← Back to study plan

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

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

Project: Hangman

Hard 19 minutes

Description

2 / 2 Prerequisites

- ✓ While loop 8 ★ — Stage 6
- ✓ Loop control statements 5 ★ — Stage 6

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 then!

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 between tries and in the end.

Examples

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

Example 1

```
H A N G M A N

------
Input a letter: > t

--t---
Input a letter: > z
No such letter in the word

--t---
Input a letter: > t
No improvements

--t---
Input a letter: > t
No improvements

--t---
Input a letter: > y

-yt---
Input a letter: > x
No such letter in the word

-yt---
Input a letter: > y
No improvements

-yt---
Input a letter: > p

pyt---
Input a letter: > p
No improvements

pyt---
Input a letter: > q
No such letter in the word

pyt---
```

```
Input a letter: > p
No improvements
You are hanged!
```

Example 2

```
H A N G M A N

----
Input a letter: > j

j---
Input a letter: > i
No such letter in the word

j---
Input a letter: > g
No such letter in the word

j---
Input a letter: > g
No such letter in the word

j---
Input a letter: > g
No such letter in the word

j---
Input a letter: > g
No such letter in the word

j---
Input a letter: > a

ja-a
Input a letter: > v

java
You guessed the word!
You survived!
```

Write a program

Code Editor | IDE

Python

```python
# Updated on 24th June, 2020 by Shovan Saha

import random

word_list = ["python", "java", "kotlin", "javascript"]
random_word = random.choice(word_list)
dashdash = ["-" for i in range(len(random_word))]


def word_revealer(user_input1=None):
    global dashdash
    indices = []
    if user_input1:
        # for determining all the position of the user_input in random_word
        for position, value in enumerate(random_word):
            if value == user_input1:
                indices.append(position)

        # inserting the user_input on dashdash in the positions where it
        #  is found in the random_word

        for value in indices:
            dashdash[value] = user_input1


def status_printer():
    print("".join(dashdash))


print("H A N G M A N")
count = 1
while count < 9:
    print()
    status_printer()
    user_input = input("Input a letter: ")

    word_revealer(user_input)

    if user_input not in random_word:
        print("No such letter in the word")
    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)