# FINAL EXAM

**MADSC102 – Unlocking the Power of Big Data**

Hachem Sfar

Term 1 (2025/2026)

**Exam Dates: 8, 9, 10 December 2025**

MUC07313 → Monday 08 December

MUC07322→ Tuesday 09 December

MUC07319→ Wednesday 10 December

---

### 1. Overview

This final exam evaluates your ability to design and implement complete data analytics mini-project using one modern data platform of your choice:

1. Microsoft Fabric
2. SnowflakeDB
3. Databricks

You will build an end-to-end workflow from data ingestion to analytics, dashboarding, and optional automation.

This is an **individual exam**. Each student must complete and present their own original work.

---

### 2. Project Requirements

You must choose **one** of the following technologies for your entire project:

- Microsoft Fabric (using Lakehouse, PySpark, notebook, SQL, and Data Factory pipelines....)
- SnowflakeDB (using Tables, SQL Worksheets, Snowpark, notebook, Python, and Streamlit...)
- Databricks (using Delta tables, PySpark, notebook, SQL, and Databricks dashboards...)

Your chosen platform must be used for both ingestion and analysis.

---

### 3. Data Source Requirements

You must choose **one** of the following data ingestion methods:

**Option A: API**

Ingest data from a public API. Your code must read the API, load the response, clean it, and save it into one or more tables.

**Option B: Kaggle CSV File**

Download a dataset from Kaggle and upload it into your platform.
You must perform structured ingestion, cleaning, and table creation.

- You are not allowed to reuse the datasets from any of the previous labs.

---

## 4. Core Technical Requirements

### 4.1 Data Ingestion

Depending on your chosen data source:

- API: Build a script/notebook that retrieves data from one or more endpoints
- CSV: Upload your dataset into your chosen platform using its ingestion tools.

You must load the cleaned data into at least **one** table.

### 4.2 Data Cleaning,Preparation and analysis

Use PySpark (Fabric or Databricks) or Snowpark Python (Snowflake) to:

- Clean the data
- Prepare and create the final table(s)

### 4.3 Data Storage

Your pipeline must store the cleaned dataset in:

- Fabric: Delta tables in Lakehouse or warehouse
- Snowflake: Database tables
- Databricks: Delta tables in your catalog

### 4.4 SQL Analysis

You must create at least **2 SQL analytical queries**

### 4.5 Notebook Analysis

Use a notebook with PySpark or Snowpark to:

• Load your table
• Explore the dataset

### 4.6 Dashboard

Create one dashboard inside your chosen environment:

• Fabric: Power BI dashboard linked to Lakehouse
• Snowflake: Streamlit dashboard
• Databricks: SQL Dashboard

Your dashboard must include:

• At least two charts
• Filters
• Clear titles

---

## 5. Bonus Components (Optional)

Students may earn bonus marks by implementing one or more of the following:

1. **Automated Data Pipeline**
   • Fabric Data Factory pipeline
   • Snowflake Tasks / Snowpipe
   • Databricks Jobs or Workflows

relevant when using API data.

2. **Machine Learning Component**
   A small ML model such as:
   • Regression
   • Classification
   • Forecasting
   Use PySpark MLlib, Snowpark ML, or Fabric ML options.

---

## 6. Final Deliverables

Upload all your work in github and put the link in the moodle submission

---

## 7. In-Class Presentation

Each student will present:

• A 5-minute live demonstration of their project
• A walkthrough of ingestion, cleaning, SQL, and dashboard

**Suggested Data Sources**

Public APIs

Geolocation & Environment

• OpenWeatherMap – Weather data (current, forecast, historical)

https://openweathermap.org/api

• AirVisual API – Global air quality and pollution data

https://www.iqair.com/world-air-quality

Demographics & Government

• REST Countries – Country info: population, currencies, flags

https://restcountries.com/

• US Census Bureau – U.S. population and economic data

https://www.census.gov/data/developers/data-sets.html

• World Bank API – Global development stats and economic indicators

https://datahelpdesk.worldbank.org/knowledgebase/topics/125589

Energy

• EIA Open Data – U.S. energy production and consumption

https://www.eia.gov/opendata/

• Open Power System Data – European electricity data (load, generation, price)

https://open-power-system-data.org/

• OpenEI Utility Rates – Global utility rates and energy costs

https://openei.org/services/doc/rest/util_rates/

Health

• OpenFDA – Drug, device, food safety data

https://open.fda.gov/apis/

• CDC WONDER – U.S. public health statistics

https://wonder.cdc.gov/wonder/help/wonder-api.html

• WHO Athena API – Global health indicators

https://www.who.int/data/gho/info/athena-api

• ClinicalTrials.gov – Active and past clinical study info

https://clinicaltrials.gov/api/gui

Stock Market & Finance

• Alpha Vantage – Stocks, forex, crypto, technical indicators

https://www.alphavantage.co/

• Twelve Data – Real-time and historical stock/forex data

https://twelvedata.com/

• Finnhub – Real-time quotes, news, fundamentals

https://finnhub.io/

• Polygon.io – Stocks, options, forex, crypto data

https://polygon.io/

• MarketStack – Stock market data for 70+ exchanges

https://marketstack.com/

• EOD Historical Data – End-of-day prices and financials

https://eodhistoricaldata.com/

Currency & Forex

• ExchangeRate-API – Live and historical exchange rates

https://www.exchangerate-api.com/

• Frankfurter API – ECB-based currency rates

https://www.frankfurter.app/

• CurrencyLayer – Forex data for 160+ currencies

https://currencylayer.com/

Cryptocurrency

• CoinGecko API – Crypto data: prices, market cap, trends

https://www.coingecko.com/en/api

Science & Space

• NASA APIs – Imagery, astronomy, satellites, and more

https://api.nasa.gov/

• Open Notify – ISS current location and crew

http://open-notify.org/Open-Notify-API/

Others

• SEC EDGAR – U.S. company filings (10-K, 10-Q, etc.)

https://www.sec.gov/edgar/sec-api-documentation

• Big List of No-Auth APIs – Huge curated list of open APIs

https://mixedanalytics.com/blog/list-actually-free-open-no-auth-needed-apis/

More APIs: Public APIs GitHub List

Open Datasets

• Kaggle Datasets

• Data.gov

• World Bank Open Data

• UCI Machine Learning Repository

• Google Dataset Search

**Grading Criteria**

| Criteria | Exceptional (90–100) | Very Good (80–89) | Good (70–79) | Pass (60–69) | Fail (Below 60) |
|---|---|---|---|---|---|
| Understanding of Big Data Concepts (20%) | Demonstrates excellent understanding of big data workflows, platforms, and ingestion. | Clear understanding with minor gaps. | Adequate understanding with some inconsistencies. | Limited understanding; partially correct explanations. | Major misunderstandings or unclear work. |
| Technical Implementation: Ingestion, Tables, Cleaning (30%) | Fully functional ingestion, clean tables, correct data types, strong pipeline logic. | Mostly correct implementation; few issues. | Basic implementation with some errors. | Minimal correctness; weak data preparation. | Incomplete, incorrect, or missing ingestion. |
| SQL and Notebook Analytics (25%) | High-quality, accurate SQL + notebook work showing depth and insights. | Strong queries with small issues. | Correct queries but lacking depth. | Basic queries with weak results. | Incorrect or missing SQL and analysis. |
| Dashboard Quality (15%) | Professional, clear, insightful dashboard; strong charts and filters. | Good structure, readable visuals. | Functional but unpolished. | Barely functional dashboard. | Missing or unusable dashboard. |
| Presentation and Communication (10%) | Clear, confident, well-organized | Good presentation with minor issues. | Understandable but not polished. | Poorly structured or unclear. | Not presented or fails to explain project. |

| | | | | | |
|---|---|---|---|---|---|
| | demonstration; answers questions well. | | | | |
| Bonus (Optional) | Automation, ML, or added features implemented well. | Partial bonus implementation. | Minor improvements attempted. | Weak or incorrect attempt. | No bonus work attempted. |