Lab 03

1. A palindrome is a word that is spelled the same backward and forward, like “noon”  
   and “redivider”. Recursively, a word is a palindrome if the first and last letters are the same and the  
   middle is a palindrome.  
   The following are functions that take a string argument and return the first, last, and middle letters:  
    def first(word):  
    return word[0]  
    def last(word):  
    return word[-1]  
    def middle(word):  
    return word[1:-1]

1. Type these functions into a file named palindrome.py and test them out. What happens if  
you call middle with a string with two letters? One letter? What about the empty string,  
which is written '' and contains no letters?  
2. Write a function called is\_palindrome that takes a string argument and returns True if it  
is a palindrome and False otherwise. Remember that you can use the built-in function len  
to check the length of a string

1. A string slice can take a third index that specifies the “step size”; that is, the number  
   of spaces between successive characters. A step size of 2 means every other character; 3 means every  
   third, etc.  
   >>> fruit = 'banana'  
   >>> fruit[0:5:2]  
   'bnn'

A step size of -1 goes through the word backwards, so the slice [::-1] generates a reversed string.  
Use the slice in this manner to write a **one-line version** of is\_palindrome from question 1.

Save this new version as q2.py

1. Word histogram

You need to write a script that will create a histogram of words in a large text file. The script will need to do the following:

-Read in a large text file using file IO handle.

-Create an empty dictionary

-Split each line into individual words, and iterate through them all to track how many times each word occurs in the text. Using the individual word as a key, create or update that word’s value in the dictionary.

Use the provided script character\_histogram.py as a starting point. Be sure to save it as a separate file named word\_histogram.py and make that file work on words rather than characters.

**At the end of your script, print out the # of unique words found in the textfile.**

1. Your super bowl score prediction