

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>)

Pizza Py



Perhaps the most popular place for pizza in [Harvard Square](https://en.wikipedia.org/wiki/Harvard_Square) (https://en.wikipedia.org/wiki/Harvard_Square) is [Pinocchio's Pizza & Subs](https://www.pinocchiopizza.net/) (<https://www.pinocchiopizza.net/>), aka Noch's, known for its [Sicilian pizza](https://www.pinocchiopizza.net/sicilian_vs_regular.html) (https://www.pinocchiopizza.net/sicilian_vs_regular.html), which is “a deep-dish or thick-crust pizza.”

Students tend to buy pizza by the slice, but Pinocchio's also has whole pizzas on its [menu](https://www.pinocchiopizza.net/menu.html) (<https://www.pinocchiopizza.net/menu.html>) too, per this CSV file of Sicilian pizzas, [sicilian.csv](#), below:

```
Sicilian Pizza,Small,Large
Cheese,$25.50,$39.95
1 item,$27.50,$41.95
2 items,$29.50,$43.95
3 items,$31.50,$45.95
Special,$33.50,$47.95
```

See [regular.csv](#) for a CSV file of regular pizzas as well.

Of course, a CSV file isn't the most customer-friendly format to look at. Prettier might be a table, formatted as [ASCII art](https://en.wikipedia.org/wiki/ASCII_art) (https://en.wikipedia.org/wiki/ASCII_art), like this one:

```
+-----+-----+-----+
| Sicilian Pizza | Small  | Large  |
+=====+=====+=====+
| Cheese        | $25.50 | $39.95 |
+-----+-----+-----+
| 1 item        | $27.50 | $41.95 |
+-----+-----+-----+
| 2 items       | $29.50 | $43.95 |
+-----+-----+-----+
| 3 items       | $31.50 | $45.95 |
+-----+-----+-----+
| Special       | $33.50 | $47.95 |
+-----+-----+-----+
```

In a file called `pizza.py`, implement a program that expects exactly one command-line argument, the name (or path) of a CSV file in Pinocchio's format, and outputs a table formatted as ASCII art using `tabulate`, a package on PyPI at pypi.org/project/tabulate (<https://pypi.org/project/tabulate/>). Format the table using the library's `grid` format. If the user does not specify exactly one command-line argument, or if the specified file's name does not end in `.csv`, or if the specified file does not exist, the program should instead exit via `sys.exit`.

► Hints

Demo

```
$ python pizza.py
Too few command-line arguments
$ python pizza.py foo
Not a CSV file
$ python pizza.py foo bar
Too many command-line arguments
$
```

Recorded with **asciinema**

Before You Begin

Log into [cs50.dev \(https://cs50.dev/\)](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 pizza
```

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

Then execute

```
cd pizza
```

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

```
code pizza.py
```

to make a file called `pizza.py` where you'll write your program. Be sure to run

```
wget https://cs50.harvard.edu/python/2022/psets/6/pizza/sicilian.csv
```

to download [sicilian.csv](https://cs50.harvard.edu/python/2022/psets/6/pizza/sicilian.csv) into your folder. Also run

```
wget https://cs50.harvard.edu/python/2022/psets/6/pizza/regular.csv
```

to download [regular.csv](https://cs50.harvard.edu/python/2022/psets/6/pizza/regular.csv) into your folder.

How to Test

Here's how to test your code manually:

- Run your program with `python pizza.py`. Your program should exit using `sys.exit` and provide an error message:

```
Too few command-line arguments
```

- Be sure to download [regular.csv](https://cs50.harvard.edu/python/2022/psets/6/pizza/regular.csv) and [sicilian.csv](https://cs50.harvard.edu/python/2022/psets/6/pizza/sicilian.csv), placing them in the same folder as `pizza.py`. Run your program with `python pizza.py regular.csv sicilian.csv`. Your program should output:

```
Too many command-line arguments
```

- Run your program with `python pizza.py invalid_file.csv`. Assuming `invalid_file.csv` doesn't exist, your program should exit using `sys.exit` and provide an error message:

```
File does not exist
```

- Create a file named `sicilian.txt`. Run your program with `python pizza.py sicilian.txt`. Your program should exit using `sys.exit` and provide an error message:

```
Not a CSV file
```

- Run your program with `python pizza.py regular.csv`. Assuming you've downloaded [regular.csv](#), your program should print a table like the below:

```
+-----+-----+-----+
| Regular Pizza | Small | Large |
+=====+=====+=====+
| Cheese       | $13.50 | $18.95 |
+-----+-----+-----+
| 1 topping    | $14.75 | $20.95 |
+-----+-----+-----+
| 2 toppings   | $15.95 | $22.95 |
+-----+-----+-----+
| 3 toppings   | $16.95 | $24.95 |
+-----+-----+-----+
| Special     | $18.50 | $26.95 |
+-----+-----+-----+
```

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/pizza
```

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/pizza
```

