

Coding Challenge

Code Challenge – Hangman

The Problem

Create a game of hangman

Rules & Requirements

1. Greet the user welcoming them to the game
 - `Game.startGame()`
2. The program randomly selects a word from a list of 10 words with different lengths, you as the developer choose the words.
 - Create a list with words (vector?)
 - `Game myGame = Game(<pass a word as a parameter>)`
3. The program indicates to the user how many letters are in the word
 - `Game.word.getLength()`
4. The user is asked to guess a letter
 - build a function (inside game): `isLetterInside(userInput)` = analyze if the letter is inside the word. How?
 - traverse through each char and compare.
 - return bool so that it can be run through the `if()` statement :
`if(game.isLetterInside(userInput)) {}`

0	1	2	3	4	5
s	p	i	d	e	r

vector correctGuesses:

p, e, d

check if each char of the word is in the vector,
if so, then add it to the word.

chars or string correctGuesses[size of the word]

0	1	2	3	4	5
—	—	—	—	e	—

- a. If the letter is in the word, the letter is displayed in the correct position of the word with all previously guessed correct letters

`displayCorrect()`

b. If the letter is not in the word, display the letter indicating it is not in the word with all previously guessed letters that are not in the word

`displayIncorrect()`

5. The program displays how many guesses have been made, with how many correct and incorrect guesses.

`displayTotalGuesses()`

`Should display guessesLeft()`

6. The program continues to ask the user for guesses until all the letters in the word are guessed correctly.

`do-while loop`

7. When all letters of the word are guessed correctly,

- `when correctGuesses == word`

a. the program tells the user they have correctly guessed the word

`winner()`

b. and indicates the number of guesses it took

8. The program then asks the user if they would like to try again or quit

`in main:`

`do{`

`} while (PlayAgain())`

a. If the user indicates they want to continue, the program chooses a different word randomly and the play continues

b. If the user indicates they want to quit, the **program thanks them for playing** and quits.

9. You can choose to use a terminal interface or a web interface

Extra credit

10. Draw a gallows and person being hanged drawing a new body part each time a guess is wrong

`in excel`

	1	2	3	4	5	6	7	8
1	_	_	_	_	_	_	_	
2								
3								
4								
5							o	
6						/		\
7								
8						/		\

9	_	_	_	_	_			
10	=	=	=	=				

☐ write all functions in the main class and then make classes if i have time

Main function:

- playAgain():
 - ask the user if he wants to play again. returns true or false.
 - This function is used in the do {} while (playAgain())
- calls the startGame() function.

Game class:

- Attributes:
 - word
 - hangman?
 - correct guesses
 - incorrect guesses
 - set of wrong guesses (char??) (set to avoid duplicates)
 - array correct guesses (char??)
 - initialize it with the length of the word, and display it with _ _ _ _
 - —
- Constructor:
 - pass a word randomly selected previously.
- Functions:
 - void Startgame()
 - void Welcome message()
 - void displayIncorrect()
 - void displayCorrect()
 - voidDisplay()
 - getTotalGuesses()
 - ...

Hangman class:

- different shapes, according to the percentage of completion.