What do I need to buy today?

Description:

Mike is terribly forgetful and wants a C++ program to keep track of his shopping list (he feels that these new-fangled smart phones are nothing but a fad and wants to go back to good old console input and output).

Write a shopping list program for Mike that achieves all constraints outlined below and as many of the criteria as you want. You may assume that Mike never enters more than 100 items in his lists. You are free to design whatever data structures you want to solve this problem. It may be as complicated or as simple as you wish.

Constraints

1) Mike must be able to enter items or receive a nicely formatted shopping list to print.
2) Mike must be able to enter both an item name and, optionally, a quantity that he wants to purchase.
3) Mike stops entering items when he enters the item code NO_MORE_ITEMS. Once he's done entering items, he must be given the option to either quit the program or print the list immediately. If he chooses to quit, the list should be saved.

Criteria

1) If a saved list exists and Mike wants to add items to the shopping list, he should be asked if he wants to overwrite the existing list or add to it. Your program should then handle both situations.
2) Mike is **very** forgetful, and will sometimes enter the same item twice. You should remind him that he's already entered the item. You may then decide what to do from the list below.
    a. Level 1: Do not enter the item again.
    b. Level 2: Add the quantity of the new item entry to that of the old item entry. If Mike chose not to put in a quantity, assume that the quantity is one unit.
3) Mike shops for many different things at different stores. He thinks it would be nice if he could make separate lists for as many stores as he wants, and recall them by entering the exact file name.