*Start a new Python project folder … do not reuse the previous workspace … As with any keyboard-driven console-like environment, developing muscle -memory for the common commands is also part of the learning curve.*

Other examples of regex methods

re.search()

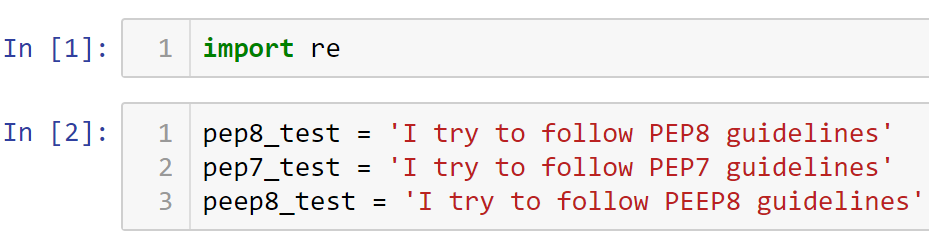
re.match()

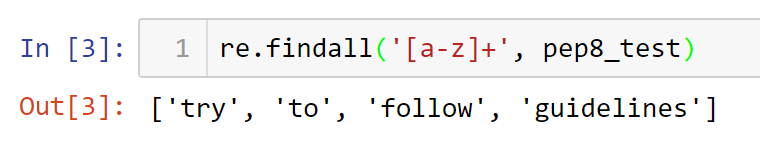
re.fullmatch()

re.finditer()

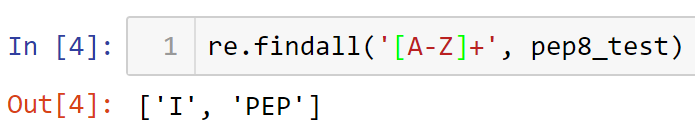
PEP8 is a style guide for Python Code. Words maybe misspelled such as PEP7 or PEEP8.

So you want to build a process that when you replace PEP8 the process should also include the misspelled words.

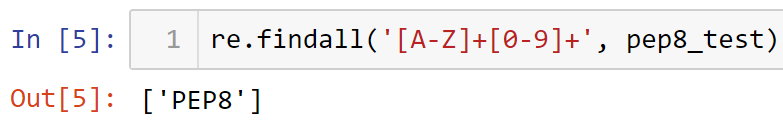




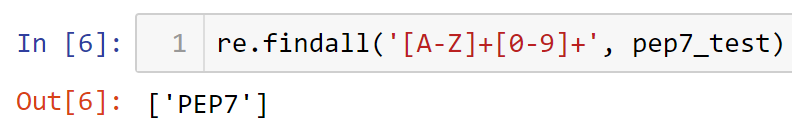
It did not find ‘I’ because it is capital letter; not ‘PEP8’ because it has a number. Regex is case sensitive.



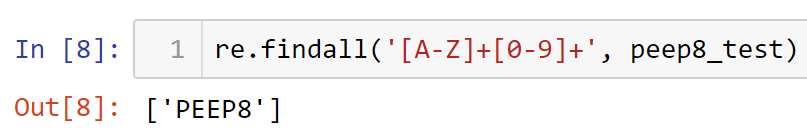
Returns only capital ‘I’ and ‘PEP’ without the 8. For PEP8, we want to capture letters AND numbers. The expression ‘[A-Z0-9]+’ will only capture letters OR numbers. To make this regex an AND, put a bracket around A-Z.



Let’s test the regex on the other strings …



It found ‘PEP7’

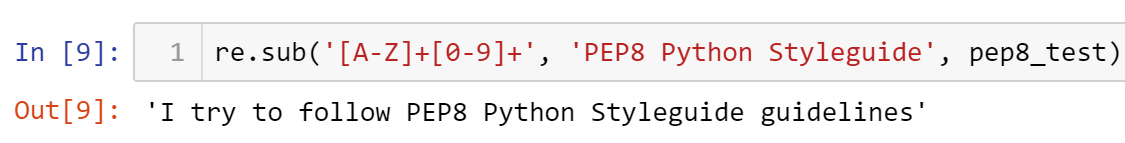


It captured ‘PEEP8’

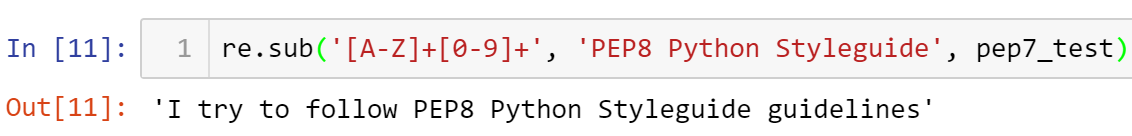
Therefore, this regex captured the real ‘PEP8’, and the most likely misspellings.

**Replacing a specific string**

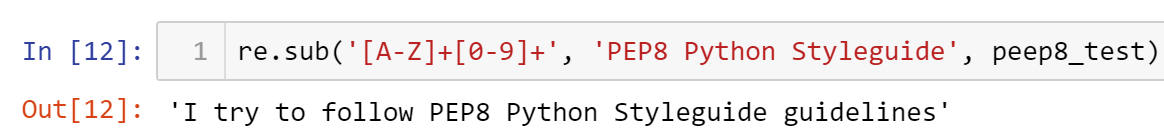
Now that we know how to capture the real ‘PEP8’ and its misspellings, it is time to replace all its instances. The code below means that when you find the word captured by the regex pattern we just built, replace that word with ‘PEP8 Python Styleguide’. The function we will use is sub*string*.



Testing the code against a misspelling pep7\_test ...



Lastly, we will test it on the misspelling in peep8\_test example …



The regex code also worked with the third test string.

Note that the regex is not perfect because it will not work with lowercases or with space embedded on the string. You can continue to refine your regex to work on other use cases. The best way to capture all the other scenarios is to do a quick test like what you just did on the early part of this exercise.

Your turn: Expand the regex above so it will capture letters that also are in lowercase such as Pep8, pep8, pEP8, peP8, etc.

Place your answer and its associated output below:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* All submissions should be separate from other exercises and quests. Please do not lump all your answers into one document and re-using that same workspace to gain multiple points. Thanks.
* Place your name at the bottom of your code, download your Python program in html format, and submit your work in Canvas.