Omar Al Taher
Recipe Recommendation Program
7/18/2016

Project README

Purpose:

The program will take an input of ingredients, and return a list of recipes that can be made using those ingredients. The user can select one of the recipes for display. An an option, the user can display an image of the recipe in their browser.
A planned option to convert any recipe into a vegetarian or vegan option was not implemented due to the complexity being higher that anticipated.

Entities:

Classes:

Recipe class:

Class Attributes:
all_ingredients_list: takes unfiltered ingredients as scraped from allingredients.com
recipe_list, recipe_dict: list and dictionary, respectively, of all initialized instances of Recipe Class
pure_ingredient_dict: dictionary of ingredients that is used to create instances of Ingredient Class, and for performing searches based on user input ingredients

Instance Attributes:
title: recipe title
number: recipe number within the program
time: total recipe time in minutes
ingredient_list: recipe ingredients with measurements and description, as scraped from allrecipes.com
instructions: recipe cooking instructions, as scraped from allrecipes.com
image_link: hyperlink for an image of the recipe

Methods:
__init__: initializes recipe instances, and adds the recipe to recipe_dict, recipe_list, and adds ingredient_list to all_ingredient_list
ingredients (@property): cleans ingredient_list, due to minor issue with the scraper script
search_ingredients (@property): returns individual Ingredient objects, to be searched against input ingredients
lookup_byname: looks up the recipe object by name
lookup_bynumber: looks up the recipe object by number
display_image: displays image_link in browser
display_recipe: displays the complete recipe text

Ingredient Class:

Class Attributes:
ingredient_dict: of all initialized instances of Ingredient Class names as keys, and a list of the object, veggie_flag, vegan_flag, and alternate names as values

Methods:
__init__: initializes Ingredient instances, and adds the Ingredient to ingredient_dict
__str__: returns the name of the Ingredient object
alt_name: returns list of potential alternate names to Ingredient

Functions:

find_recipe(ingredient_list): this function takes an input of up to three ingredients to search for in the recipes that are loaded into the program. The function returns a print-out of the recipes that match all the ingredients, some of the ingredients, and each individual ingredient. Each recipe is displayed with its recipe number, for easy retrieval and display.

Omar Al Taher
Recipe Recommendation Program
7/18/2016

Challenges:

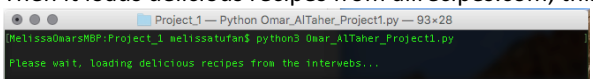
1) The biggest challenge was finding recipe data to use with the program. I tried to find recipe files that were open source and available online, but could not find any with a suitable format and selection of recipes. Some of the recipe websites have their own systems, but I could not find a way to get to the source data files. I did find one set of files that were available, but they were using a proprietary xml format that while probably decipherable, was out of the scope of this project.
2) The second challenge was understanding the way I need to structure my classes, lists, and functions in order to achieve the result I wanted. I found myself iterating often, and by the time I was done, I found many of the elements I built were not necessarily needed (like the alt_names attribute for Ingredient class instances).
3) The third challenge, was understanding how to initiate many instances of a class, and being able to keep track of them. This was finally accomplished by using list comprehensions running off of other lists containing information needed to initialize instances of each object. This is really due to the different thinking approach needed for object-oriented programming, which I found a bit challenging.
4) The last big challenge was adding handling of plural ingredient names. This seemed straightforward, but proved to be time consuming.
5) Time. Not enough time to do what I wanted. Namely: the veggiefy and veganize options, which would be difficult to add. Less difficult would be the ability to add recipe and ingredient objects from the program while it is running.

Use and Testing

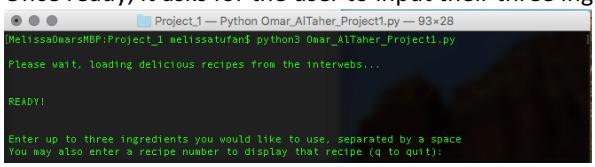
User Interface:

Program is run by initiating it in python 3, via the command shell

Then it loads delicious recipes from allrecipes.com, this takes 30-60 seconds



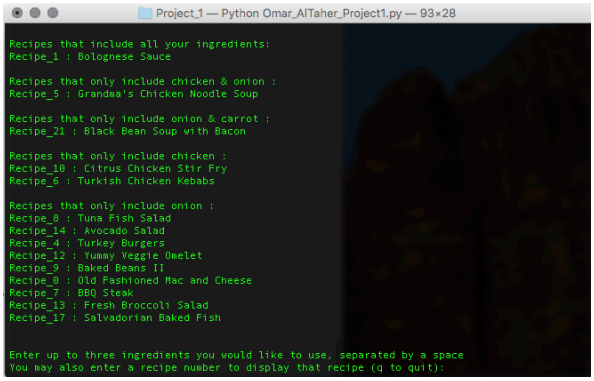
Once ready, it asks for the user to input their three ingredients, or a recipe number, or q to quit



If user enters three ingredients

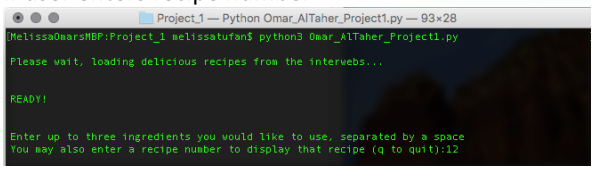
the program returns a list of recipes:

Omar Al Taher
Recipe Recommendation Program
7/18/2016

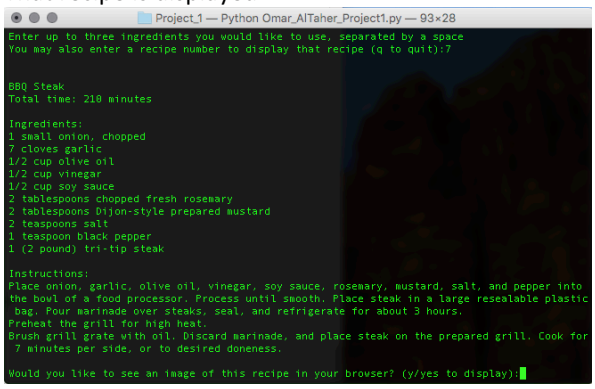


The user can enter the recipe number to display it

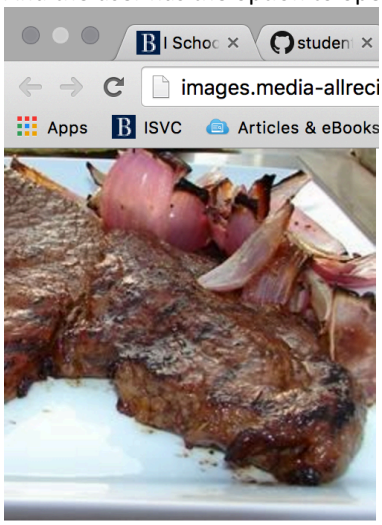
If user enters recipe number



That recipe is displayed

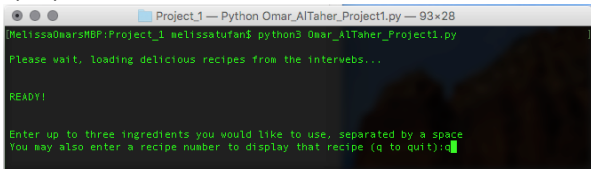


And the user has the option to open an image of the recipe in their browser



If user enters q, the program quits!

Omar Al Taher
Recipe Recommendation Program
7/18/2016



Testing:

The program can be tested by entering different ingredients, displaying recipes and their images. Different "nonsense" inputs, or different numbers of inputs can be tried to see if that crashes the program. The program should also tolerate plural vs singular ingredients, returning the same results for both.