For today, we’d be creating a basic Mad Libs game. According to Wikipedia, Mad Libs is a phrasal template word game which consists of one player prompting others for a list of words to substitute for blanks in a story before reading it aloud. It is usually played as a party game.

**How to play**

Mad Libs can be played with one or more persons. Whoever is in charge of the story (in our case the program) prompts others for a list of whatever is needed to complete the blanks in the story and after everything is supplied, the person in charger reads the completed story back to the user.

**Game Strategy**

In order to replicate this, we are going to make use of string concatenation, input and print functions. The function of the input is to prompt the user to enter something which would then be saved as a string in the variable assigned to it. The function of the print is to well return the outcome.

Now there are three ways we can do this string concatenation.

* Using the plus (+) sign
* Using curly braces and then calling string.format()
* Using the f string. F string is a way of formatting strings to make it more readable. Basically, just add a curly brace ({}) and insert the variable name in between the curly braces and then before the string add an f eh <code>matlib = f”this is a {nstring}”</code> nstring above is a variable name.

From the code snippet above, we can see that they three methods all return the same output (line 6) but for the purpose of this program, we are going to be using the f string.

The first thing to do in the game of mad libs is to prompt the other participants of the game to provide details (e.g. verbs, nouns etc.) needed to fill in the blanks. To do this, we are going to make use of the input function. (Insert snippet)

After the user inputs the different details required, they are stored in memory with different variable names as seen above.

The next step is to concatenate the string so that the variables complete our story as in the snippet below. (Insert snippet)

The next step is to output out our mad lib variable so we can see the completed story. To do this, we would make use of the print function like so. (snippet)

If you are interested, I made a random mad lib that contains a list of 4 mad libs and then randomly picks one and calls the “.madlib” method on the file it picks. (Insert snippet). This can be accessed <a href=”https://github.com/thedumebi/100-days-express-blog/tree/master/100%20days/scripts/python”>here</a>. The code looks like this. (Insert snippet)

Random is a python module that can pick out a random item from a non-empty sequence. In our case we passed in a list containing the different mad libs imported into the python script. By default, when we create a python script, python automatically sets the \_\_name\_\_ to “\_\_main\_\_”. When we import other scripts into a python script, the \_\_name\_\_ of those scripts become set to the filename of that script for example, hello.py was imported above. In hello.py file itself, the \_\_name\_\_ is set to “\_\_main\_\_” but since it’s imported, the \_\_name\_\_ of that file in our script at the moment is set to “hello”. Anyways, the reason why that line of code is there is so that the code under would only be executed if it’s the main file and to prevent those imports from just running on their own.