

Exercise 9 - Strings

1. Using the string methods `rjust` and `center` shown here to create the following output. The program should read in word from the user (3 times) and print out the word formatted as shown. Save your program as `formatWord.py`.

When done, experiment with the fill character option.

```
str.center(width[, fillchar])
```

Return centered in a string of length `width`. Padding is done using the specified `fillchar` (default is a space).

```
str.rjust(width[, fillchar])
```

Return the string right justified in a string of length `width`. Padding is done using the specified `fillchar` (default is a space). The original string is returned if `width` is less than `len(s)`.

Changed in version 2.4: Support for the `fillchar` argument.

```
<terminated> /Users/aaronkilpatrick/Documents/Python/ICS3U-V
Please enter a wordgorilla
*****
gorilla
gorilla

Please enter a wordmonkey
*****
monkey
monkey

Please enter a wordelephant
*****
elephant
elephant
```

2. Read a word in from the user and print out the middle letter. If the word has an odd number of letters, print the middle letter. If the word has an even number of letters, print the middle two letters. Save your program as `middleLetter.py`.

3. Remove the commas from a number entered by the user. Save your program as `removeCommas.py`.

Example/

Enter a number in the format 12,123: 123,123,234

Your number is: 123123234

4. Write a method that reads in two words from the user and prints them out in alphabetic order. Save your program as `alphaOrder.py`.
5. Write a test program to figure out what the method `str.rfind(sub [,start [, end]])` does. Save your program as `strrFind.py`.
6. Write a program that accepts a word from the user. Secondly, use the method `str.title()` to change the word and print it out. Lastly, use the method `str.istitle()` to check if the word starts with a capital, print out YES if it does, NO if it doesn't. Save your program as `wordTitleCapital.py`.

Explain the difference between the methods `str.title()` and `str.istitle()`.

7. Write a program that reads a word from the user and creates a new word that has a space between every letter. Print out the new word. Save your program as `wordSpace.py`.

Example/

Please enter a word: elephant

e l e p h a n t

8. Write a method that returns a string reversed. The method should accept the word as a parameter and then return the reverse image of it. The main should ask the user for the word and then print out the reverse. Save your program as `stringReversed.py`.

Example/

Please enter a word: gorilla

You word reversed is: allirog

9. Write a method that determines if a string is a palindrome. The means that the word and its reverse are equal. The method should return True if the word is a palindrome and False if not. Save your program as `palindrome.py`.

Example/

Please enter a word to check: radar

Yes, radar is a palindrome

Please enter a word to check: yellow

No, yellow is not a palindrome

10. Modify the following code so that Ouack and Quack are spelled correctly. Save your program as `correctSpelling.py`.

```

27 prefixes = "JKLMNOPQ"
28 suffix = "ack"
29
30 for letter in prefixes:
31     print letter + suffix

```

11. Create a method called `countLetters` using the code below. The method should count the number of times a particular letter appears in the word. For example, the letter 'a' appears 3 times in the word 'banana'. Save your program as `countLetters.py`.

The method should **accept** parameters for the string and the letter.
The method should **return** the number of times the letter appears.

```

35 fruit = "banana"
36 count = 0
37 for char in fruit:
38     if char == 'a':
39         count += 1
40 print count
41

```

12. Write a method that counts the vowels in a word or sentence. It should return the number of vowels it found. Save your program as `countVowels.py`.

Example/

Please enter a word or sentence: Hello world!

There are 3 vowels

Please enter a word or sentence: supercalifragilisticexpialidocious

There are 16 vowels

13. Read a word in from the user, remove the middle letter or middle two letters and put them at the end of the word. Save your program as `removingMiddleLetters.py`.

Example/

Enter a word to encode: help

Coded word: hpel

Enter a word to encode: thank

Coded word: thnka