Data Engineer Candidate Assessment

2022

# Introduction

The City of Toronto produces monthly reports of its employee’s expense card transactions. The transactions can be downloaded from the City’s [Open Data Portal](https://open.toronto.ca/dataset/pcard-expenditures/).

# Objectives

The objectives of this exercise are to assess your skills, knowledge and creativity in the following areas:

* Perform a task that is consistent with the duties of the position for which you are being considered.
* Design and implement a prototype of an end-to-end self-service analytics solution.
* Create a data model to represent financial information.
* Build a data warehouse that can feed reports, including visualization of relevant information.
* Apply DevOps/DataOps approaches and techniques.

Specifically, the objective of the exercise is to build an end-to-end solution that imports, cleans and formats data from Excel workbooks, populates a database with it and, ultimately, produce a dashboard.

# Procedures

To achieve this, please do the following:

* Implement the steps outlined in the sections below. The implementation details, design choices are up to you. The objectives of the dashboard are purposely simple, so that you have a chance to think about design.
* We use the Microsoft Data Platform Stack, which you can use. Alternatively, feel free to implement your solution using PostgreSQL.
* Please consider DevOps and DataOps practices on packaging and presenting the solution.
* There are sections marked as *(Optional).* Please note that these sections are related with Machine Learning Engineering practices and can be used to demonstrate your knowledge in the area.
* This test is designed to be completed under a “reasonable” time and will be evaluated with this in mind. You might want to be strategic about your implementation so you demonstrate your capabilities as much as possible.

# Implementation Steps

## Obtain and Clean Data

* Please download the data set from <https://open.toronto.ca/dataset/pcard-expenditures/>
* Write a procedure to clean the data: remove unnecessary records, recode/relabel values as necessary. You can do so in Python and/or SQL.

## Data Model

Create a data model:

* Infer entities from the records.
* Create tables that map entities, their attributes and the relationship among entities.
* Assign keys as necessary, normalizing the data while balancing redundancy and performance.
* Briefly explain the rationale behind the choices in your data model.

## Data Pipeline and Warehouse

* Design and describe the operation of a in import procedure such that the data set is updated on a regular basis and consistent with the steps taken above.
* Build a small data warehouse that contains weekly expense reports by Division. The data structure should be based on well recognized patterns. Consider preserving data history.
* Include as many variables/features as you may need for the following section.
* The ETL/ELT process should be granular and ready to be automated and orchestrated
* Consider keeping track of relevant events that occur before, during, and after the execution of the ETL/ELT process.

## *(Optional)* Model Framework

* Build a simple model that predicts next week’s total expenses based on this week’s expenses. The model must depend on at least one hyper-parameter which would need to be tuned.
* Store performance metrics as well as hyper-parameter configurations in a data set. You could write your own experiment manager or use a readily available library such as MLFlow or Sacred.

## Visualization

* Create a simple Semantic Model to support Self-Service Analytics and the dashboard of the next point
  + It should be easy and user friendly to interact with
  + It should allow for high-level and low-level data details
* Create an Interactive Dashboard to tell a Data Visual Story related with the proposed data set:
  + This visual story should give valuable Insights that would be difficult to get otherwise. Please create any story, insights or/and measures of your choice.
  + *(Optional)* Visualize Model predictions of next week’s expenditures, and any features that are important for prediction.
* Build a simple dashboard that displays the weekly expenditures by division.

# Guidelines for your submission

* Please provide your submission in a zip archive, including all code necessary to run the solution. Additionally, please feel free to use could-based storage platform, GitHub or similar. The solution should relatively easy to deploy for demo purposes. You might also want to consider including the source files.
* Include a readme.md or readme.txt file that walks us through installing/deploying your solution, as well as any instructions to operate it. You can assume that we have access to a DBMS with admin privileges.
* Please feel free to include a write-up or solution design document with considerations or any highlight you might want to include.
* As well, include a requirements file, if you decide to include code written in Python.

We appreciate your interest in working with us and look forward to speaking to you about your skills, this opportunity and, of course, your analysis.