ETL Pipeline for Sports Data – Detailed Project Report

Project Information

Author: Chander Parkash Roll No: DS-032/2024-25 Project Title: ETL Pipeline for Sports Data Repository: https://github.com/ChanderParkash179/ETL_Pipeline_ChanderParkash_DS-032-2024-25

Overview

This project implements a robust ETL (Extract, Transform, Load) pipeline specifically designed to process multi-format sports data. The pipeline reads data from CSV, JSON, Excel, and MongoDB, transforms and cleans the data, and stores it back into MongoDB as well as exports it to CSV for analysis and sharing.

Key Features

- Multi-source Extraction: CSV, JSON, Excel, MongoDB - Transformation: Deduplication and Missing value handling - Loading: Cleaned data to MongoDB - Export: Final data to CSV

Technical Stack

Language: Python 3 Database: MongoDB Visualization: matplotlib, seaborn Libraries: pandas, pymongo, requests, faker

Installation Instructions

git clone https://github.com/ChanderParkash179/ETL_Pipeline_ChanderParkash_DS-032-2024-25 cd etl-pipeline pip install -r requirements.txt

Pipeline Usage

Run: python etl_pipeline.ipynb Data Flow: 1. Extract (CSV, JSON, Excel, MongoDB) 2. Transform (clean, deduplicate) 3. Load to MongoDB 4. Export to CSV

Sample Output

› Extracting CSV from /content/sports_data.csv... › Extracting JSON from /content/sports_data.json... › Extracting Excel from /content/sports_data.xlsx... › Extracting data from MongoDB... › Transforming data... › Loading data into MongoDB collection 'load_sports_data'... ■ Inserted 1287 records ■ Total records were 1700

Data Schema

Name – Name of athlete Nationality – Country represented Sport – Type of sport Team – Team name or affiliation

Configuration File

{ "mongo_uri": "mongodb+srv://root:root@mid-term-cluster.mxl0ovi.mongodb.net/" }

Output Details

MongoDB Collection: sports_data.load_sports_data CSV File: /output/final_cleaned_data.csv

Performance

Efficient duplicate removal Processed: 1700 records Cleaned: 1287 records

Automation: Scheduler

Runs every 24 hours via scheduler.py for automated processing.

CI/CD Pipeline – GitHub Actions

Trigger: On push or PR to main branch Steps: 1. Checkout Code 2. Setup Python 3.9 3. Create Required Dirs 4. Set Mongo URI from Secrets 5. Clean notebook code & run ETL 6. Upload final CSV as artifact

Conclusion

This ETL pipeline offers a scalable, automated, and reliable method to process and manage sports data. With data cleaning, MongoDB integration, and CI/CD, it's well-suited for analytics and automation tasks.