PyQL

A Custom SQL–ish Data Processing Engine

Prepared By Jhene Ekuwem

1

Introduction

This project will follow the **SQL Option** for a **one person group.**

The primary objective of this project is to develop a compact and robust data processing library in Python mirroring the functionality of the pandas library. This consists of a proper csv file parser, DataFrame object class for storing data, and accessor and mutator functions for data operations and manipulation.

The culmination of this project will be a simple web application that will allow a user to seamlessly query data using an intuitive UI.

Implementation

This project will consist of the following design and implementation:

1. **CSV Parser:** Will open and sequentially read a .csv file, establishing column names and column data.
2. **DataFrame Object Class:** Will store all data in a Python dictionary where keys are column names as type strings and values are Python lists consisting of column data. The object class will have the following functions to retrieve and alter data:
   a. **Selection/Projection:** Will use Python __getitem__ to select columns.
   b. **Filtration:** Will use Python __getitem__ to and boolean logic to filter data.
   c. **Aggregation/Group By:** Will use logical partitioning and arithmetic methods like max(), sum(), etc to perform aggregation.
   d. **Join:** Will use Python dict functionality to join DataFrames objects as anticipated.

The following documents my anticipated timeline for each deliverable:

Week	Dates	Deliverables	Status
1	9.22 - 9.28	Parser Implementation	Pending

2	9.29 - 10.5	Core DataFrame Class	Pending
3	10.6 - 10.12	Filtering Functionality	Pending
4	10.13 - 10.19	Midterm Progress Report	Pending
5	10.20 - 10.26	Aggregation Functionality	Pending
6	10.27 - 11.2	Join Functionality	Pending
7	11.3 - 11.9	Web App Development:	Pending
8	11.10 - 11.16	Web App Integration	Pending
9	11.17 - 11.23	Demo Preparation:	Pending

Labor Division

Due to this being a solitary project, I'll be completely response for the full stack development of the project, which includes the following:

- Design and implementation of the CSV parser and DataFrame object class.
- Implementation of all accessor and mutator functions.
- Development of the web application.
- Completion of reports and final demo.