School of Languages, Linguistics and Film Assessed Coursework Coversheet

For undergraduate (BA) modules coded:
CAT-, COM-, EAL-, FLM-, FRE-, GER-, HSP-, LAN-, LIN-, LLU-, POR-, RUS-, SML-

Please read and note the following guidelines:

1. To assist with anonymous marking, please use your nine-digit student ID number only: do **NOT** use your name anywhere on your coursework.
2. Normally you will be required to submit one electronic copy of coursework via the module's QMplus area. Most deadlines in this School are set for a Sunday night (23:55). You will be informed by the module organiser of any exceptions to this procedure, either regarding the time or method of submission. It is your responsibility to ensure that you know and meet the submission requirements for each piece of coursework.
3. You must keep a copy of all coursework you have submitted.

4. Extensions to deadlines may ONLY be granted by the Senior Tutor for your department. In order to be granted an extension, you must submit a claim for Extenuating Circumstances BEFORE the coursework deadline. SLLF has an online EC claim form. Details and links to the form can be found here: . http://sllf.qmul.ac.uk/extenuating-circumstances/

5. Late submission, without an agreed extension due to extenuating circumstances, will be penalised according to the SLLF regulations relevant to your level of study.
6. Work submitted within 5 DAYS of the deadline will be accepted but subject to a late submission penalty against the marks awarded. The work will be marked normally, and then a late submission penalty of five marks (or 5% of the marks if not marked out of 100) per 24 hour period will then be applied.
7. Work that is more than 5 DAYS late will not be accepted and will not be marked and will receive a mark of ZERO.

You are reminded that plagiarism, that is copying someone else's words or ideas without attributing them to that person, is cheating. This is a serious examination offence and at the very least will result in a mark of zero being awarded for this piece of work; it could result in your expulsion from Queen Mary.

By handing in this coursework you acknowledge that it represents your own, unaided work and that you have appropriately acknowledged all sources.

Please complete the following details:

Student ID Number:(9-digit number): 190176444
Module CODE and TITLE: LIN6209 Coding for Linguists
Title of Coursework:
Essay no:
Number of words written: 306
Module Organiser: Peter McGinty
Seminar Tutor (if applicable):

Please continue your coursework on the next page

LIN6209 CODING PROJECT PROPOSAL

Most likely next word coding function

This Python coding project will aim to produce an output that returns the next most likely word following the input string, which will be a single word. An example purpose of the project is to find common reporting words used in media in regards to political speakers; for instance, if the input string was ‘Johnson’, the output may return ‘claims’.As such, text data can be freely accessed and used from news websites, such as the British Broadcasting Corporation, in order to test the code.

The basic functionality of the code will include an all-encompassing function that can be defined as something similar to ‘next_word(word1)’, with ‘word1’ being a single-word string. The function would read the text file contents into a single string, and then be split into lines so that each word is on its own line. The function would then look for ‘word1’ within the file and, upon identifying it, continue onto the following line and adding its contents to a dictionary with a value of +=1 depending on whether the dictionary already contains that string or not. When all instances of ‘word1’ have been identified and the above steps completed, the key in the dictionary with the highest value would be instructed to be returned, thus ‘predicting’ the most likely next word following ‘word1’.

As specified above, the code can be tested by using text files from the BBC and keywords such as names in order to see whether the correct string would be returned by manually reading through the article and comparing the result to the code’s output. Additionally, the assert() function can be used in Python to directly find if the output of the function or value types matches what is expected. As such, the software that will be used to both construct and test this coding project will be Python IDLE.