CS469 Week 2

Description

This assignment is to gain knowledge on Linked List and Selection Sort. A shopping list will be implemented for this exercise. The program stores shopping items in both simple array or linked list. Each storing mechanism is separated into different files, such as "simple_array_shopping_list_manager.py" or "linked_list_shopping_list_manager.py," where each file contains FileNameClass classes with essential methods for data manipulation. Note that "shopping_list.py" with the "main" method has already been provided (download attachment). As part of the assignment, compare the actual runtime of insert/delete/lookup operation between two lists and justify in a short paragraph on how the algorithms perform. Keep in mind to always comment and document your class and methods.

Documentation reference

Mertz, J. (n.d.). Documenting Python Code: A Complete Guide. https://realpython.com/documenting-python-code/

Expected result

`shopping_list.py`

This is Main python file which is already provided and contains main and test procedure which calls methods implemented on "simple_array_shopping_list_manager.py" or "linked_list_shopping_list_manager.py" file to manage shopping items. (This is already provided, but please include this file on your submission).

`simple_array_shopping_list_manager.py`

This is a file that includes the class of simple array based shopping list manager. This class contains such methods as `init`, `insert_item`, `print_items`, `delete_item`, `get_last_item`, `selection_sort`. Please keep in mind the following notes for each method during implementation:

1. `Init()`: initializes simple array to be used throughout object life.
2. `insert_item(item)`: inserts item at the front of the array.
    - **Parameters**: item name. Note that you are allowed to use the insert() method from Python Array Module.
3. `print_items()`: simply prints item
4. `delete_item(item)`: deletes item. **Parameters**: item name. Note that for runtime study purpose, DO NOT use remove() method from Python Array module but you are allowed to use pop()/index() method
5. `get_last_item()`: returns the last item name from the simple array.
6. `selection_sort()`: sorts items on the array from the current object and replaces it with the newly sorted result.

linked_list_shopping_list_manager.py

This is a file that includes the class of linked list based shopping list manager. This class contains such methods as init, insert_item, print_items, delete_item, get_last_item, selection_sort. In addition, this class requires the inner class to hold onto data as a linked list. Please keep in mind the following notes for each method during implementation:

1. `Init()`: initializes linked list object to be used throughout object life.
2. `insert_item(item)`: inserts item at the front of the linked list. Parameters: item name.
3. `print_items()`: simply prints item throughout linked list. Note that try to print as [item1 item2] by using some combinations of "print(item, end = " ")".
4. `delete_item(item)`: deletes item. Parameters: item name.
5. `get_last_item()`: returns the last item name from the linked list.
6. `find_smallest()`: returns the smallest item from linked list. (ex. [apple banana cheese] -> apple is smallest)

`insert/delete/lookup`

As part of the assignment, compare the actual runtime of insert/delete/lookup operation between two lists and justify in a short paragraph on how and their performance.

Example output:

Challenge question (not mandatory)

Implement the following selection sort for the linked list class. Note that this test case has been commented out on the Main class.

find_smallest(linkedList): modification is required to take in the linked list object parameter, and returns the smallest item from the given linked list.
selection_sort(): sorts items on the linked list from the current object and replaces it with the newly sorted result.