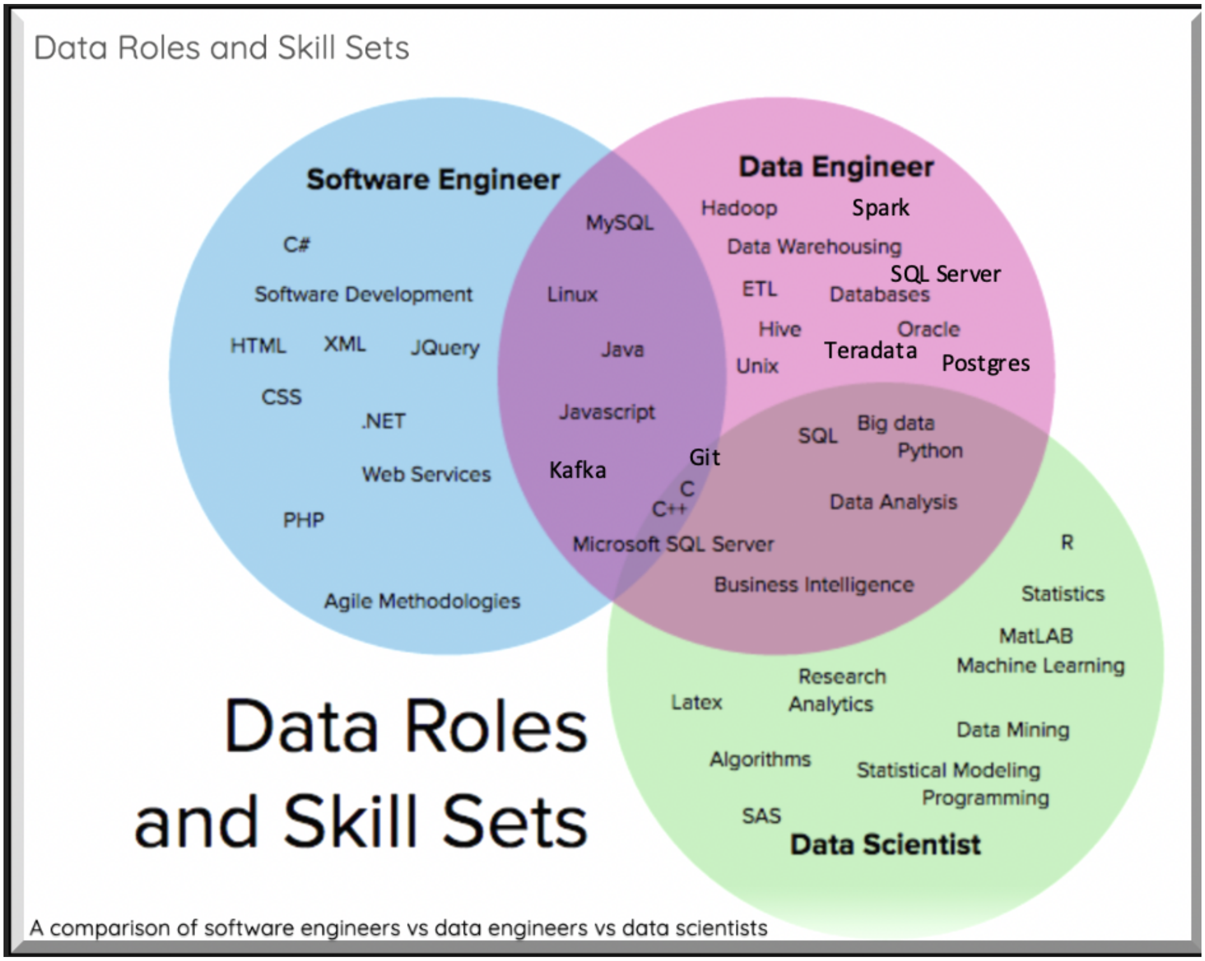
Homework1.md 5.48 KB

Homework Assignment

Problem 1: Flight Schedule



Background:

The role of data engineer is intended to support the work of our data scientists by helping them get the data they need in order to continue their research. As such, "practical" and "done" is more important than "perfect", but the validity of our models depends on obtaining good quality, timely correct data.

This assignment is intended to see how you would approach an every-day data engineering problem. We expect that it should take approximately three hours to complete.

Your work will be assessed on a number of criteria:

- Does it work as required? We will follow any instructions you have provided and attempt to re-run your project.
- We will assess code-quality - for example correct idiomatic use of Python, readability and style.

Please submit your work back as a zip file. Include any file or component which is needed to re-run the assignment such as scripts and instructions. Please do not include any database internal files.

Assignment:

TASK 1: Write an SQL query to report all the available flights in the table below with their most recent status. The Sample data is provided. Assume that the data is already ingested in a database.

TASK 2: Similar to the above Task 1 but in Python -- Write python function with input parameter as file path, ingest the provided dummy data and report all the available flights with their most recent status. The Sample data is provided.

Table: Flight Leg

Column Name	**Type**
flightkey	varchar
flightnum	varchar
flight_dt	date
orig_arpt	varchar
dest_arpt	varchar
flightstatus	varchar
lastupdt	datetime

flightkey is the primary key column for this table. Each row of this table contains an flight information. \

Example 1:

Input: Flight Leg table:

flightkey	**flightnum**	**flight_dt**	**orig_arpt**	**dest_arpt**	**flightstatus**	**lastupdt**
DL4346615ATLLAX	3401	2021-01-01	ATL	LAX	boarding	2019-01-01T09:02:17
DL4346615ATLLAX	3401	2019-01-01	ATL	LAX	In	2019-01-01T09:20:17

Return the result table in any order.

The result format is in the following example.

Output:

flightkey	**flightnum**	**flight_dt**	**orig_arpt**	**dest_arpt**	**flightstatus**	**lastupdt**
DL4346615ATLLAX	3401	2019-01-01	ATL	LAX	In	2019-01-01T09:20:17

Explanation: flightkey is repeated two times. The most recent status is 'In'.

Guidelines

- This is meant to be an assignment that you spend approximately three hours of dedicated, focused work. Do not feel like you need to overengineer the solution with dozens of hours to impress us. Be biased toward quality over quantity.

- Think of this like an open source project. Create a repo on Github, use git for source control, and use README.md to document what you built for the newcomer to your project.

- Our team builds, alongside our customers and partners, systems engineered to run in production. Given this, please organize, design, test, deploy, and document your solution as if you were going to put into production. We completely understand this might mean you can't do as much in the time budget. Be biased for production-ready over features.

- Think out loud in your submission's documentation. Document tradeoffs, the rationale behind your technical choices, or things you would do or do differently if you were able to spend more time on the project or do it again.

- Our team meets our customers where they are in terms of software engineering platforms, frameworks, tools, and languages. This means you have wide latitude to make choices that express the best solution to the problem given your knowledge and favorite tools. Make sure to document how to get started with your solution in terms of setup.

Features:

1. Use SQL to complete the task
2. use python to complete the task

Data Sources

Flights data

Implementation:

- Preferred Technology:
  - Python
  - SQL
  - Java

However, you may use other tech if you are more comfortable with something else. You can use any additional technologies/frameworks/DBs/libraries you would like to.

How To submit your solution:

- Return your solution within 3 business days as a zip file, unless other directions provided.
- Feel free to ask questions at any time.

Task Summary

- Write a SQL query to dedup the flight leg data and return the most recent records per flightkey.
- Use Python, ingest the provided data, then dedup the flight leg data and return the most recent records per flightkey..
- Provided some instructions/commects which will allow us to repeat/replcate everything you have done.

Requirements

We store all our data in a postgres database.

- Please use a test-driven development approach.
- Please use Python as your main programming language. You may also use shell-scripts, SQL and other tools.
- We will want to re-run the script ourselves, so so please make sure that all aspects of this assignment is easily rerunnable. It should be possible for us to re-create your schema and run your import script.