

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>) 

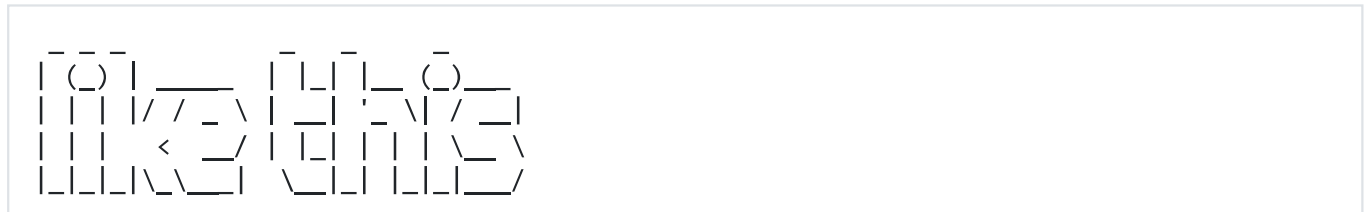
(<https://www.instagram.com/davidjmalan/>)  (<https://www.linkedin.com/in/malan/>) 

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

 (<https://twitter.com/davidjmalan>)

Frank, Ian and Glen's Letters

FIGlet (<https://en.wikipedia.org/wiki/FIGlet>), named after [Frank, Ian, and Glen's letters](http://www.figlet.org/faq.html) (<http://www.figlet.org/faq.html>), is a program from the early 1990s for making large letters out of ordinary text, a form of [ASCII art](https://en.wikipedia.org/wiki/ASCII_art) (https://en.wikipedia.org/wiki/ASCII_art):



Among the fonts supported by FIGlet are those at [figlet.org/examples.html](http://www.figlet.org/examples.html) (<http://www.figlet.org/examples.html>).

FIGlet has since been ported to Python as a module called [pyfiglet](https://pypi.org/project/pyfiglet/) (<https://pypi.org/project/pyfiglet/>).

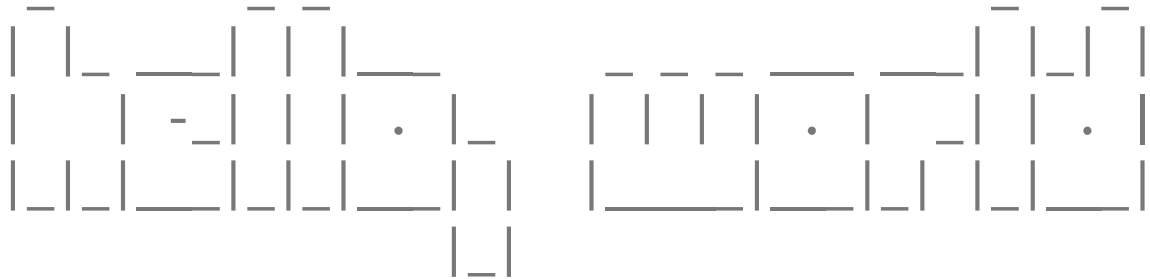
In a file called `figlet.py`, implement a program that:

- Expects zero or two command-line arguments:
 - Zero if the user would like to output text in a random font.
 - Two if the user would like to output text in a specific font, in which case the first of the two should be `-f` or `--font`, and the second of the two should be the name of the font.

- If the user provides two command-line arguments and the first is not `-f` or `--font` or the second is not the name of a font, the program should exit via `sys.exit` with an error message.

Demo

```
$ python figlet.py --font rectangles
Input: hello, world
Output:
```



```
$
```

Before You Begin

<https://cs50.harvard.edu/python/psets/4/figlet/>

```
$
```

Next execute

```
mkdir figlet
```

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

Then execute

```
cd figlet
```

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

```
code figlet.py
```

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

How to Test

Here's how to test your code manually:

- Run your program with `python figlet.py test`. Your program should exit via `sys.exit` and print an error message:

```
Invalid usage
```

- Run your program with `python figlet.py -a slant`. Your program should exit via `sys.exit` and print an error message:

```
Invalid usage
```

- Run your program with `python figlet.py -f invalid_font`. Your program should exit via `sys.exit` and print an error message:

```
Invalid usage
```

- Run your program with `python figlet.py -f slant`. Type `CS50`. Your program should print the following:

```
 /  _  /  _  /  _  \
 /  /  \  \  \  \  /  /

```

```

/ / _ _ / / _ /
\ / / _ _ \ / _ /

```

- Run your program with `python figlet.py -f rectangles`. Type `Hello, world`. Your program should print the following:

```

  _ _ _ _ _ _ _ _ _ _
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |
 | | | | | | | | | |

```

- Run your program with `python figlet.py -f alphabet`. Type `Moo`. Your program should print the following:

```

M   M
MM MM
M M M ooo ooo
M   M o o o o
M   M ooo ooo

```

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/figlet
```

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/figlet
```

