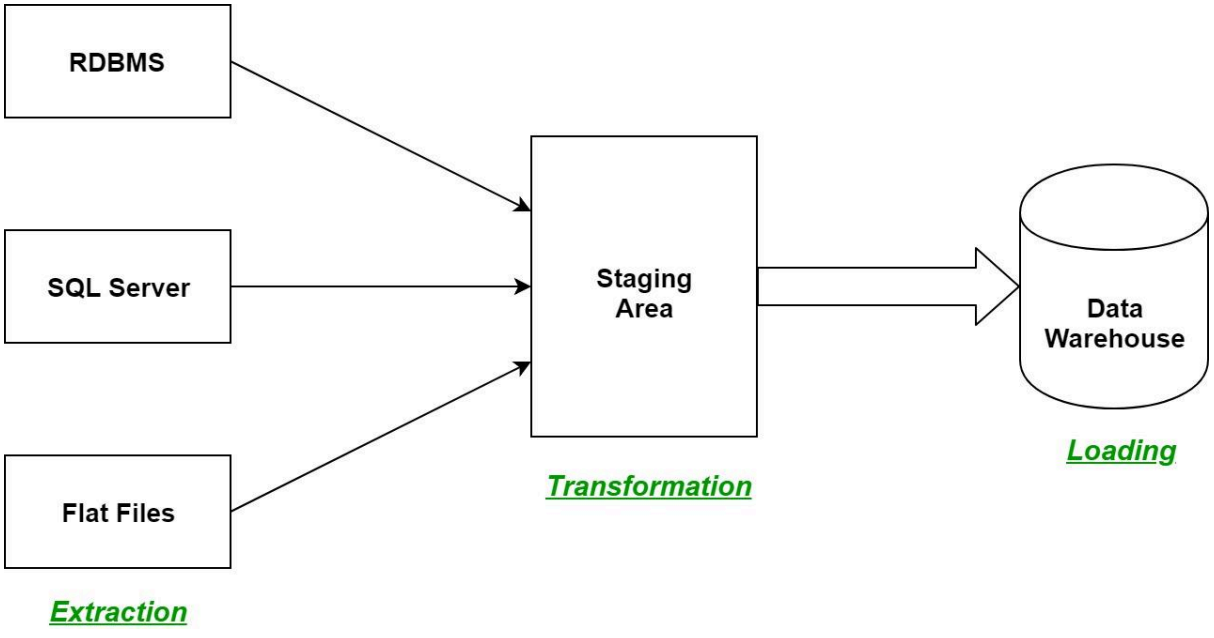
ETL Pipeline Using Multiple Data

Course: Big Data Analytics (BDA)

(CT-592)

Program: Masters in Data Science



Submitted by:

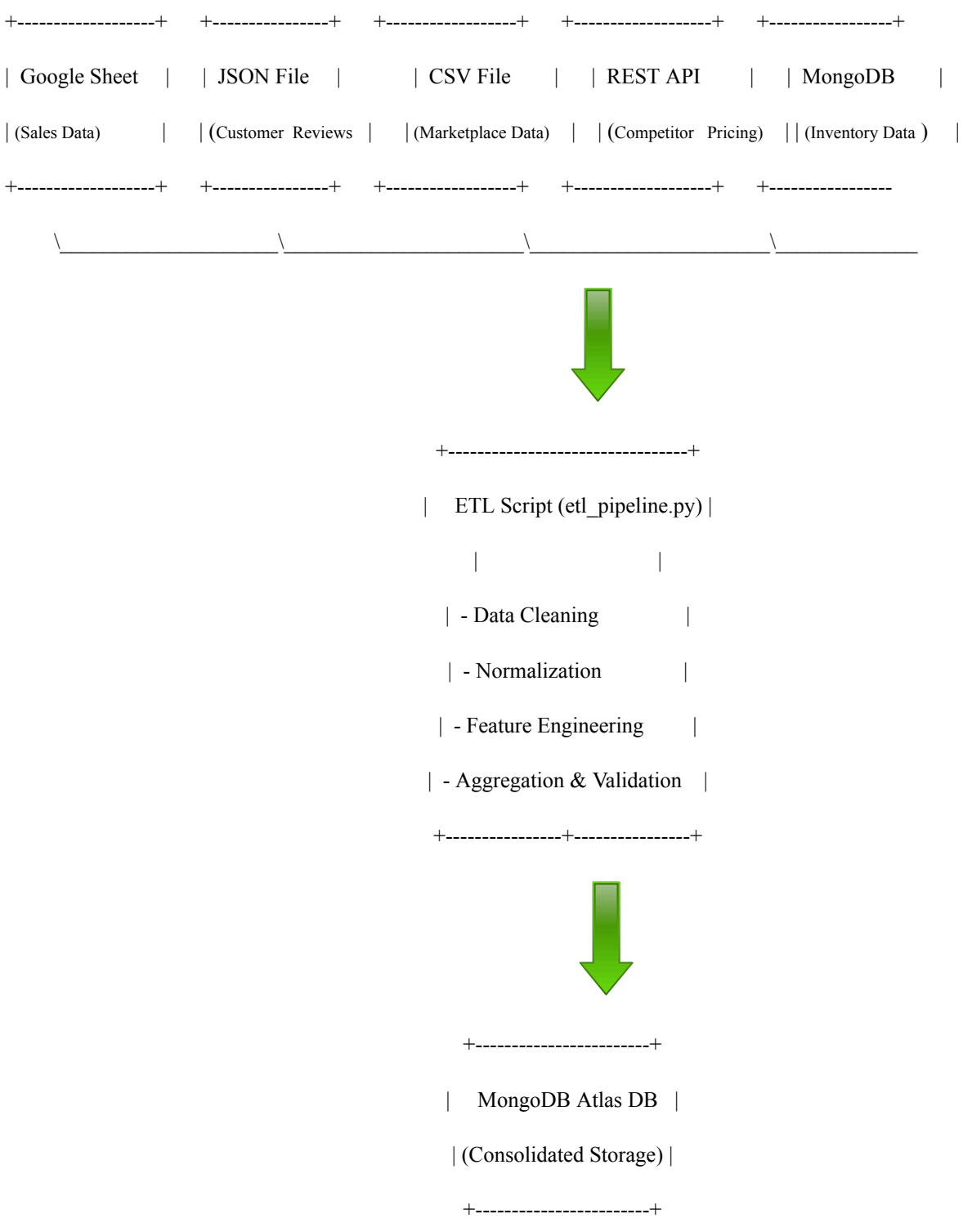
Name: Mehwish
Roll no: DS-W-013/2024-25

Submitted to Dr. Muhammad Umer Farooq

1. Pipeline Design Overview

This ETL pipeline integrates and processes mobile phone sales data from five diverse sources. The pipeline follows a modular design to ensure maintainability, scalability, and automation:

Pipeline Flow Diagram



2. Technology & Tool Justification

✅ Python

- Why: Easy integration with diverse data formats, strong data manipulation libraries (Pandas), and a rich ecosystem for scheduling, testing, and automation.

✅ Pandas

- Why: Fast, flexible, and expressive tool for data analysis and manipulation.

✅ MongoDB Atlas (Cloud DB)

- Why: Free-tier cloud-hosted NoSQL database with flexible schema support and native JSON storage that suits semi-structured ETL data.

✅ Schedule (Python Library)

- Why: Lightweight job scheduler for setting up cron-like automated task execution in a simple Python script.

✅ GitHub Actions (CI/CD)

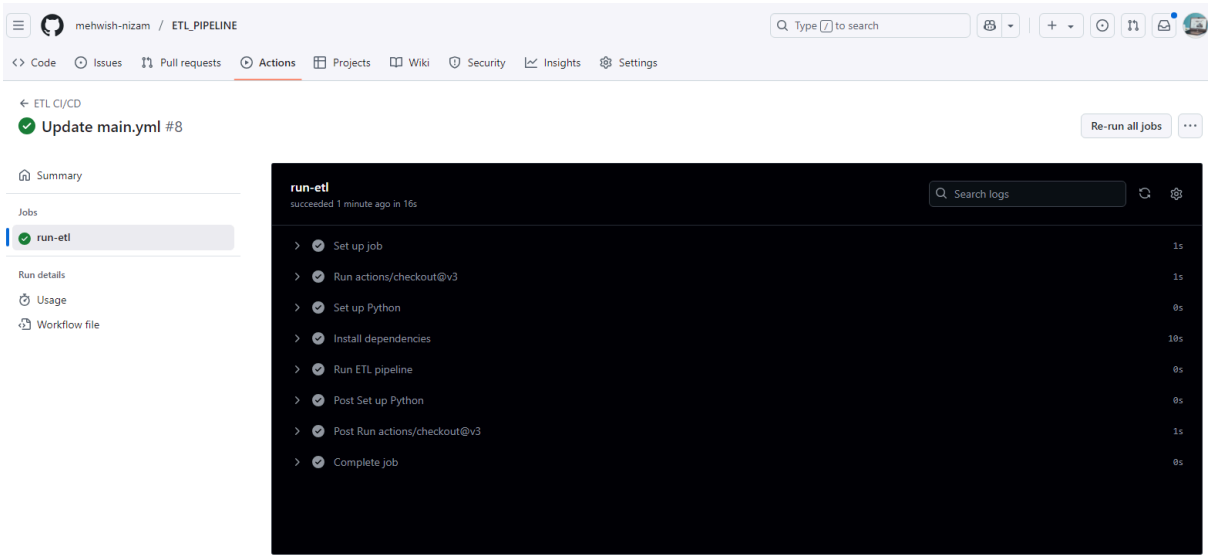
- Why: Seamless integration with GitHub repo, automates testing, linting, and validation of code for every push or pull request.

3. CI/CD Pipeline Overview

🔧 GitHub Actions Workflow (`main.yml`)

- Trigger: On push or pull request to `main`
- Steps:
    1. Setup Python
    2. Install dependencies via `requirements.txt`
    3. Lint Python code using `flake8`
    4. Run unit tests (if any)
    5. Check successful data transformation (e.g., shape, nulls, schema)

Screenshot



4. CI/CD Value Justification

✅ Reduces Manual Errors

- Automates repetitive tasks like validation, formatting, and testing before database load.
- Ensures data integrity is preserved by testing schema compatibility before upload.

🔁 Facilitates Rapid Feedback Loops

- Developers receive instant feedback upon commit or PR.
- Bugs or data issues can be addressed immediately, improving turnaround time.

✅ Improves Data Integrity through Automated Testing

- Linting and unit tests validate ETL steps.
- Prevents invalid or malformed data from corrupting the database.

🚀 Accelerates Development & Deployment

- One-click commits trigger full validation + load to MongoDB.
- Future integration with cron/scheduler can enable an end-to-end production pipeline.