← Back to study plan

Work on project. Stage 6/8: The value of life

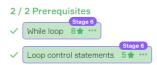
Project: Hangman

→ Hard ② U 19 minutes

1915 users solved this problem. Latest completion was in less than a minute.

Description

The recent version of the game is not as fun until we don't handle the player's victory. A player has 8 attempts to guess letters and its number is reduced even if the letter was correct.



Now a player will have a lot of attempts and is limited only by the number of mistakes they make. A player can be mistaken 8 times and wins when all the letters are guessed and there are still some tries left. If the player uses the last try and actually guesses the word, they are lucky

Objectives

The player starts the game with 8 "lives", that is our player can input the wrong letter 8 times.

- 1. Print No such letter in the word and reduce the attempts count if the word guessed by the program doesn't contain this letter.
- 2. Print No improvements and reduce the attempts count if the guessed word contains this letter but the user tried this letter before.
- 3. The attempts count should be decreased only if there are no letters to uncover.

Please, make sure that your program's output formatting precisely follows the example output formatting. Pay attention to the empty lines

Examples

The greater-than symbol followed by space (>) represents the user input. Notice that it's not the part of the input.

Example 1

```
HANGMAN
    Input a letter: > t
    Input a letter: > z
    No such letter in the word
10
11 Input a letter: > t
    No improvements
14
     Input a letter: > t
    No improvements
    Input a letter: > y
     -vt---
    Input a letter: > x
    No such letter in the word
24
    Input a letter: > y
    No improvements
     -yt---
30
     Input a letter: > p
     Input a letter: > p
34
    No improvements
     Input a letter: > q
38
     No such letter in the word
40
     pyt---
```

```
1 Input a letter: > p

No improvements

You are hanged!
```

Example 2

```
\mathsf{H} \mathsf{A} \mathsf{N} \mathsf{G} \mathsf{M} \mathsf{A} \mathsf{N}
     Input a letter: > j
     j---
Input a letter: > i
6
8
     No such letter in the word
9
10
11 Input a letter: > g
12 No such letter in the word
14 j---
     Input a letter: > g
16
     No such letter in the word
    j---
Input a letter: > g
18
19
20
    No such letter in the word
23 Input a letter: > g
     No such letter in the word
24
26
     Input a letter: > a
28
29
30
     Input a letter: > v
32 java
      You guessed the word!
34 You survived!
```

Code Editor

IDE

```
Python
1 # Updated on 24th June, 2020 by Shovan Saha
3 import random
word_list = ["python", "java", "kotlin", "javascript"]
for random_word = random.choice(word_list)
dashdash = ["-" for i in range(len(random_word))]
10 def word_revealer(user_input1=None):
      global dashdash
11
       indices = []
12
13
       if user_input1:
          # for determining all the position of the user_input in random_word
14
15
           for position, value in enumerate(random_word):
             if value == user_input1:
16
           indices.append(position)
17
18
           # inserting the user_input on dashdash in the positions where it
19
20
            # is found in the random_word
21
            for value in indices:
22
23
          dashdash[value] = user_input1
24
25
26 def status_printer():
27 print("".join(dashdash))
28
29
30 print("H A N G M A N")
31 count = 1
32 while count < 9:
33
      print()
34
       status_printer()
35
       user_input = input("Input a letter: ")
36
37
       word_revealer(user_input)
38
39
       if user_input not in random_word:
40
           print("No such letter in the word")
41
        count = count + 1
```

```
42
43
44 print()
45 print(
46 """Thanks for playing!
47 We'll see how well you did in the next stage"""
  48 )
   49
   50
Run
                                                                                                                                                            View solution ( 100)
```

Show discussion (415)

About Support Contribute Terms How do we teach 🍪 👍

