

## This assignment: concord2.py

Assignment #1 introduced you to the concept of “keyword in context”. In this assignment we will introduce some output formatting that should help make output more readable.

Consider the following input – similar to that from in07.txt in assignment #, but with significantly different output.

```
2
    ' ' ' '
of
and
the
too
on
who
to
that
' ' ' '
that fortune
sense and sensibility
life of robert browning
the man who knew too much
legend of montrose
visit to iceland
orthodoxy
the mountains
on the track
ward of king canute
```

```

    life of robert BROWNING
      ward of king CANUTE
        that FORTUNE
          visit to ICELAND
            ward of KING canute
              the man who KNEW too much
                LEGEND of montrose
                  LIFE of robert browning
                    the MAN who knew too much
                      legend of MONTROSE
                        the MOUNTAINS
                          man who knew too MUCH
                            ORTHODOXY
                              life of ROBERT browning
                                SENSE and sensibility
                                  sense and SENSIBILITY
                                    on the TRACK
                                      VISIT to iceland
                                        WARD of king canute
```

The biggest difference is that *index words are now left-aligned*, and some of the text from the surrounding words also appear. However, there are a few more differences, too – one of which is clear from the example above, another related one which can be seen in tests sets 09 and higher.

- Index words now begin at the 30th column of the output
- Words to the left of the index word may appear but only if they do not go *further left than column 10*.
- Words to the right of the index word may appear but only if they do not go *further right than column 60*.

To illustrate further, consider one of the lines taken from `in09.txt`:

```
verzierten sofa dessen polster hellgelb ueberzogen waren warf einen
```

And below are extracted from `out09.txt` the lines generated for this input line. (The numbers are the top are to indicate the column numbering. These must not appear in the output.)

```
00000000011111111122222222223333333333334444444444555555555566666666667
1234567890123456789012345678901234567890123456789012345678901234567890
      sofa dessen polster HELLGELB ueberzogen waren warf
            sofa dessen POLSTER hellgelb ueberzogen
            verzierten SOFA dessen polster hellgelb
      polster hellgelb UEBERZOGEN waren warf einen
            VERZIERTEN sofa dessen polster
hellgelb ueberzogen WAREN warf einen
      ueberzogen waren WARF einen
```

In the first line of output above, notice that “verzierten” does not appear even though it is in the input line. (i.e., it would go too far left in the output). Also in the first line of output above, notice that the word “einen” is also missing (i.e., it would go too far to the right). In the second line of output, “verzierten” is missing from the left, and “waren warf” are missing from the right. And so on and so forth.

Your implementation of `concord2.py` must produce this kind of formatting of the indexed-output lines.

And in addition:

- Lines-for-input may now contain words in different cases. However, for the purposes of indexing you can ignore case (i.e., when comparing words, first convert them into all upper-case or all lower-case). If the word in a line is not an index word, however, it must appear in its capitalization (or lack thereof).
- There is no limit to the number of exclusion words, input lines, or index words.
- Lines-for-indexing can contain blank lines. These blank lines may be ignored.