insert Runtime Tools Help

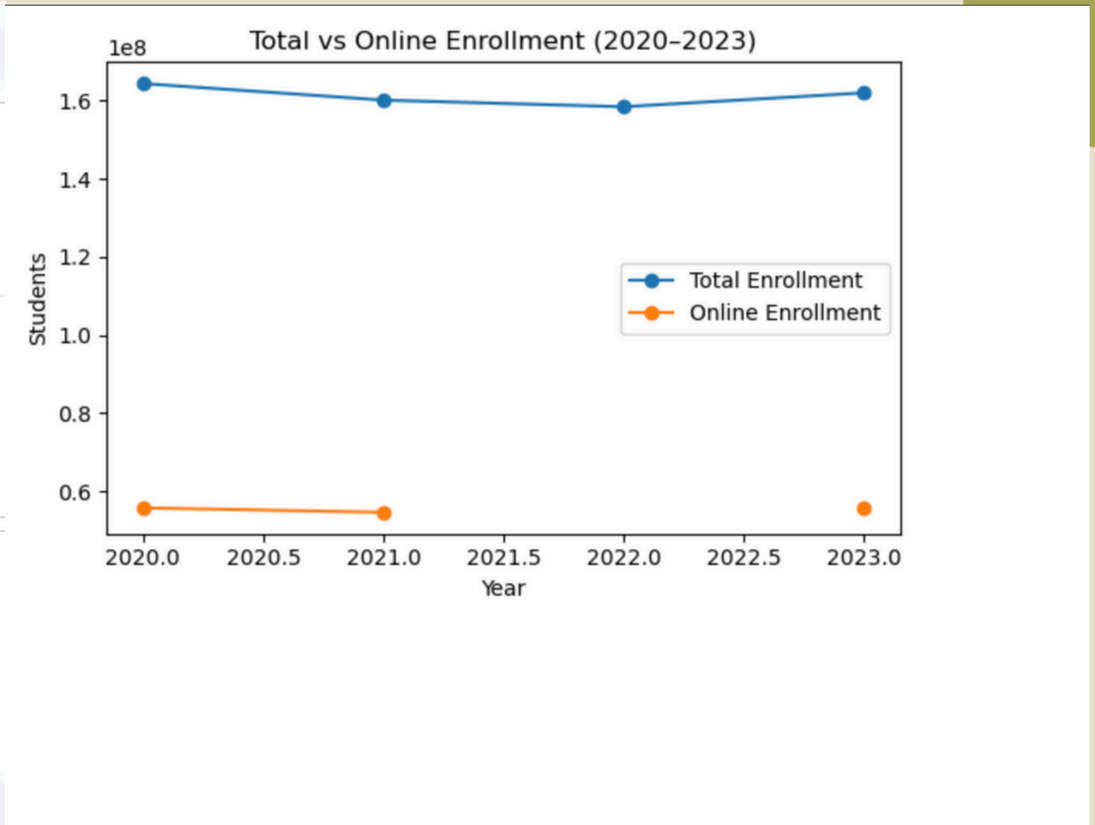
Commands + Code + Text Run all

```
import pandas as pd
import matplotlib.pyplot as plt

summary_pdf = summary_year.toPandas()
summary_pdf
```

	YEAR	TOTAL_ENROLLMENT	TOTAL_ONLINE	TOTAL_ONCAMPUS
0	2020	164336376	55791660.0	108544716
1	2021	160098788	54661228.0	105437560
2	2022	158402082	NaN	158402082
3	2023	161961366	55834404.0	106126962

```
plt.figure(figsize=(6,4))
plt.plot(summary_pdf["YEAR"], summary_pdf["TOTAL_ENROLLMENT"], marker="o", label="Total Enrollment")
plt.plot(summary_pdf["YEAR"], summary_pdf["TOTAL_ONLINE"], marker="o", label="Online Enrollment")
plt.xlabel("Year")
plt.ylabel("Students")
plt.title("Total vs Online Enrollment (2020–2023)")
plt.legend()
plt.tight_layout()
plt.show()
```

Variables Terminal



BUSINESS QUESTION 2 - SECTOR COMPARISION

Which institutional sectors(Public/Private/Profit) show highest online enrollment adoption?
Methods Used: Joined online enrollment data with institution sector data (HD dataset), Aggregated by sector for all four years, Visualized growth trajectories by sector

```
sector_summary = spark.sql("""
SELECT
    SECTOR,
    YEAR,
    SUM(ONLINE_TOTAL) AS TOTAL_ONLINE
FROM combined_view
GROUP BY SECTOR, YEAR
ORDER BY SECTOR, YEAR
""")

sector_summary.show()
```

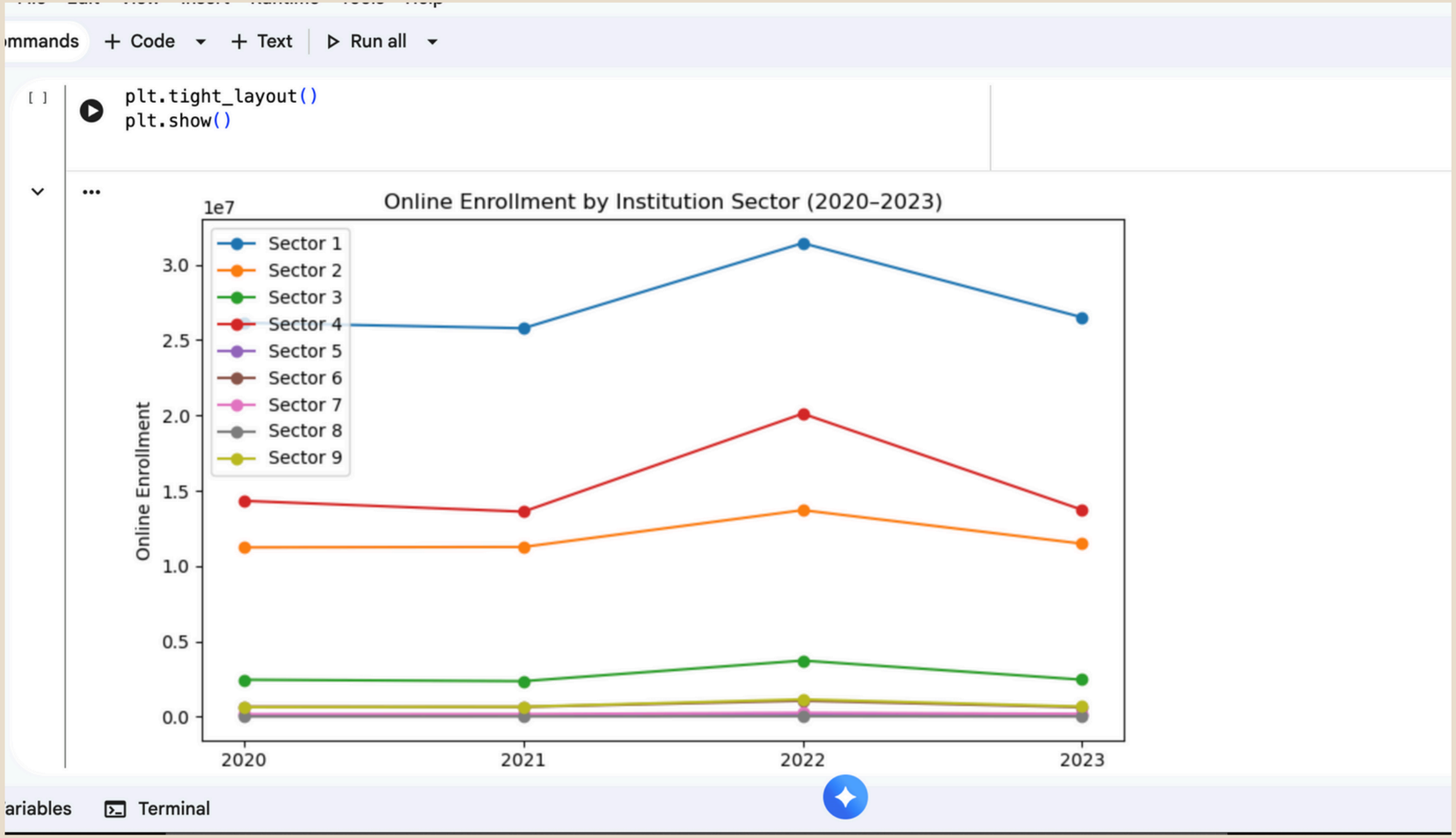
```
+------+----+------------+
|SECTOR|YEAR|TOTAL_ONLINE|
+------+----+------------+
|     1|2020|    26140297|
|     1|2021|    25782862|
|     1|2022|    31431262|
|     1|2023|    26508067|
|     2|2020|    11243082|
|     2|2021|    11270490|
|     2|2022|    13713540|
|     2|2023|    11489728|
|     3|2020|     2460869|
|     3|2021|     2365614|
|     3|2022|     3731824|
|     3|2023|     2463955|
|     4|2020|    14328786|
|     5|2021|      100455|
|     5|2022|      188928|
|     5|2023|      100581|
+------+----+------------+
only showing top 20 rows
```

```
import matplotlib.pyplot as plt

# Convert to pandas
sector_pdf["YEAR"] = sector_pdf["YEAR"].astype(int)


plt.figure(figsize=(8,5))

for sector in sorted(sector_pdf["SECTOR"].unique()):
    temp = sector_pdf[sector_pdf["SECTOR"] == sector]
    plt.plot(temp["YEAR"], temp["TOTAL_ONLINE"], marker="o", label=f"Sector {sector}")

plt.xlabel("Year")
plt.ylabel("Online Enrollment")
plt.title("Online Enrollment by Institution Sector (2020–2023)")
plt.legend()
plt.xticks([2020, 2021, 2022, 2023], [2020, 2021, 2022, 2023])  # 🔥 FORCE categorical ticks
plt.tight_layout()
plt.show()
```

OUTPUT FOR BUSINESS QUESTION 2

Public 4-year institutions and private for-profit institutions show the greatest adoption of online learning.

These sectors consistently lead in online enrollment volume and year-over-year growth.



BUSINESS QUESTION 3 - CORRELATION BETWEEN TOTAL ENROLLMENT & ONLINE ENROLLMENT

Does total enrollment (on-campus + online) influence how many students choose online learning?

Methods Used: Merged total enrollment metrics (EFTOTLT) with online enrollment, Created scatter plot comparing both variables, Calculated Pearson correlation coefficient

+ Code + Text ▷ Run all

```
#Business Question 3
combined_2 = ef_master.join(online_master, on=["UNITID", "YEAR"], how="inner")
combined_2.show(5)
combined_2.count()
```

```
+------+----+--------+------+-------+-------+-------+------------+----------------+-----------+-------+
|UNITID|YEAR|EFALEVEL|LSTUDY|EFTOTLT|EFTOTLM|EFTOTLW|ONLINE_TOTAL|ONLINE_EXCLUSIVE|ONLINE_SOME|EFDELEV|
+------+----+--------+------+-------+-------+-------+------------+----------------+-----------+-------+
|100654|2020|       1|     4|   5977|   2302|   3675|         884|             471|         73|     12|
|100654|2020|       1|     4|   5977|   2302|   3675|           3|               1|          1|     11|
|100654|2020|       1|     4|   5977|   2302|   3675|        5090|             791|        144|      3|
|100654|2020|       1|     4|   5977|   2302|   3675|        5093|             792|        145|      2|
|100654|2020|       1|     4|   5977|   2302|   3675|        5977|            1263|        218|      1|
+------+----+--------+------+-------+-------+-------+------------+----------------+-----------+-------+
only showing top 5 rows

1847918
```

```
corr_df = combined_2.select("EFTOTLT", "ONLINE_TOTAL")
corr_df = corr_df.dropna()
corr_df.show(5)
```

```
+-------+------------+
|EFTOTLT|ONLINE_TOTAL|
+-------+------------+
|   5977|         884|
```

Terminal

+ Code + Text ▷ Run all

```
corr_df = combined_2.select("EFTOTLT", "ONLINE_TOTAL")
corr_df = corr_df.dropna()
corr_df.show(5)
```

```
+-------+------------+
|EFTOTLT|ONLINE_TOTAL|
+-------+------------+
|   5977|         884|
|   5977|           3|
|   5977|        5090|
|   5977|        5093|
|   5977|        5977|
+-------+------------+
only showing top 5 rows
```

```
corr_value = corr_df.stat.corr("EFTOTLT", "ONLINE_TOTAL")
corr_value
```

```
0.5606536614921085
```

```
import pandas as pd
import matplotlib.pyplot as plt

pdf = corr_df.toPandas()
```

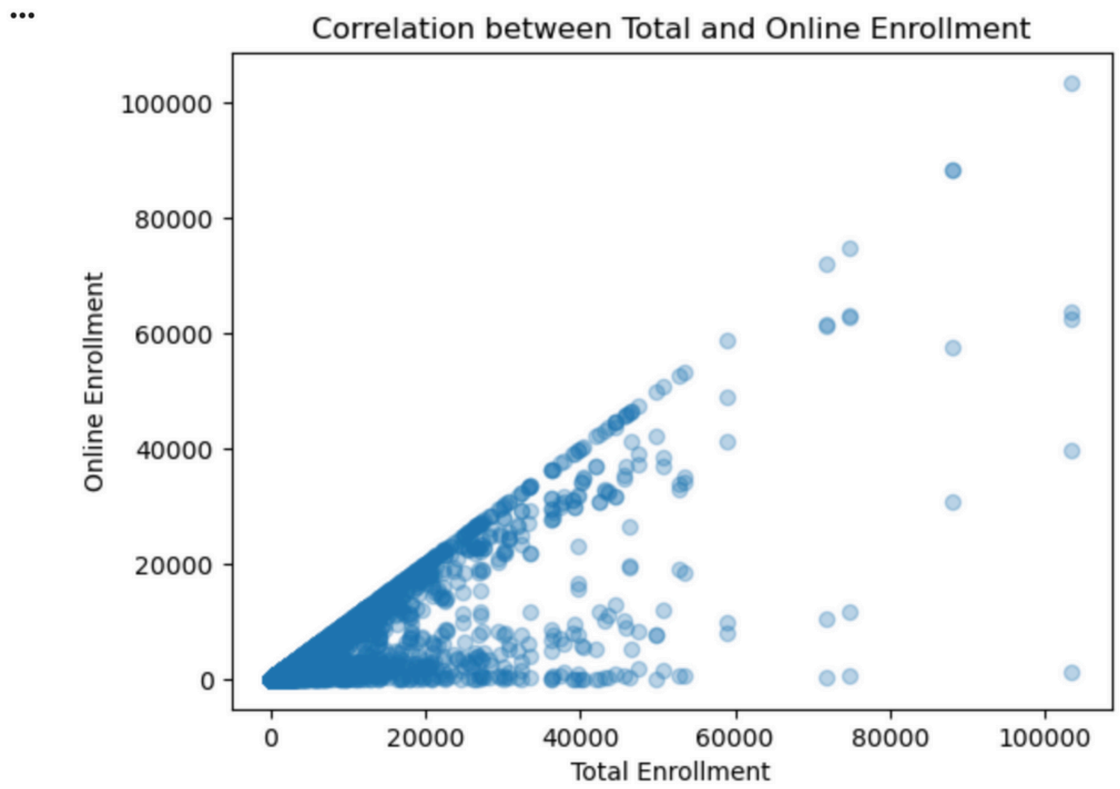
Terminal

OUTPUT FOR BUSINESS QUESTION 3

A positive but moderate correlation exists.

Larger institutions generally enroll more online students, but online learning is also widely adopted among mid-size and smaller institutions—indicating broad flexibility in online education models.

```
plt.title("Correlation between Total and Online Enrollment")
plt.show()
```



SUMMARY OF FINDINGS

- Online enrollment has increased sharply from 2020–2023.
- Adoption patterns differ significantly across institutional sectors.
- There is a meaningful relationship between institutional size and online engagement, but online learning remains widely distributed across all institution types.
- Multi-year data revealed consistent schema variations requiring normalization.

CHALLENGES AND LEARNINGS

CHALLENGES:

- Schema mismatches across years
- Missing or renamed columns
- Hive warehouse location conflicts
- Data ingestion and storage path issues

LEARNINGS:

- Effective use of Spark for distributed joins & aggregations
- Importance of schema normalization for multi-year analytics
- Workflow design for end-to-end big data architecture
- Visualization to interpret large-scale education patterns

THANK YOU

QUESTIONS ?