yipitDATA

SpendHound Data Engineer/Data Integration Engineer Take-Home Exercise

Overview

At YipitData, we work with many different types of data. Recently YipitData has been looking into gathering data about Oscar-nominated movies from 1927 - 2014. Below is an API endpoint URL that contains movies and movie details, which we would like you to scrape for this assignment. These data are messy and will require some cleaning; you'll also have to use the original endpoint to find others that gather more details about each film. To build a web scraping process and an ETL pipeline for this data, we will need some helper toolkits to help clean up the data. Your task is to create some helper utility functions or a library to help extract, transform, and analyze the data.

The toolkit or library should contain functions to help with the following:

1. Scraping the data from an endpoint http://oscars.yipitdata.com/ to gather information about different movies.
    a. The schema should at the minimum contain the following: film, year, wikipedia_url, Oscar winner, original budget, and budget converted to USD
        i. The latter two can be found using the pages in the "Detail URL" field
2. At a minimum, create functions for the following:
    a. Cleaning up the "Budget" column to be an integer
        i. Any non-USD amount should be converted to USD
        ii. Any NaN should be converted to 0
        iii. Any budget that is a range (ie: $10,000,000 - $20,000,000), should be removed or $0
    b. [Bonus] Cleaning up the "Year" column
3. Using the tool that you have built, export a CSV file of the cleaned data

This assignment will require you to make certain assumptions and judgment calls. Please code your solution in Python. We expect you to work on this assignment for 2-3 hours. You will have 2 days upon receiving this prompt to submit your solution as a ZIP file, which should contain:

- All code to satisfy the requirements above
- A README.md file with instructions on how to run the code, as well as documentation on how to run the library, including installing any additional libraries and versions as needed. This should include any assumptions and/or examples of usage of the function. *(Max. length: 1 page)*

yipit DATA

- An EXPLAIN.md file that clearly explains your approach and assumptions made

How We Evaluate Submissions

We evaluate whether…

- The solution was submitted on time (within 2 days). Note: There are no bonus points for early submissions
- The solution is correct
- We can install and run the project by following the instructions from the README file
- The code is easy to understand
- The output file contains well-structured data (all necessary columns, reasonable column types, correct file format)
- The EXPLAIN.md file clearly explains your approach and assumptions made

Hints

- Python Requests is a great library for web scraping
- Inspect the raw data to identify and resolve data inconsistencies. Data can be messy to work with :)
- You can use tools like Pandas to help with manipulating the data
- Don't worry about optimizing for performance right now. If you encounter any edge cases, feel free to use your best judgment and document those decisions appropriately

Please submit the ZIP file to **the link that was provided in our previous email** (where we asked when you'd like to receive the exercise). **Please only click this link when you are ready to submit your work, as the link is very sensitive and can be corrupted easily**.