**Formatting Methods:**

* Three methods that can change the casing of a string *.lower(), .upper(), .title()*
* *.lower()* – returns the string with all lowercase characters
* *.upper()*- returns the string with all uppercase characters
* *.title()* - returns the string in title case (first letter of each word is capitalized)

**Splitting Strings:**

* *.split()* – takes an argument, returns a list of substrings found between the given argument (known as a **delimiter)**
* If no argument is provided, it will default to splitting at spaces
* If we provide a n argument to split on (as a “string”) we can dictate what character the split occurs on – note that it doesn’t include the character chosen for split
* When you split a string with a character that it ends on you will end up with an empty “” at the endA screenshot of a computer screen

  Description automatically generated A computer screen shot of text

  Description automatically generated

**Escape Sequences:**

* Used to indicate that we want to split something in a string that is not necessarily a character
* *\n – Newline* – Allows us to break a multi-string line by line breaks
* *\t – Horizontal Tab* - Allows us to break a multi-string line by tabs  
  A screenshot of a computer program

  Description automatically generated A screenshot of a computer screen

  Description automatically generated

**Joining Strings:**

* *.join()* – Used to join a list of strings back together using a **delimiter**
* Syntax is pretty much backwards from *.split()*
* Can join together using “ , “ (**comma separated values – csv**)
* Or escape characters (\n, \t) **A screenshot of a computer

  Description automatically generated**

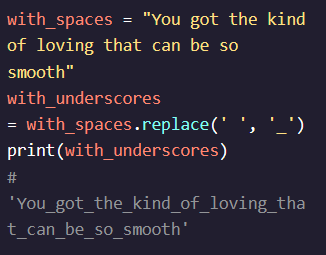
**Cleaning Strings:**

* *.strip()* – Used to remove unnecessary whitespace, line breaks, tabs, and other characters from strings
* Using *.strip()* by itself with no argument strips all whitespace characters from beginning and end of string
* Can also be used with a character argument to strip that character from either end of the string (only strips that character NOT also whitespace)  
  A computer screen shot of a computer code

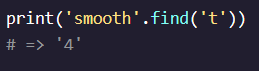
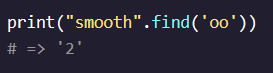
  Description automatically generated A computer screen shot of text

  Description automatically generated

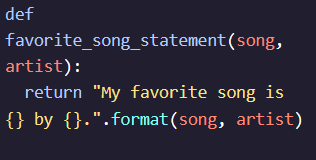
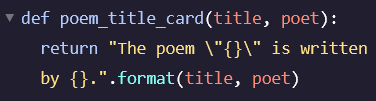
**Replacing Strings:**

- *.replace()* – Takes two arguments and is used to replace all instances of first argument with second argument (basically CTRL + H)  


**Finding Strings:**

- *.find()* – Takes a string as an argument and searches string it was ran on for that argument in it (CTRL + F)  
- Returns answer as *index* spot where that argument is found  
- Can also search for larger strings and it will return *index* position of the first character of that string  
 

**Formatting Strings:**

*- .format()* – Used to insert variables into an argument and create a new string with those variables included  
- Can take as many arguments are there are {} in the string it is run on  
- Can accomplish the same thing using string concatenation (+) but using this makes code more readable and also allows you to reuse same base string with different variables  
 

- Can also assign variables as keywords to negate necessity of inputting arguments in correct order  
- Can reverse order of keywords and it won’t matter  
