**Nutrition Tracker Using Python**

Group Member:

1.Siam Masud - 2011057642

2.Kh Md Nazmul Hossain - 2021894642

Github link : [siam3773/Nutrition-Tracker (github.com)](https://github.com/siam3773/Nutrition-Tracker)

Introduction

Our Project is a Nutrition tracker. We have made software that enables individuals to track their food’s nutritional information. This project aims to utilize Python programming language to create a user-friendly and interactive platform for accessing nutritional data.We have created an app that allows Users to search for a specific food, view detailed nutritional information, compare the nutritional profiles of 2 food items side by side and Display a sorted list of foods based on the nutrient with the highest value

The app uses an API that we have created to gather data on different foods, (such as carbohydrates, proteins, and fats), micronutrients, vitamins, minerals, and calories.

The user interface is designed using Python's GUI framework Tkinter providing an intuitive and visually appealing experience.

# Our Python-based project consists of Three Parts : Data collection, Api creation and Gui software.

# Project Overview

# Our Python-based project consists of Three Parts : Data collection, Api creation and Gui software.

# PythonAnywhere.com:

We have chosen to use [www.pythonanywhere.com](https://www.pythonanywhere.com/), a cloud-based Python development and hosting platform, to host our Python code for data collection and our API. This platform enables us to run our Python scripts and web applications in the cloud effortlessly. Furthermore, our database is securely stored on PythonAnywhere as well. The primary reason behind selecting PythonAnywhere is its ability to provide easy accessibility to our API for anyone interested in utilizing it. By leveraging this platform, we ensure that our API is readily available for anyone use.

# Data collection:

We collected our data by making API calls to "spoonacular.com" using Python. To interact with the API, we utilized the following Python libraries: `requests`, `json`, and `mysql.connector`.

Initially, we retrieved the data by accessing the API endpoint:

`api = "https://api.spoonacular.com/food/ingredients/{id}/information?apiKey=43585d37781e4241b34151b42eaa6cfd&includeNutrition=true&amount=100&unit=%22gm%22"`

Subsequently, we processed the received data and converted it into a dictionary data structure. This allowed us to conveniently work with the information retrieved from the API.

To store the collected data, we used MySQL as our database management system. We then used the `mysql.connector` library to establish a connection with the database.

The data processing involved iterating through each data entry, extracting relevant information such as the food's name and nutrient values, and constructing a dictionary to store this data.

To ensure data persistence, we created a table named "FoodsNew" in the MySQL database. This table was designed to have columns representing various nutrients. If the table didn't exist, we ensured its creation by executing a CREATE TABLE query. Subsequently, we inserted the extracted data into the "FoodsNew" table using an INSERT query.

# 

# 

# API Creation:

We created our own API with the collected data. using Flask.Here we used flask, mysql libraries of python.

We first connected to the MySQL database using the correct configuration, including the host, username, password, and database name.After that we created the API endpoints.The endpoints are:

1.[psyduck3773.pythonanywhere.com/NameList](https://psyduck3773.pythonanywhere.com/NameList) - This route retrieves all the food names from the `Foods` table using a SELECT query. The retrieved data is formatted into a list of dictionaries and returned as a JSON response.

2.[psyduck3773.pythonanywhere.com/foodsList](https://psyduck3773.pythonanywhere.com/foodsList) - This route retrieves all the records from the `Foods` table using a SELECT query. The retrieved data is formatted into a list of dictionaries and returned as a JSON response.

3.[psyduck3773.pythonanywhere.com](http://psyduck3773.pythonanywhere.com/)/SearchByName/<string:x> - This route accepts a parameter `x` representing the food name and performs a SELECT query to retrieve the matching records from the `Foods` table. The retrieved data is then formatted into a list of dictionaries and returned as a JSON response.

4. [psyduck3773.pythonanywhere.com/calories](https://psyduck3773.pythonanywhere.com/calories) - This route retrieves the food names and their corresponding calorie values from the `FoodsNew` table, sorted in descending order by calories. The retrieved data is formatted into a list of dictionaries and returned as a JSON response.

5.[psyduck3773.pythonanywhere.com/Protein](https://psyduck3773.pythonanywhere.com/Protein) - This route retrieves the food names and their corresponding protein values from the `FoodsNew` table, sorted in descending order by protein. The retrieved data is formatted into a list of dictionaries and returned as a JSON response.

6. [psyduck3773.pythonanywhere.com/Vitamin\_A](https://psyduck3773.pythonanywhere.com/Vitamin_A) - This route retrieves and sorts data based on Vitamin A.

7. [psyduck3773.pythonanywhere.com/Vitamin\_C](https://psyduck3773.pythonanywhere.com/Vitamin_C) - this route retrieves and sorts data based on Vitamin C.

8. [psyduck3773.pythonanywhere.com/Vitamin\_D](https://psyduck3773.pythonanywhere.com/Vitamin_D) - this route retrieves and sorts data based on Vitamin D

9.[psyduck3773.pythonanywhere.com/Vitamin\_E](https://psyduck3773.pythonanywhere.com/Vitamin_E) - this route retrieves and sorts data based on Vitamin E

# GUI Development:

We created an App with GUI using tkinter of python.Our app is easy to use and it shows the nutritional information of foods.We used our own API to collect and show information.We used the ‘requests’ library to call our API.

# The main functions in the code are:

# 1. `goHome()`: This function sets up the initial state of the GUI, including creating buttons for finding food, comparing foods, and sorting by highest values.

# 2. `find\_food()`: This function is called when the "Find Food" button is clicked. It makes an HTTP GET request to an API endpoint to retrieve a list of food items based on the given category ("foodsList", "Protein", "calorie", etc.). The retrieved data is then displayed in a listbox, and when a food item is selected, its details are shown in a text widget.

# 3. `compare\_foods()`: This function is triggered when the "Compare" button is clicked. It retrieves a list of food names from an API, creates dropdown menus for selecting two food items, and displays the nutritional information for the selected foods.

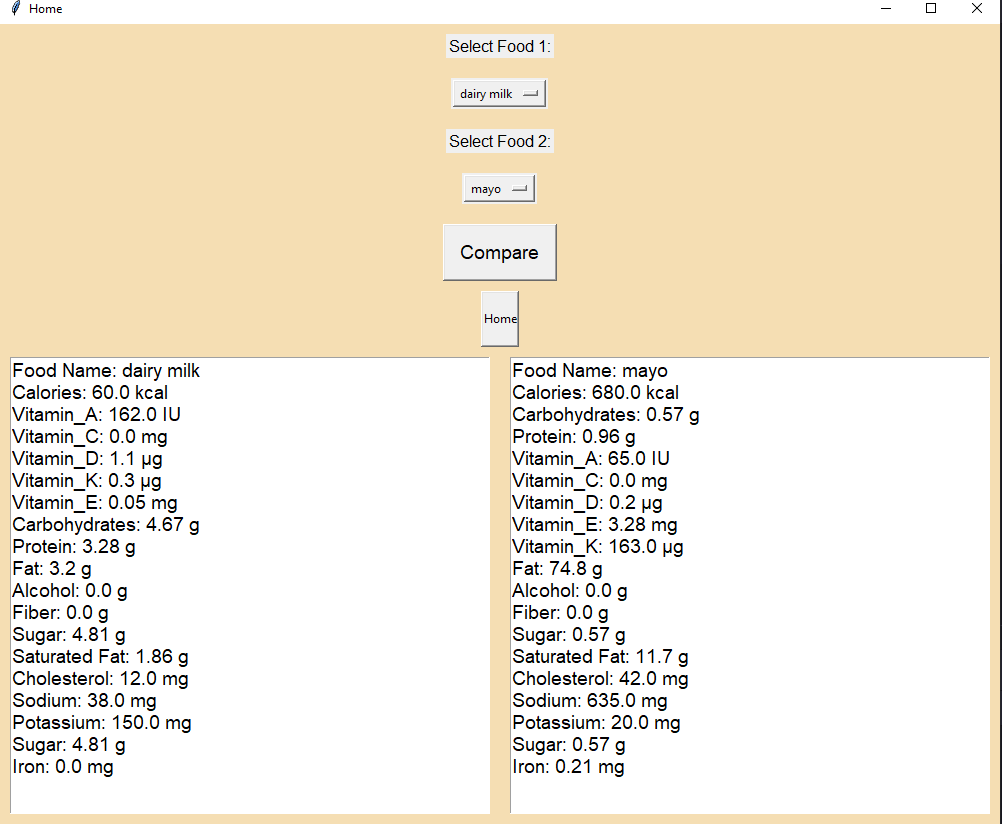
# 4. `sort\_by\_highest()`: This function is called when the "Sort by Highest" button is clicked. It displays buttons for different categories (e.g., Protein, Vitamin\_A, Calorie), and when clicked, it calls the `find\_food()` function with the corresponding category to display the sorted food items.

# 

# There are 3 buttons on the home page: Find Food , Compare, Sort by Highest.

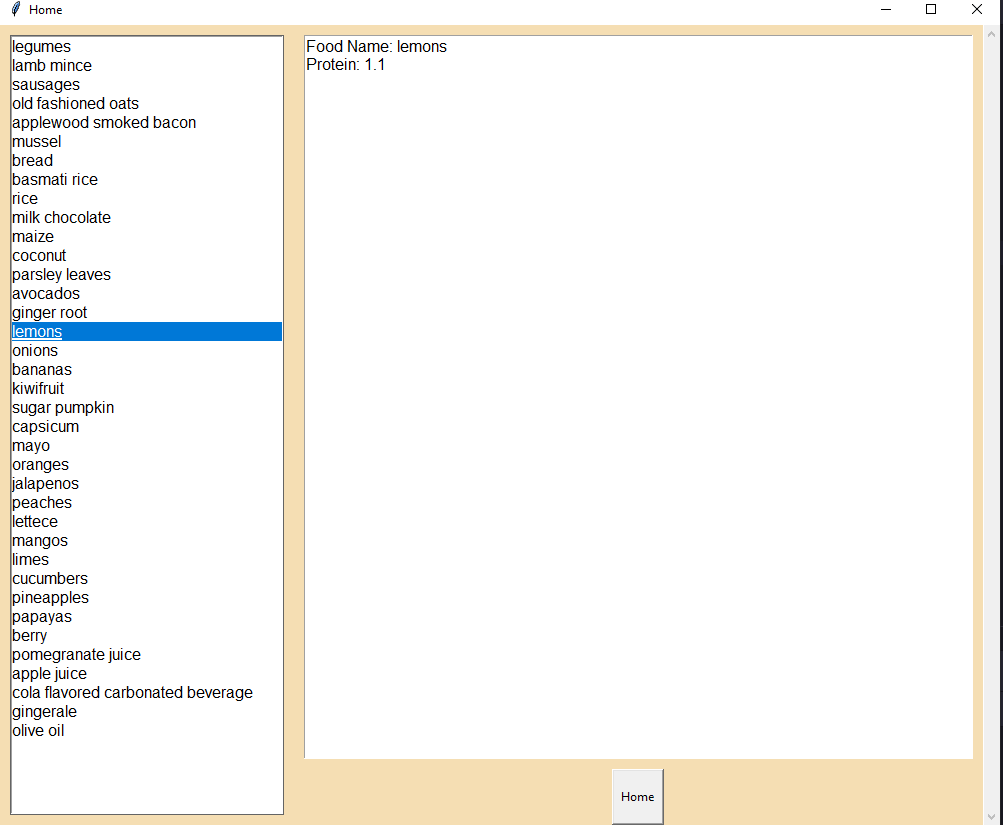
# 

By clicking the find food button the user can see nutritional values of a food. The food list is scrollable.



By clicking on the compare button the user can compare between 2 foods.

By clicking on Sort by the highest the user can choose the food to see the sorted list based on a nutrient.



If the user clicks on the protein button the user can see the sorted list for foods that have the highest protein.