

# Lab 4.04 - Shopping List ## 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 list 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 it's 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 Cart ```python shopping\_cart = [ ['tooth paste', 'q-tips', 'milk'], ['milk', 'candy', 'apples'], ['planner', 'pencils', 'q-tips'] ] ``` ### User Inputs `update` \* The program asks which shopping list the user wants to update, which position it should update, and the new value to update. `view item` \* The program asks which shopping list the item is on and which position `i` occupies, then prints the items name. `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 ```python >>>What would you like to do? view list Which shopping list would you like to see? 1 tooth paste, q-tips, gum ``` ## 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 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 items in `shopping\_cart`. The list should look like the following when printed: ```python 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]`