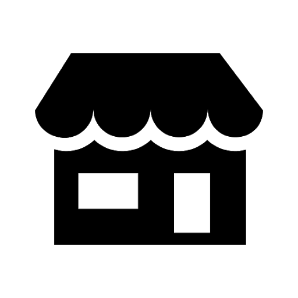
# Problem 3 - Shopping



*You are given some notes with products you need to buy, and you should order them correctly in a notepad before you start circling around the town's stores.*

Create a program that lists **stores** and the **items** that can be found in them. You will be receiving **commands** with the information you need until you get the "**Go Shopping**" command. There are **three possible commands**:

* **"****Add->{Store}->{Item},{Item1},…{ItemN}"**
  + **Add** the **store** and the **items** to your notepad.
  + If the **store** already **exists**, just **add the additional items** which need to be bought from it.
  + **All items in the notepad should be unique! If an item is already added to your notepad, no matter in which store, do not add it.**
* **"****Important->{Store}->{Item}"**
  + It is a **major item** which must be bought! So, you need to **add** the **item** at the **top of the store's list**.
  + If the **store does not exist**, **add it** to your notepad along **with the item**.
  + **If an item is already in your notepad, no matter in which store, do not add it.**
* **"****Remove->{Store}"**
  + **Remove the store** and its items from your notepadif **it exists**.
  + **If an important item exists in the store's list, ignore the command.**

In the end, print the storesin the following format:

**"{store}:**

**- {item1}**

**- {item2}**

**…**

**- {itemN}"**

## Input / Constraints

* You will be receiving information until the **"Go Shopping"** command is given.
* There will always be **at least one** store in the notepadand **at least one item** in it.
* In the end, you will **never have** an **empty** store's list.

## Output

* Print the list of stores in the format given above.

## Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Add->Grocery->Dried-fruit,Nuts,Lemons  Add->Market->Nuts,Juice  Important->Market->Snack  Remove->Market  Go Shopping | Grocery:  - Dried-fruit  - Nuts  - Lemons  Market:  - Snack  - Juice |
| Add->Peak->Batteries,Umbrella  Add->Groceries->Water,Juice,Food  Add->Peak->Tent  Important->Groceries->Batteries  Remove->Store  Go Shopping | Peak:  - Batteries  - Umbrella  - Tent  Groceries:  - Water  - Juice  - Food |
| Add->Peak->Batteries,Umbrella  Add->Groceries->Water,Juice,Food  Add->Peak->Water  Go Shopping | Peak:  - Batteries  - Umbrella  Groceries:  - Water  - Juice  - Food |

## JS Examples

The input will be provided as an array of strings.

|  |  |
| --- | --- |
| **Input** | **Output** |
| (["Add->Grocery->Dried-fruit,Nuts,Lemons",  "Add->Market->Nuts,Juice",  "Important->Market->Snack",  "Remove->Market",  "Go Shopping"]) | Grocery:  - Dried-fruit  - Nuts  - Lemons  Market:  - Snack  - Juice |
| (["Add->Peak->Batteries,Umbrella",  "Add->Groceries->Water,Juice,Food",  "Add->Peak->Tent",  "Important->Groceries->Batteries",  "Remove->Store",  "Go Shopping"]) | Peak:  - Batteries  - Umbrella  - Tent  Groceries:  - Water  - Juice  - Food |
| (["Add->Peak->Batteries,Umbrella",  "Add->Groceries->Water,Juice,Food",  "Add->Peak->Water",  "Go Shopping"]) | Peak:  - Batteries  - Umbrella  Groceries:  - Water  - Juice  - Food |