

U2: Abstraction

L2: Making Lists

P1: Shopping List App

Aim: How do we write scripts that add items to and delete items from a list?

Unit 2 Lab 2 Page 1

Making Lists

Objectives: Shopping List App

In this lab, you will create tools for storing and accessing data.

On this page, you will create a shopping list app.

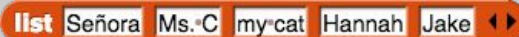


Beauty and Joy of Computing by University of California, Berkeley and Education Development Center, Inc. is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/). The development of this site is funded by the National Science Foundation under grant number 1441075. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Many computer applications—contact lists, playlists, calendars, reminders—involve manipulating *lists* of information using tools that search, sort, or change the items on the list. You've worked with lists before as you customized the gossip project.

Some programming languages use the word **array list** instead of **list**.

On the AP Exam...


The list expression  would be written as `[Señora, Ms. C, my cat, Hannah, Jake]`

The items are positioned in the list in the order they appear in the text: "Señora" has index 1, "Ms. C" has index 2, and so on.

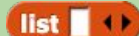



Remember to Do Pair Programming Swaps

For You To Do

1.  Click here to load. Save to your Snap! account. This project contains button costumes for the sprites that the user will click to control. **Save as U2L2-ShoppingList.**

[Click for a video showing how this might look.](#)

2. Initialize a **shopping list** variable to store the information.
 - a. Make a global variable called **shopping list**.
 - b. Set **shopping list** to an *empty* list (a list with nothing in it). You will need a  block.

Use its left arrow to get rid of the input slot so that it looks like this: . That way, you will have an empty list rather than a list with one empty item.



On the AP Exam...

The assignment instruction **set shopping list to list** would be written as `shoppingList ← []` or `shoppingList ← []`

The assignment instruction **set shopping list to list apples bread carrots rice pasta** would be written as `shoppingList ← [apples, bread, carrots, rice, pasta]` or `shoppingList ← [apples, bread, carrots, rice, pasta]`

(In this app, you'll want the **shopping list** to begin empty, and then the user will **add** or **insert** additional grocery items one at a time.)

You can use the **insert** at 1 of **add** to **lock** to add elements to a list.

- **Insert** puts the new item *before* the place you specify.
- **Add** puts the item *after* the last existing item.

Vocabulary:

An **element** is another name for an item in a list. (If the same value is in the list twice, that counts as two different elements.) Each element has a unique **index** (position) in the list.

Assigning a list to a variable lets you use one name to represent all the elements of a list as a unit.

On the AP Exam...

insert tomatoes at 2 of shopping list would be written as `INSERT(shoppingList, 2, "tomatoes")` or `INSERT [shoppingList, 2, tomatoes]`

add tomatoes to shopping list would be written as `APPEND(shoppingList, "tomatoes")` or `APPEND [shoppingList, tomatoes]`




Beauty and Joy of Computing by University of California, Berkeley and Education Development Center, Inc. is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/). The development of this site is funded by the National Science Foundation under grant number 1441075. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

For You To Do

3. Write a script for the "Add Item" button sprite so that when that when that sprite is clicked, it will **ask** the user for a new item, and then put the user's answer in the grocery list.
4. Test your "Add Item" button several times and fix any problems.
5. Write a script for the "Clear List" button sprite that *asks the user if they're sure* and then sets **shopping list** back to an empty list.
6. Test your "Clear List" button and fix it if needed.

You've seen the **ask** and **answer** blocks on [Unit 2 Lab 1 Page 2: Checking the Player's Guess](#).

You've worked with multiple sprites on [Unit 1 Lab 2 Page 2: Making Programs Talk](#).

You can also remove items from a list using . The **delete** block takes an item number and a list as input and it removes the item at that position from the list.

On the AP Exam...

 would be written as `REMOVE (shoppingList, 3)` or 



Beauty and Joy of Computing by University of California, Berkeley and Education Development Center, Inc. is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#). The development of this site is funded by the National Science Foundation under grant number 1441075. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.


For You To Do

7. Write a script for the "Delete Item" button sprite so that when that sprite is clicked, it will ask the user to "Enter the number of the grocery item you wish to delete," and then remove the item with that number from the grocery list.
8. Test your "Delete Item" button.

On the AP Exam...

The items in a list are *values*, so **you can use item of anywhere you can use any other expression**. For example:

- You can assign a list item to a variable. On the exam...

○  would be written as `myFavoriteFood ← shoppingList[3]` or

`myFavoriteFood ← shoppingList [3]`

- You can assign *any* value to a list item (a number, a string, a sprite, a costume, a script, another list, etc.). On the exam...

○  would be written as `shoppingList[2] ← yourFavoriteFood`

OR `shoppingList [2] ← yourFavoriteFood`

○  would be written as `shoppingList[1] ← shoppingList[3]`

OR `shoppingList [1] ← shoppingList [3]`



On the AP Exam...

When you run this script in Snap!, the second line of code assigns to **shopping list 2** the value of **shopping list** (that is, the same list, not a copy). So, the third line of code modifies *both* variables:



However on the exam, the statement `shoppingList2 ← shoppingList` makes a *copy* of the list. So modifying one of them *does not* modify the other.

The rules for how to use lists and how they behave differ depending on the programming language you are using.

After Class (Written Tasks section)

U2L2P1Journal:

1. Summarize what you learned from the ***On the AP Exam*** sections in this lab.

Do the lists inside a list need to be the same size? Why or why not?

