*Read Me*

**Preface:**

This script was developed in Python 3.7.6. It utilizes the **pandas**, **tkinter**, **os**, **sys**, **csv**, and **numPy**, packages to function. Typically, the only non-base packages that need to be installed are pandas and numPy. These can be downloaded and installed through pip.

**Functionality***:*

The basic purpose of this script is to sum the quantity of products sold through the chefpanzee portal using their original merchant pricing (instead of the marked up chefpanzee pricing) and then remove the appropriate commission rate (at the time development, 9.1%, for those who are not exempt). This produces the amount owed to each merchant during the specified pay period.

This script requires 3 input files/folders in order to produce a payout summary. They include:

* A product sales csv
  + This file is found on the Yelo analytics dashboard (*General|Analytics|Orders and Revenue*… scroll down to *Product Sales*). Clicking on the *Product Sales* carrot and selecting Filters will allow you to select the dates you wish to summarize. Once the relevant filters are set, clicking on the Product Sales carrot again and selecting Download|CSV will begin the download.
  + The important fields in this csv are “*product\_id*”, “*merchant\_name*”, and “*quantity*”. These columns can be moved, while other columns can be removed or added, as long as the header names remain the same.
* A merchant pricing folder
  + This merchant pricing folder must contain a csv for each merchant store:
    - <https://docs.google.com/spreadsheets/d/1nmoE8yKa5dXsWhCB2dqodL9KaEXk9GncZjqbMWVMp8g/edit?usp=sharing>
    - In order to download these files, click the *csv|export as csv files* in the top ribbon of the google sheet. “Running script” should appear. When the export is complete, a dialogue box should appear specifying the name of the saved folder. Select OK. The google sheets should be saved in your google drive. Right click this and select the download option to commence the download of a zip file. Once the download is complete, unzip this file in a designated merchant pricing folder.
  + The script will attempt to match the fields in the “*merchant\_name” column* from the product sales csv with the titles of the csv’s in this merchant pricing folder. If it cannot finds file titles in this folder that match a particular name from the “*merchant\_name*” column, it will ask you to select one.
  + The important fields in these csv’s are “*Product Id*”, “*Original Price*”, “*Category (Internal)*” and “*Cost Variation”*.
* An exempt merchant csv
  + This file is produced by you. Its purpose is to tell the script which merchants are exempt from paying a commission fee.
  + The csv contains a single column with the header “*merchant\_*name”. It should look like this:

A screenshot of a cell phone

Description automatically generated

* + The merchant\_name’s should match the exact formatting of the merchant names in the product sales csv.
  + You can add and remove names as needed. Make sure to have no empty rows between merchants.

After running the script, the output will look something like this:

A screenshot of a social media post

Description automatically generated

The first table includes a list of the merchant names and their associated payouts (sum of sales using their base pricing minus commission if it applies). Donations are listed separately for accounting purposes.

The second table lists products that are not summed because the base price may be higher than listed (due to combos, add ons, etc). These must be dealt with manually until unique product Ids for each variation are possible.

The third table lists Product Ids that are not found in the merchant csv the script is searching. The example above does not include any, but they will appear below the Product IDs Not Found header with the relevant merchant name if this occurs. In addition, if a Product ID experiences any other errors when the script attempts to sum, it will also appear here. If you search the merchant csv and find the appropriate product ID, another error must have occurred. Please reach out for help if this is the case.

**Before running for the first time:**

* Python 3 and the relevant packages must be installed.
* Make sure the relevant csv’s are available (as listed above).
* The python script includes a common “shebang” line in order to make it an executable. For this to function, complete the following:
  + Open Terminal
  + Change the terminal directory to the location of the script. For example, if the script is on your desktop, type:
    - cd /Users/Username/Desktop
      * Username should be replaced with whatever your personal username is
  + chmod +x Payouts\_Script.command

**INSTRUCTIONS***:*

1. Double click Payouts\_Script.command to begin the program
2. The first window that opens is meant for you to select the appropriate payouts csv you wish to summarize
3. The second window that opens is meant for you to select the folder that houses the merchant price csv’s
4. The third window that opens is meant for you to select the csv that houses the merchants that are exempt from paying commission.
5. The script should run, only to stop to ask for you to clarify merchant names that don’t match any particular merchant file.
6. The last window that opens is meant for you to choose folder where you’d like the summary to be saved.