Programming Principles (CSP1150)

Assignment 2: Individual programming project ("One Must Go" Program)

Name: Savitha Fernando

Student Number: 10705112

Github Repository: https://github.com/savithafdo/One-Must-Go

Pseudocode (admin.py)

Overview

This program lets you create categories with up to five options each. People can later record votes for those options and you can list, search, view, or delete categories. Your categories are saved to a file so they are there the next time you run the program.

Start Up Steps

- Try to open a file named "data.txt". If it exists and contains valid JSON, load it into a variable called data.
- If something goes wrong (for example, the file does not exist or the content is broken), show a short message and start with an empty list.
- Show a welcome message.
- Keep showing the menu until the user chooses to quit.

Menu Choices

- [a] add a new category
- [l] list all categories
- [s] search categories
- [v] view a single category's results
- [d] delete a category
- [q] quit the program

Choose 'a' (Add a category)

You create a new category and add its options.

- Ask for a category name. Do not allow an empty name.
- If a category with the same name already exists (ignore upper/lower case), ask again until the name is unique.
- Start an empty list of options.
- Ask for option text one at a time using input something.
- You can add up to 5 options.
- If the user types 'x' and there are already at least 2 options, stop asking for more.
- If the option text matches an existing option (ignore case), ask again.
- Otherwise, add the option as {"name": <text>, "votes": 0}.
- When finished (reached 5 options or typed "x"), make {"name": <name>, "options": <options>} and append it to data.
- Save everything by calling save_data(data).

Choose 'l' (List categories)

You see a summary of each category.

- If there are no categories, show, "No categories saved".
- Otherwise, show each number, name, count of options (N), and total votes (T).
  Example: 1) Snacks (3 options, 12 votes).

When you choose 's' (Search)

You find categories by a term in the name or option names.

- If there are no categories, show a message and stop.
- Ask for a search term (not empty).
- For each category, it matches if, the term appears in the category name (ignore case), or the term appears in any option name (ignore case).
- Show all matches like the list view.
- If nothing matched, show, "No results found".

When you choose 'v' (View results)

You see the vote counts for one category.

- If there are no categories, show a message and stop.
- Ask for the category number (1 to how many categories exist).
- Show the category name.
- If all option votes are 0, show, "No votes recorded".
- Otherwise, show each option and its votes (you may sort by highest first).

When you choose 'd' (Delete a category)

You remove a category from the file.

- If there are no categories, show a message and stop.
- Ask for the category number (1 to how many categories exist).
- Remove that category from data.
- Save the file again by calling save_data(data).
- Show, "Category deleted".

When you choose 'q' (Quit) and other cases

- If the user chooses 'q', show a goodbye message and stop the loop.
- If the user enters anything else that is not recognized, show "Invalid choice".

Functions Explained

input_something(prompt)

- Keep asking for text until the user types something that is not just spaces.
- Return that text.

input_int(prompt, max_value)

- Keep asking until the user enters a whole number.
- If the number cannot be converted or is not between 1 and max_value, remind the user and ask again.
- Return the valid number.

save_data(data)

- Open the file "data.txt" for writing.
- Convert data to JSON with indentation so it is easy to read.
- Write it to the file and close the file.