Assignment 2: Hangman Game

Assignment Overview

In this assignment, you will implement a variation of the classic Hangman game using **Python** and **Java** programming languages. The game, played in the console, will feature multiple categories from which players can choose. The words and categories will be loaded from an external text file. After every guess, the game will show the list of previously guessed letters. The goal of this assignment is to help you practice working with data structures such as strings and lists; and use selection and iteration control structures. Finally, you will compare the implementation between the two programming languages.

Requirements

1. Game Design:

- The game will present a hidden word with letters replaced by underscores (_).
- The player will guess one letter at a time. If the letter is in the word, it will be revealed in the correct positions. If the letter is not in the word, the player will lose one try.
- The player has a maximum of 6 incorrect guesses before he/she loses the game.
- The game will display the list of previously guessed letters after each guess.
- The game should load the categories and words from an external text file (words.txt) provided with the assignment manual.
- The file will contain several categories. Each category, ends with a colon (:), will be followed by words related to that category.
- Words should be lowercase.
- The format of the file is as follows:

```
Animals:
elephant
giraffe
kangaroo
tiger
panda

Movies:
inception
gladiator
matrix
avatar
jaws
```

2. Game Play:

- The player will be asked to select a category from the available categories.
- The game will choose a random word from the selected category.
- The player will make guesses by entering one letter at a time.
- After each guess, the game will:
  - Show the current state of the word (with guessed letters filled in).
  - Show the list of tried letters.
  - Show how many tries are remaining.
- The game will end when:
  - The player guesses the word (all the letters) correctly. The player is congratulated.
  - The player runs out of incorrect guesses (6 tries). The correct word is revealed.

3. Implementation steps:

1. Place the provided text file named words.txt in the same directory as your python script or java app. You may add additional categories (e.g., Animals, Movies, Countries) and a list of words under each category.
2. Write the game so that it:
   a. Loads categories and words from the words.txt file.
   b. Prompts the player to choose a category .
   c. Chooses a random word from the selected category.
   d. Displays the hidden word with underscores and allows the player to guess letters.
   e. Displays the list of previously guessed letters after each guess.
   f. Ends the game when the word is guessed or the maximum number of incorrect guesses is reached.
   g. Handles errors, such as invalid input (e.g., more than one character entered, non-alphabetical characters, a letter entered again, etc.).
3. Play your game to test it:
   a. Test your game with multiple categories and words.
   b. Ensure that incorrect guesses are counted and displayed properly.
   c. Ensure that the guessed letters are displayed correctly after each guess.

4. Rating your experience:

- Write 1-2 paragraphs on your overall experience of implementing the games in Python and Java languages (open ended; write whatever you want).
- Rate your coding experience using following table. Enter values from 1 to 5 where 1 being the lowest rating and 5 is the highest)

Programming Language	Development Speed	Ease of Testing and Debugging	Code Readability	# lines of code
Python	e.g., 5			
Java				

5. Notes:

1. Carefully read all the requirements before starting the implementation.
2. Compilation Errors: Any code that has compilation errors or doesn't run correctly will receive a grade of zero. Please test your code thoroughly to ensure that it runs without errors.
3. Make sure to test the game thoroughly, including handling edge cases like entering invalid characters.
4. Assume that the words.txt file is formatted correctly; each category name should end with a colon (:), followed by words on new lines. A different file of a similar format will be used to test your programs during marking.
5. Only use native and standard libraries of Python and Java. Do not use any third party libraries.
6. Your game's output should verbosely match the example output provided.
7. The Python and Java versions of the game should perform identically in terms of game play and output.
8. Follow best practices for code readability, such as consistent naming conventions, code formatting, and the use of comments especially for key functions (e.g., loading words, game flow, etc.).

Submission Guidelines

Submit a zip file containing:

1. The source code files for Python (hangman_game.py) and Java (HangmanGame.java) programs.
2. Include a separate PDF file named "A2_ experience.pdf" corresponding to Rating your experience:

Marking Scheme

Criteria	***Points***
Correctly loading categories and words from the words.txt file.	10
The game runs as expected, with categories, word guessing, and winning logic.	10
Correctly displaying the current state of the word, list of guessed letters, and tries left after each guess.	10
Invalid input handling.	10
A2_ experience.pdf	10
Total	*50*

Sample Runs:

1. Winning game:

```
Welcome to the Hangman Game!
Select a category:
1. Animals
2. Movies
3. Countries
4. Food
Enter the number of your category choice: 1

You selected 'Animals'. The word is 5 letters long.

Current word: _ _ _ _ _
Tried letters:
Tries left: 6
--------------------------------------------------------------------------------

Guess a letter: a
Sorry, the letter 'a' is not in the word.

Current word: _ _ _ _ _
Tried letters: a
Tries left: 5
--------------------------------------------------------------------------------

Guess a letter: e
Good guess! The letter 'e' is in the word.

Current word: _ _ _ e _
Tried letters: a, e
Tries left: 5
--------------------------------------------------------------------------------

Guess a letter: t
Good guess! The letter 't' is in the word.

Current word: t _ _ e _
Tried letters: a, e, t
Tries left: 5
--------------------------------------------------------------------------------

Guess a letter: s
Sorry, the letter 's' is not in the word.

Current word: t _ _ e _
```

```
Tried letters: a, e, t, s
Tries left: 4
--------------------------------------------------------------------------------

Guess a letter: i
Good guess! The letter 'i' is in the word.

Current word: t i _ e _
Tried letters: a, e, t, s, i
Tries left: 4
--------------------------------------------------------------------------------

Guess a letter: g
Good guess! The letter 'g' is in the word.

Current word: t i g e _
Tried letters: a, e, t, s, i, g
Tries left: 4
--------------------------------------------------------------------------------

Guess a letter: r
Good guess! The letter 'r' is in the word.

Congratulations! You've guessed the word: tiger
```

2. Losing game:

```
Welcome to the Hangman Game!
Select a category:
1. Animals
2. Movies
3. Countries
4. Food
Enter the number of your category choice: 2

You selected 'Movies'. The word is 9 letters long.

Current word: _ _ _ _ _ _ _ _ _
Tried letters:
Tries left: 6
--------------------------------------------------------------------------------

Guess a letter: a
Good guess! The letter 'a' is in the word.
```

```
Current word: _ _ a _ _ a _ _ _
Tried letters: a
Tries left: 6
--------------------------------------------------------------------------------

Guess a letter: e
Sorry, the letter 'e' is not in the word.

Current word: _ _ a _ _ a _ _ _
Tried letters: a, e
Tries left: 5
--------------------------------------------------------------------------------

Guess a letter: i
Good guess! The letter 'i' is in the word.

Current word: _ _ a _ i a _ _ _
Tried letters: a, e, i
Tries left: 5
--------------------------------------------------------------------------------

Guess a letter: t
Good guess! The letter 't' is in the word.

Current word: _ _ a _ i a t _ _
Tried letters: a, e, i, t
Tries left: 5
--------------------------------------------------------------------------------

Guess a letter: k
Sorry, the letter 'k' is not in the word.

Current word: _ _ a _ i a t _ _
Tried letters: a, e, i, t, k
Tries left: 4
--------------------------------------------------------------------------------

Guess a letter: s
Sorry, the letter 's' is not in the word.

Current word: _ _ a _ i a t _ _
Tried letters: a, e, i, t, k, s
Tries left: 3
--------------------------------------------------------------------------------

Guess a letter: v
Sorry, the letter 'v' is not in the word.
```

```
Current word: _ _ a _ i a t _ _
Tried letters: a, e, i, t, k, s, v
Tries left: 2
--------------------------------------------------------------------------------

Guess a letter: m
Sorry, the letter 'm' is not in the word.

Current word: _ _ a _ i a t _ _
Tried letters: a, e, i, t, k, s, v, m
Tries left: 1
--------------------------------------------------------------------------------

Guess a letter: n
Sorry, the letter 'n' is not in the word.

Game Over! The correct word was: gladiator
```