Practice - Analysing Text Files

Last lecture we learnt how to read and write text files. This lecture we will analyse the properties of the text within in it.

The book we will analyse is "A Study in Scarlet" by Arthur Conan Doyle. Published in 1887 this was the first outing of the detective duo Watson & Holmes.

We will use the Project Gutenberg transcription of the book which has the filename `244-0.txt` .

Getting the book "A Study in Scarlet"

The transcription is available at no cost from Project Gutenberg. Download it and save it to your folder. It should have the name `244-0.txt` .

Book files from Project Gutenberg begin with a 'header' at the beginning of the file which contains information about the contents, when it was transcribed, and so on. It also has a 'footer' at the end of the file with information about licensing, Project Gutenberg itself, and making a donation. Essential though that header and footer information is, it is not part of the book and therefore we need to remove it.

The header ends with the line: *** START OF THIS PROJECT GUTENBERG EBOOK A STUDY IN SCARLET ***
The footer begins with the line: *** END OF THIS PROJECT GUTENBERG EBOOK A STUDY IN SCARLET ***

Read book and re-write without header and footer

The task is to read the book and write it out again, but without the header and footer. We can create a function to do this. The function parameter will

be the existing filename and the new file name.

```
def remove_hf(oldfilename, newfilename):
    """read the file from Project Gutenberg and remove the header and f
    contain the book"""
    with open(oldfilename, 'r', encoding='utf-8') as oldtext:
        # read the whole file into a string
        g_book = oldtext.read()
        # split string into three parts
        parts123 = g_book.partition('*** START OF THIS PROJECT GUTENBEF
        # and we only want the part after that line
        g_book = parts123[2]
        # partition it again
        parts123 = g_book.partition('*** END OF THIS PROJECT GUTENBERG
        # and we only want the part before that line
        g_book = parts123[0]
    # now write just the book out to the new file
    with open(newfilename, 'w', encoding='utf-8') as newtext:
        # the book is in the variable g_book, so write all of that
        newtext.write(g_book)
    #  all done, so just return
    return
```

Does a letter character occur in the book

The task is to read the book, search it for the occurrence of a specified character. If it is present return `True`, else return `False`.

```
def contains(a_str, filename):
    # Open the file and read it into a string
    with open(filename, 'r', encoding='utf-8') as text:
        # test if a_str is in text
        if a_str in text:
            return True
        else:
            return False
```

Get the set of all the characters that occur anywhere in the text

Read the book into string (the default) and then make a set from that string

```
def all_chars(filename):
    # Open the file and read it into a string
    with open(filename, 'r', encoding='utf-8') as text:
        # test if a_str is in text
        if a_str in text:
            return True
        else:
            return False
```