

## TD 2 – MapReduce Programming and Analysis

### Objectives

- Understand the MapReduce paradigm through practical data-processing exercises.

### Exercise 1 – Word Frequency Counter

**Goal:** Count the occurrences of each word in a given text file.

| Example Input:                                                                   | Expected Output:                                                                       |
|----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| Hadoop is good for Big Data<br>Big Data means large data sets                    | Hadoop 1<br>is 1<br>good 1<br>for 1<br>Big 2<br>Data 2<br>means 1<br>large 1<br>sets 1 |
| Mapper logic:<br>For each line in input:<br>Split into words      Emit (word, 1) |                                                                                        |
| Reducer logic:<br>For each word:<br>Sum all values      Emit (word, total_count) |                                                                                        |

### Exercise 2 – Comparing Workload Growth

**Goal:** Compute the average temperature per weather station.

| Example Input:                                                                             | Expected Output:  |
|--------------------------------------------------------------------------------------------|-------------------|
| ST01,2025-01-01,10<br>ST01,2025-01-02,12<br>ST02,2025-01-01, 8                             | ST01 11<br>ST02 8 |
| Mapper logic:<br>Emit(station, temperature)                                                |                   |
| Reducer logic:<br>Compute average(temperature) for each station<br>Emit(station, avg_temp) |                   |

### Exercise 3– Total Sales Per Product

**Goal:** Count the total revenue for each product

| Example Input:                                                                               | Expected Output:        |
|----------------------------------------------------------------------------------------------|-------------------------|
| 2025-01-01,Milk,20,100<br>2025-01-02,Milk,10,100<br>2025-01-01,Bread,50,20                   | Milk 3000<br>Bread 1000 |
| Mapper logic:<br>product = fields[1]<br>revenue = quantity * price<br>Emit(product, revenue) |                         |
| Reducer logic:<br>Sum all revenues for each product<br>Emit(product, total_revenue)          |                         |

### Exercise 4 – Average Movie Rating

**Goal:** Compute the average rating per movie

| Example Input:                                                                   | Expected Output:   |
|----------------------------------------------------------------------------------|--------------------|
| M001,U01,4<br>M001,U02,5<br>M002,U01,3                                           | M001 4.5<br>M002 3 |
| Mapper logic:<br>Emit(movie_id, rating)                                          |                    |
| Reducer logic:<br>Compute average rating per movie<br>Emit(movie_id, avg_rating) |                    |

### Exercise 5 – Word Count by First Letter

**Goal:** Count how many words start with each alphabet letter

| Example Input:                                                            | Expected Output: |
|---------------------------------------------------------------------------|------------------|
| Hadoop is Highly Helpful                                                  | H 3<br>I 1       |
| Mapper logic:<br>Emit(first_letter, 1)                                    |                  |
| Reducer logic:<br>Sum counts by first_letter<br>Emit(letter, total_count) |                  |