CSC108H Worksheet: Random Story Generation 3: Working with the Data

You have worked out that it makes sense to represent the words in the story as a list of strings. You'll need to read the words in the file and create that list. This string method will help:

```
>>> help(str.split)
Help on method_descriptor:

split(...)
    S.split(sep=None, maxsplit=-1) -> list of strings

    Return a list of the words in S, using sep as the
    delimiter string.  If maxsplit is given, at most maxsplit
    splits are done. If sep is not specified or is None, any
    whitespace string is a separator and empty strings are
    removed from the result.
```

Read the last sentence of that description carefully.

If a variable `training_file` is a file open for reading, then `training_file.read()` returns a string containing the entire contents of the file. You can use method `split` to make your list.

Write some Python statements to do this, assigning the list to variable `story`:

```
training_file = open('my_story.txt', 'r')
```

Here is an example of a story list:

```
['And', 'the', 'fan,', 'and', 'the', 'cup,', 'And', 'the', 'ship,', 'and', 'the', 'fish.']
```

You have worked out that it makes sense to represent the context information as a dictionary where the keys are tuples of strings and the values are lists of strings. For the story above, draw the dictionary with a context length of 2. We have given you the first key below; write the list of values for it.

Fill in the dictionary with all the 2-word contexts and their values.

Reminder: Python is case sensitive, so `'And'` and `'and'` are not equal.

```
{ ('And', 'the') : [
```

```
}
```