Ministry of Higher Education and Scientific Research
University Center of Mila
Institute of Mathematics and Computer Science
Department of Computer Science
Master 2 I2A – Big Data
2025/2026

Practical Works 5 - PySpark Applied Practice

Dataset A: Students Performance

```
Name,Department,GPA,Credits
Amine,CS,15.5,42
Sara,Math,12.3,38
Rachid,CS,17.1,46
Lina,Bio,13.8,40
Malek,Math,11.5,32
```

Dataset B: Products Sales

```
Product,Category,Price,Quantity,City
Keyboard,Electronics,25,10,Oran
Mouse,Electronics,15,20,Algiers
Orange,Fruit,1.4,30,Oran
TV,Electronics,480,3,Setif
Apple,Fruit,2.1,25,Algiers
Camera,Electronics,350,2,Constantine
```

Dataset C: Flight Data

```
FlightID,Airline,Price,Passengers,City
F01,AirAlgerie,150,120,Oran
F02,Turkish,650,180,Algiers
F03,AirAlgerie,200,140,Constantine
F04,Qatar,700,160,Algiers
F05,Emirates,900,170,Oran
```

Exercise 1: Spark Initialization & Inspection

Using any dataset:

1. Create a SparkSession named "TP5_Practice".
2. Print the Spark version, master, and app name.
3. Explain (in a markdown cell or comments):
   - What is the role of the driver?
   - When would executors be used?

Exercise 2: Transformations, Actions & DAG Execution

Using Dataset A (Students Performance):

1. Load the CSV with schema inference.
2. Show the inferred schema.
3. Compute:
   - Students with GPA ≥ 14
   - Only keep Name + Department
4. Trigger execution and show the result.

Questions to answer (in comments):

- List all transformations you used.
- Identify the action that executed the DAG.

Exercise 3: Schema Manipulation & Column Operations

Using Dataset B (Products Sales):

1. Load the CSV with correct types (Price as float, Quantity as integer).

2. Add a new column:
   Revenue = Price × Quantity.
3. Cast Revenue to integer.
4. Compute for each category:
   - total quantity sold
   - average revenue
5. Print the schema before and after casting.

Questions:

- How does schema enforcement help Spark optimization?
- Which operations caused shuffling?

Exercise 4: Caching & Performance Optimization

Using Dataset C (Flight Data):

1. Load the data.
2. Add column TotalRevenue = Price × Passengers.
3. Cache the DataFrame.
4. Show the DataFrame twice.
5. Measure/observe differences in execution time (Spark UI if available).

Questions:

- Why does caching improve the second execution?
- What happens if you remove the cache?

Exercise 5: “Algerian Cities Analytics”

Using a merged DataFrame that joins Dataset B and Dataset C on City:

1. Load both datasets with proper types.
2. Perform a join on City.
3. Compute:
   - Total product revenue per city
   - Total flight revenue per city
   - Combined revenue per city
4. Sort cities by highest combined revenue.
5. Extract the top 1.
6. Cache any DataFrame that you reuse multiple times.

Questions:

- Which transformations were narrow?
- Which were wide (explain the shuffle)?
- Which action(s) executed your plan?
- How would Spark recover from a partition failure?