Lab 4.04 - Shopping Lists

Part 1

The goal of this lab is to practice using and accessing items from lists of lists.

You have a few errands to run and have created a few shopping lists to help you remember what to buy. You stored your notes in a nested list, `shopping_cart`. This program will allow the user to ask for a specific item by its index or update what items are in the cart. The user can request to `view list` to see the items in a specific shopping list.

Shopping Lists

```
shopping_lists = [
    ['toothpaste', 'q-tips', 'milk'],
    ['milk', 'candy', 'apples'],
    ['planner', 'pencils', 'q-tips']
]
```

User Inputs

`1` - Update

- The program asks which shopping list the user wants to update, which position it should update, and the new value to update.

`2` - View Item

- The program asks which shopping list the item is on and which position it occupies, then prints the item's name.

`3` - View List

- The program asks which shopping list the user wants and prints all of the items associated with that shopping list.

Functions

`update_list`

- Takes in an integer representing the index of the shopping list, an integer representing the index of the item to update, and a string representing the new value for that item. Does not alter the length of the list.

`print_item`

- Takes an `int` representing the index of the shopping list followed by an `int` representing the index of the item to print.

`print_list`

- Takes an `int` representing the index of the shopping list to print.
- Feel free to add more functions as you see fit.

Example

```
>>> Choose 1 = update item, 2 = view item, or 3 = view list: 1
Which shopping list would you like to update? 1
Which item would you like to change? 2
New value for item #2? cheese
toothpaste, cheese, milk
>>> Choose 1 = update item, 2 = view item, or 3 = view list: 2
Which shopping list do you want to choose? 2
Which item on list #2 do you want to see? 3
apples
>>> Choose 1 = update item, 2 = view item, or 3 = view list: 3
Which shopping list would you like to see? 3
planner, pencils, q-tips
```

Part 2

In this part of the lab you will go through your shopping list program and perform a few different calculations.

1. Create a function, `all_in_one`, that will put all the shopping lists into a single combined list using a `for` loop.
2. Create a function, `count_q_tips`, which will go through all items of the list and keep a count of how many times `'q-tips'` occurs.
3. In order to make the shopping lists more calcium rich, write a function, `drink_more_milk`, that adds `'milk'` to each of the lists (unless it's already there).
4. You can't have milk without cookies. Write a function `if_you_give_a_moose_a_cookie`, that will go through every element of `shopping_cart` and update `'milk'` to be `'milk and cookies'`.

Bonus

Write a function to reverse the order of the lists, and also reverse the order of the items in each list, in the `shopping_cart` nested list.

The new reversed list should look like the following when printed (newlines and spacing added for clarity):

```
shopping_cart = [
    ['q-tips', 'pencils', 'planner'],
    ['apples', 'candy', 'milk'],
    ['milk', 'q-tips', 'tooth paste']
]
```

Tip

- Last item can be gotten by `my_list[-1]`

- Second to last element: `my_list[-2]`
- Third to last element: `my_list[-3]`