

Alternate Project 2: TODO List

Created By Brian Weinfeld

Using Python, you will create an interactive TODO list that allows users to add and remove tasks from a list.

Overview

One of the most common tasks we use computers for is to store and track data. Perhaps the most simple form of this task is the TODO list. A TODO list is simply a list of information (for example, chores that needs to be done) that the user wants to track. They can add tasks to the list, remove them when they are complete and examine the list as they progress. Have you ever used a notes app on your phone?

Details

Behavior

```
Welcome to your TODO list
What would you like to do? (add, remove, list, exit) list
[]

What would you like to do? (add, remove, list, exit) add
What would you like to add to your list? go to gym
Successfully added
What would you like to do? (add, remove, list, exit) list
['go to gym']

What would you like to do? (add, remove, list, exit) add
What would you like to add to your list? buy food
Successfully added
What would you like to do? (add, remove, list, exit) list
['go to gym', 'buy food']

What would you like to do? (add, remove, list, exit) add
What would you like to add to your list? go to gym
This is already on your list!
What would you like to do? (add, remove, list, exit) list
['go to gym', 'buy food']

What would you like to do? (add, remove, list, exit) remove
What would you like to remove? go to gym
Sucessfully removed
What would you like to do? (add, remove, list, exit) list
['buy food']

What would you like to do? (add, remove, list, exit) remove
What would you like to remove? clean bedroom
This is not in the list!
What would you like to do? (add, remove, list, exit) list
['buy food']
```

```
What would you like to do? (add, remove, list, exit) exit
Goodbye
```

Implementation Details

- The program offers the user 4 options. **add** will add elements to the TODO list. **remove** will remove elements from the list. **list** will display the entire list and **exit** will exit the program.
- After a user selects **add** they should then be prompted for the item they want added to the list. Be sure to check before you add the item to the list. You don't want to add the item twice!
- After a user selects **remove** they should then be prompted for the item they want removed from the list. Be sure to check before removing the item from the list. You should let the user know if they are trying to remove an item that isn't in the list!

Challenge

This section contains additional components you can add to the project. These should only be attempted **after** the project has been completed.

- It is a bit inconvenient to have to type two commands to add or remove an element from the list. Modify the program so that tasks can be added or removed from the TODO list with a single command like **add go to gym** or **remove go to gym**.

```
Welcome to your TODO list
What would you like to do? (add, remove, undo, list, exit) add go to gym
Successfully added
What would you like to do? (add, remove, undo, list, exit) list
['go to gym']

What would you like to do? (add, remove, undo, list, exit) add clean bedroom
Successfully added
What would you like to do? (add, remove, undo, list, exit) list
['go to gym', 'clean bedroom']

What would you like to do? (add, remove, undo, list, exit) remove bedroom
This is not in the list!
What would you like to do? (add, remove, undo, list, exit) remove clean bedroom
Successfully removed
What would you like to do? (add, remove, undo, list, exit) list
['go to gym']
```

- It is helpful to have a feature to immediately remove the most recently added element to the list. This is often because the user made a mistake in adding the element to the list in the first place. Add an **undo** option to the program that will remove the most recently added item. If the item is no longer in the list, print an error.

```
Welcome to your TODO list
What would you like to do? (add, remove, undo, list, exit) add go to gym
Successfully added
What would you like to do? (add, remove, undo, list, exit) add clean bedroom
Successfully added
What would you like to do? (add, remove, undo, list, exit) list
['go to gym', 'clean bedroom']

What would you like to do? (add, remove, undo, list, exit) undo
Successfully undid last add.
What would you like to do? (add, remove, undo, list, exit) list
['go to gym']

What would you like to do? (add, remove, undo, list, exit) undo
Cannot undo. You already removed this item
What would you like to do? (add, remove, undo, list, exit) list
['go to gym']
```

Super Challenge

The super challenge will require knowledge that has not been taught yet. You will need to do additional research on your own. Good luck!

It is possible to track enough information so that the **undo** command can be called repeatedly. Modify the **undo** command so that it will always work by removing the most recently added element in the TODO list that is still in the list. The only time the command should do nothing is when the TODO list is empty.