**Exercise 7**

1. Copy class Passage to your source file. Modify it as needed

**class Passage**

{

private HashMap<String, Integer> HM; // all unique words & counts

private int totalCount;

public readPassage()

{

// Read the passage from text file. Check each word

// (1) If it doesn’t exist in HM, put it in HM

// (2) If it already exists in HM, update its count

// (3) Calculate total count (of all words)

}

public HashMap<String, Integer> getWordsAndCounts() { return HM; }

public HashSet<String> getWords()

{

HashSet<String> keys = new HashSet<String>(HM.keySet());

return keys;

}

}

1. Copy class SetOperations to your source file. Modify it as needed

This class acts as a utility class, so its methods are static

**class SetOperations**

{

public static HashSet<String> intersect(HashSet<String> H1, HashSet<String> H2)

{

// find words exist in both H1 and H2

}

public static HashSet<String> difference(HashSet<String> H1, HashSet<String> H2)

{

// find words exist in H1 but not in H2

}

}

1. Write a main class to do the following:

* Create 2 Passage Objects & call method readPassage() of each object
* Print total number of words & total number of unique words in each passage
* Use methods in SetOperations to find words (and counts) existing only in each passage, and those existing in both

Hint

- Convert all words to lowercase or Uppercase, so that the and The are considered

the same word

- When reading each word, check & strip punctuation (. , ; : ! ?)





