Chapter 12 Programming Project (100 points)

Read Chapter 12 in your book if you have not done so yet.

Write a hangman game that randomly generates a word (The program reads the words stored in a text file named **hangman.txt**. Words are delimited by spaces or each on their own line.) and prompts the user to guess one letter at a time, as presented in the sample run. Each letter in the word is displayed as an asterisk. When the user makes a correct guess, the actual letter is then displayed. When the user finishes a word, display the number of misses and ask the user whether to continue to play with another word. Declare an array to store words, as follows (where the words are inserted in the array from the file):

```
//Add any words you wish in this array

String[] words = {"write", "that",...};
```

```
(Guess) Enter a letter in word ******* > p Enter
(Guess) Enter a letter in word p****** > r Enter
(Guess) Enter a letter in word pr**r** > p Enter
     p is already in the word
(Guess) Enter a letter in word pr**r** > o Enter
(Guess) Enter a letter in word pro*r** > g Enter
(Guess) Enter a letter in word progr** > n Enter
     n is not in the word
(Guess) Enter a letter in word progr** > m Enter
(Guess) Enter a letter in word progr*m > a Enter
The word is program. You missed 1 time
Do you want to guess another word? Enter y or n>
```

You will need to create the **hangman.txt** file of words.

You should also include error checking that a letter is entered (not a number or special character).

Name your file Chapter12_*yourlastname*. Submit your **Java** file **AND** text file to D2L.

You will be graded on the following criteria:

- Complete comment block at beginning
- Comments throughout code
- Code compiles and runs (failure to compile and compile will be a automatic 50% deduction)
- Code works and produces desired output
- Variables are named by the standards previously provided
- Only concepts from chapters 1 through 10 and 12 of your book are used.