

CS50's Introduction to Programming with Python

OpenCourseWare

Donate  (<https://cs50.harvard.edu/donate>)

David J. Malan (<https://cs.harvard.edu/malan/>)

malan@harvard.edu

 (<https://www.facebook.com/dmalan>)  (<https://github.com/dmalan>) 

(<https://www.instagram.com/davidjmalan/>)  (<https://www.linkedin.com/in/malan/>)

 (<https://www.reddit.com/user/davidjmalan>) 

(<https://www.threads.net/@davidjmalan>)  (<https://twitter.com/davidjmalan>)

Grocery List

Suppose that you're in the habit of making a list of items you need from the grocery store.

In a file called `grocery.py`, implement a program that prompts the user for items, one per line, until the user inputs control-d (which is a common way of ending one's input to a program). Then output the user's grocery list in all uppercase, sorted alphabetically by item, prefixing each line with the number of times the user inputted that item. No need to pluralize the items. Treat the user's input case-insensitively.

▼ Hints

- Note that you can detect when the user has inputted control-d by catching an `EOFError` (<https://docs.python.org/3/library/exceptions.html#EOFError>) with code like:

```
try:
    item = input()
except EOFError:
    ...
```

- Odds are you'll want to store your grocery list as a `dict`.
- Note that a `dict` comes with quite a few methods, per docs.python.org/3/library/stdtypes.html#mapping-types-dict (<https://docs.python.org/3/library/stdtypes.html#mapping-types-dict>), among them `get`, and supports operations like:

```
d[key]
```

and

```
if key in d:  
    ...
```

wherein `d` is a `dict` and `key` is a `str`.

- Be sure to avoid or catch any `KeyError` (<https://docs.python.org/3/library/exceptions.html#KeyError>).

Demo

```
$ python grocery.py  
apple  
banana
```

Recorded with [asciinema](#)

Before You Begin

Log into cs50.dev (<https://cs50.dev/>), click on your terminal window, and execute `cd` by itself. You should find that your terminal window's prompt resembles the below:

```
$
```

Next execute

```
mkdir grocery
```

to make a folder called `grocery` in your codespace.

Then execute

```
cd grocery
```

to change directories into that folder. You should now see your terminal prompt as `grocery/ $`. You can now execute

```
code grocery.py
```

to make a file called `grocery.py` where you'll write your program.

How to Test

Here's how to test your code manually:

- Run your program with `python grocery.py`. Type `mango` and press Enter, then type `strawberry` and press Enter, followed by control-d. Your program should output:

```
1 MANGO
1 STRAWBERRY
```

- Run your program with `python grocery.py`. Type `milk` and press Enter, then type `milk` again and press Enter, followed by control-d. Your program should output:

```
2 MILK
```

- Run your program with `python grocery.py`. Type `tortilla` and press Enter, then type `sweet potato` and press Enter, followed by control-d. Your program should output:

```
1 SWEET POTATO
1 TORTILLA
```

You can execute the below to check your code using `check50`, a program that CS50 will use to test your code when you submit. But be sure to test it yourself as well!

```
check50 cs50/problems/2022/python/grocery
```

Green smilies mean your program has passed a test! Red frownies will indicate your program output something unexpected. Visit the URL that `check50` outputs to see the input `check50` handed to your program, what output it expected, and what output your program actually gave.

How to Submit

In your terminal, execute the below to submit your work.

```
submit50 cs50/problems/2022/python/grocery
```

