Lab 4 B - WordWhiz

Due No Due Date **Points** None

In this lab, you will learn to use abstract classes and apply the principle of polymorphism.

Problem statement: There are two data files, one with words and their synonyms (Fig.1: Thesaurus.txt)), and the other with words and their meanings (Fig.3: Dictionary.txt). You need to create a program named **WordWhiz** that takes in a word as input, and then displays its synonym or meaning depending on the menu option chosen. The program is case insensitive.

Fig.1: Thesaurus.txt: It is a csv file with varying number of words in each row. WordWhiz displays all the words from the row in which the user's word is the first word as shown in Fig. 2.

```
above, higher up, in a higher place, to a higher place
abrupt, precipitous, sharp
accept, take, have
acceptance, credence
accident source, cause of the accident
accordion, squeeze box, piano accordion
account, business relationship
```

Fig.2: Menu option 1

```
*** Welcome to WordWhiz ***
1. Look up Thesaurus
2. Look up Dictionary
Enter your choice
1
Enter word
abrupt
1.  precipitous
2.  sharp
```

Fig.3: Dictionary.txt: A row in has the format: "*word (…) meaning*". You need to extract the word before "(" and its meaning after ")" from each row of data. A word may have multiple meanings and hence multiple rows. WordWhiz is supposed to display all meanings of a word as shown in Fig.4.

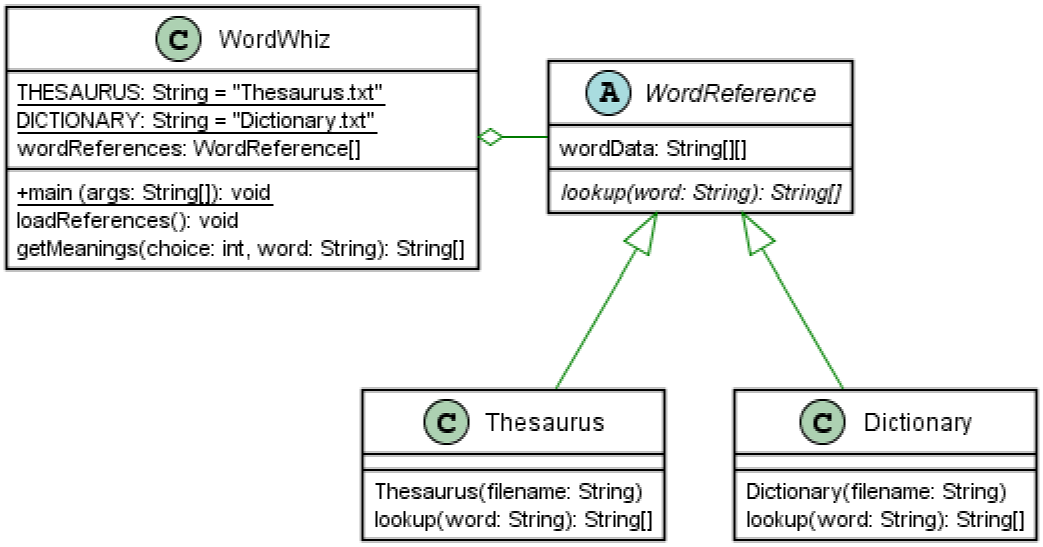
```
Aback (adv.) Toward the back or rear; backward.
Aback (adv.) Behind; in the rear.
Aback (adv.) Backward against the mast; -- said of the sails when pressed by the wind.
Aback (n.) An abacus.
Abactinal (a.) Pertaining to the surface or end opposite to the mouth in a radiate animal; -- opposed to actinal.
```

```
Abaction (n.) Stealing cattle on a large scale.
Abactor (n.) One who steals and drives away cattle or beasts by herds or droves.
```

Fig.4: Menu option 2

```
*** Welcome to WordWhiz ***
1. Look up Thesaurus
2. Look up Dictionary
Enter your choice
2
Enter word
aback
1. Toward the back or rear; backward.
2. Behind; in the rear.
3. Backward against the mast; -- said of the sails when pressed by the wind.
4. An abacus.
```

Solution Design:


C WordWhiz
THESAURUS: String = "Thesaurus.txt"
DICTIONARY: String = "Dictionary.txt"
wordReferences: WordReference[]
+main (args: String[]): void
loadReferences(): void
getMeanings(choice: int, word: String): String[]
A WordReference
wordData: String[][]
lookup(word: String): String[]
C Thesaurus
Thesaurus(filename: String)
lookup(word: String): String[]
C Dictionary
Dictionary(filename: String)
lookup(word: String): String[]

The main method in WordWhiz is fully coded. You need to code the other two methods. Refer to the comments in the code provided.

You also need to create WordReference, Thesaurus, and Dictionary classes.

Thesaurus

- The **constructor** reads each row from the file name[1] passed to it, splits it on comma, and stores the words in wordData[][] array. Each row has different number of words, as shown in the table below.
- *The **lookup**() returns an array of synonyms for the word passed to it. If the word is not found, it returns null.*

	0	1	2

wordData[i]	abrupt	precipitous	sharp
wordData[j]	accept	take	have
wordData[k]	acceptance	credence	
...and so on			

Dictionary

- The **constructor** reads each row from the file name[1] passed to it and extracts the first word as the first element in the wordData[][] array and its meaning in the second element, as shown below. *(Hint: use regex "[(]" to extract the word and "[)]" to extract its meaning from each row of data)*
- *The **lookup**() returns an array of meanings for the word passed to it. If the word is not found, it returns null.*

A row in Dictionary.txt	wordData[i][0]	wordData[i][1]
Abate (v. t.) To beat down; to overthrow.	Abate	To beat down; to overthrow.
Abate (v. t.) To bring down or reduce from a higher to a lower state, number, or degree; to lessen; to diminish; to contract; to moderate; to cut short; as, to abate a demand; to abate pride, zeal, hope.	Abate	To bring down or reduce from a higher to a lower state, number, or degree; to lessen; to diminish; to contract; to moderate; to cut short; as, to abate a demand; to abate pride, zeal, hope.
... and so on		

[1] **Do not hard code file name in the constructors**. Use the filename passed to it as parameter.

Design Constraint:

- Use of Collection classes is not allowed in this lab.
- Each class must be created in its own .java file

Instructions

1. Download
   - **WordWhiz.java** ↓ **(https://canvas.cmu.edu/courses/25253/files/6728731/download?download_frd=1)**
   - **TestDictionary.java** ↓ **(https://canvas.cmu.edu/courses/25253/files/6728712/download?download_frd=1)**
   - **TestThesaurus.java** ↓ **(https://canvas.cmu.edu/courses/25253/files/6728730/download?download_frd=1)**
   - **Thesaurus.txt** ↓ **(https://canvas.cmu.edu/courses/25253/files/6728711/download?download_frd=1)**
   - **Dictionary.txt** ↓ **(https://canvas.cmu.edu/courses/25253/files/6728710/download?download_frd=1)**
2. Copy/import java files in package named lab4 and txt files in the project folder.
3. Complete the program as needed
4. Write your name and Andrew id in all your code files as comments at the top.
5. Zip WordWhiz.java, WordReference.java, Thesaurus.java, and Dictionary.java in a zip named **AndrewId-lab4.zip**. Submit the zip file.

Rubric:

- Dictionary: 4 points
- Thesaurus: 4 points
- Correct use of polymorphism, abstract class, static/final variables etc.: 2 points
- Any submission issue may cost you up to 2 points.