Indian Institute of Technology Jodhpur

Fundamentals of Distributed Systems

Assignment – 2

Total Marks:	20
Submission Deadline:	27 July 2025

Datasets

- **Cruise data:** Cruise CSV file (click here to download)
- **Customer churn data:** Customer Churn CSV file (click here to download)
- **E-commerce customer data:** E-commerce Customer CSV file (click here to download)

Instructions

- Implement all MapReduce jobs using the `mrjob` library and `Hadoop` in Google Colab.
- At the top of your notebook, install dependencies and setup hadoop.
- Load each CSV directly from the URLs above using **wget** or **curl** command into the Google Colab.
- For each question:
  1. Write mapper, reducer (and combiners or multi-step definitions) as `mrjob` classes.
  2. Include a brief docstring explaining your design in Google Colab using markdown feature for each question and cell of Colab.
  3. Demonstrate correctness on a small inline example.
- Name your notebook `Assignment2-(Roll No of Yours)-(Name of yours).ipynb` and submit a link to GitHub or Colab and also submit the Jupyter notebook file in LMS.
- At the end, include a cell that runs all jobs on full datasets and shows final outputs.

Questions

1. **Cruiseline Aggregations (5 marks)**
   Using `cruise.csv`, implement an `mrjob` class that computes, for each `Cruise_line`:

   (a) Total number of ships.

   (b) Average `Tonnage` (to two decimals).

   (c) Maximum `crew` size.

   (Optional) Use a Combiner for partial aggregation.

2. **Company Churn Rate (5 marks)**
   From `customer_churn.csv`, create a `MultiStepJob`:

   Step 1: Mapper emits `(Company, TOTAL)` and `(Company, CHURNED)` where `Churn`==1.

   Step 2: Reducer computes churn rate = $\frac{\text{CHURNED}}{\text{TOTAL}}$, outputting four-decimal floats.

   Use a small `VIP_companies.txt` in the distributed cache to restrict to listed companies. Provide a sample file with at least three names.

3. **State-wise Spending (5 marks)**
   From `e-com_customer.csv`, extract the two-letter state code from the `Address` field. Then:

   - Mapper parses the state.
   - Reducer sums `Yearly Amount Spent` per state.
   - Output the top 5 states by total spending.

4. **Two-step Ship Filter & Median Length (5 marks)**
   On `cruise.csv`, implement a two-step `mrjob` pipeline:

Step 1: Filter ships with `passenger_density` > 35.0; emit $\langle \texttt{Cruise\_line}, \texttt{length} \rangle$.

Step 2: Compute the median of the lengths per `Cruise_line`, handling even/odd counts correctly.

Use the `steps()` API and output medians to two decimals.

Submission

- Submit `Assignment2.ipynb` with all code, inline tests, proper brief markup comments for each question and final outputs.
- Ensure each question's results are clearly labeled.
- Provide a GitHub or Colab link for assessment.