

day37 coded drawing  
Due: Friday 11/17/23

For this project you will start with code that I give you.

The program uses line of numbers and characters to draw a picture. Your task today is not to write any Python code, but instead to write a series of codes (numbers and letters) that will draw a picture.

Here is a sample drawing code:

```
art = "5.1x004.3x003.5x004.3x005.1x0000"
```

The way the code works is this: each drawing command is a single digit followed by the character you want to draw. A period ( . ) is used to indicate a space. In the above string, the first two characters "5." form a pair: a 5, then a space. This tells the program to print 5 spaces. The next two characters are "1x", which tells the program to print one x. The next two characters "00" tell the program to go to a new line; this wouldn't be obvious, I just set it up that way, two zeros is how you tell the program to print a new line. This string all together prints the following drawing when I call the function "DrawThing(art)".

```
  x
 xxx
xxxxx
 xxx
  x
```

Here are all the pairs in that code sequence broken apart, where . means a space:

5.	1x	00	4.	3x	00	3.	5x	00	4.	3x	00	5.	1x	00	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

So moving from left to right, it tells the drawing program to print 5 spaces, then 1 x, then a new line, then 4 spaces, then 3 xs, a new line, and so on.

The repl I give you in the starter file creates the following drawing:

```
xxxxxx
 x      x
x  x  x  x
x    x  x
x  (  )  x
x  ---  x
xxxxxx
```

I was trying for a happy face, I know it's not perfect. It looks more like a bear to be honest. But, it's something.

Your task today: replace the contents of the art variable in my starter code with your own drawing. You must create something entirely different from what I have shown you here and in the starter file.

(continued on next page)

To make the face, I typed the following into my repl program (this is not part of the code; it is just a template to help me visualize what codes I would use to make this picture.)

```
"""
.....xxxxxx
....x.....x
...x..x...x..x
..x.....x.....x
..x..(..)..x
...x..---..x
.....xxxxxx
"""
```

This might seem really complicated, but it's not. First, the triple quotes at the top and bottom make this whole thing a "comment", meaning repl won't try to run it like Python code. Then, I used a period (.) everywhere that I wanted a space, so I could easily count how many spaces were here and there. Then, when I had the drawing like I wanted it, I worked on coming up with drawing codes that would make it. Look at the first line:

```
.....xxxxxx
```

This is six spaces and five xs. So this line becomes "6.5x00". That tells the program to print six spaces, then five xs, then go to a new line (the 00 means go to a new line). The next line of the face:

```
....x.....x
```

is encoded as "4.1x7.1x00", that's 4 spaces, 1 x, 7 spaces, 1 x, then a new line. And so on.

Make your own creation. Create it in Python in between the triple quotes, then type out the codes that would print it. You don't have to go crazy, just make something fun.

You will be changing the part between the set of triple quotes near the top and you will put your coded drawing instructions into the art variable, just above that.

### A few last important notes:

1. You can't print the number zero. In other words, if you want to print 5 zeroes, you need to use 5 capital Os.
2. If you want to print a backslash, put two of them in your code. So to print "/" use "1/1o1\\". I know this seems wrong, but backslashes are special; just backslashes require that you put two of them in.
3. You can't use the quote character " or an actual period in your drawing, sorry.
4. The number can only be a single digit. So if you want more than 9 of something you have to break it up into parts. For example:

To draw 15 spaces, do this: 8x7x or 9x6x don't try to use 15x

**The program doesn't care, as long as you don't use a two digit number.** Every pair in the sequence must be a single digit followed by the character you want to print that many times.

Get started [here](#).