

My Recipe Pal

Members

Jordan Christian, jchristi@andrew.cmu.edu
Cliff Rosenberg, carosenb@andrew.cmu.edu
Bryce Benjamin, bryceb@andrew.cmu.edu
Aarush Gupta, aarushg2@andrew.cmu.edu

Overview

We know how hard grocery shopping can be, especially for college students who typically have a tight budget to work with. We wanted to develop a tool that would allow someone to view different recipes, see the ingredients for that recipe, and then see the total price they would spend at different stores. We used 6 common recipes, Tacos, Burgers and Fries, Spaghetti, Pancakes, and Omelets. We compared prices from 3 different stores, Wal-Mart, Target, and Giant Eagle. Again, our tool simply shows the prices for a selected recipe at the three stores mentioned above.

```

Welcome
to

My Recipe Pal

The recipe app that gets you the lowest price for your favorite meals!
Please select which meal type you want to cook:

[1] Breakfast
[2] Lunch
[3] Dinner

[4] Quit

>
```

My Recipe App

The recipe app that gets you the lowest price for your favorite meals!

--BREAKFAST RECIPES--

Please select which recipe you want to cook:

[1] Pancakes and Eggs

[2] Omelette

[3] Back to main menu

>

Ingredients list:

- Pancake Mix
- Syrup
- Eggs

	Wal-Mart	Target	Giant	Eagle
Price For Meal	\$9.38	\$5.83		\$6.87

Do you want to get prices for another recipe? (Y/N):

Files

- **Group 9 MyRecipePal.py** – ****THIS IS THE ONLY FILE YOU NEED TO RUN****. This file is the driver of our application. It handles the frontend UI and connects the UI to the backend Sqlite3 database
- **insert_data.py** – This file creates the stores.db database, creates all the necessary tables to store the web scraped data, imports all web scraping files and inserts all data into the required tables
- **stores.db** – This file does not contain any code but is the binary file for the database. This binary file allows anyone module (AppUI.py) to connect to and use the data stored in the tables. **THIS FILE MUST BE IN THE SAME DIRECTORY AS AppUI.py IN ORDER TO ACCESS THE DATA CORRECTLY**
- **WalMartWebScrape.py** – This file scrapes all the ingredients from Wal-Marts website and stores the ingredients in a dictionary for each recipe. The key represents the products name while the value is its associated price
- **TargetScrape.py** - This file scrapes all the ingredients from Target's website and stores the ingredients in a dictionary for each recipe. The key represents the products name while the value is its associated price
- **GEWebScrape.py** - This file scrapes all the ingredients from Target's website and stores the ingredients in a dictionary for each recipe. The key represents the products name while the value is its associated price

How to Run:

Our application **DOES NOT** require any additional packages (this was tested using a fresh install of Anaconda).

1. Simply open **Group_9_MyRecipePal.py** and run the file. Again, it is essential that the stores.db binary file is in the same directory or else the data will not be accessible
2. Follow the instructions of the UI!