

On Thursday, February 16th at 6:00AM EST, UTC-5, we will be conducting a brief database maintenance. The event should last about 5 minutes.



Bookmarks

- ▶ [Welcome to the edX Platform](#)
- ▶ [Entrance Survey](#)
- ▶ [Download Python and Get Motivated!](#)
- ▶ [Week 1: Python Basics](#)
- ▶ [Week 2: Simple Programs](#)
- ▶ [Week 3: Structured Types](#)
- ▼ [Week 4: Good Programming Practices](#)
 - [7. Testing and Debugging](#)
[Finger Exercises](#)
 - [8. Exceptions and Assertions](#)
[Finger Exercises](#)

[Problem Set 4](#)

Week 4: Good Programming Practices > Problem Set 4 > Problem 6 - Playing a Game

Problem 6 - Playing a Game

[Bookmark this page](#)

Problem 6 - Playing a Game


15.0 points possible (graded)

A game consists of playing multiple hands. We need to implement one final function to complete our word-game program. Write the code that implements the `playGame` function. You should remove the code that is currently uncommented in the `playGame` body. Read through the specification and make sure you understand what this function accomplishes. For the game, you should use the `HAND_SIZE` constant to determine the number of cards in a hand.

Testing: Try out this implementation as if you were playing the game. Try out different values for `HAND_SIZE` with your program, and be sure that you can play the wordgame with different hand sizes by modifying *only* the variable `HAND_SIZE`.

Sample Output

Here is how the game output should look...

Problem Set due Feb
23, 2017 15:30 PST 

- ▶ Midterm Exam
- ▶ Week 5: Object Oriented Programming
- ▶ Sandbox

Loading word list from file...

83667 words loaded.

Enter n to deal a new hand, r to replay the last hand, or e to end
You have not played a hand yet. Please play a new hand first!

Enter n to deal a new hand, r to replay the last hand, or e to end

Current Hand: p z u t t t o

Enter word, or a "." to indicate that you are finished: tot

"tot" earned 9 points. Total: 9 points

Current Hand: p z u t

Enter word, or a "." to indicate that you are finished: .

Goodbye! Total score: 9 points.

Enter n to deal a new hand, r to replay the last hand, or e to end

Current Hand: p z u t t t o

Enter word, or a "." to indicate that you are finished: top

"top" earned 15 points. Total: 15 points

Current Hand: z u t t

Enter word, or a "." to indicate that you are finished: tu

Invalid word, please try again.

Current Hand: z u t t

Enter word, or a "." to indicate that you are finished: .

Goodbye! Total score: 15 points.

Enter n to deal a new hand, r to replay the last hand, or e to end

Current Hand: a q w f f i p

Enter word, or a "." to indicate that you are finished: paw

"paw" earned 24 points. Total: 24 points

Current Hand: q f f i

Enter word, or a "." to indicate that you are finished: qi

"qi" earned 22 points. Total: 46 points

Current Hand: f f

Enter word, or a "." to indicate that you are finished: .

Goodbye! Total score: 46 points.

Enter n to deal a new hand, r to replay the last hand, or e to end

Current Hand: a r e t i i n

Enter word, or a "." to indicate that you are finished: inertia

"inertia" earned 99 points. Total: 99 points.

Run out of letters. Total score: 99 points.

```
Enter n to deal a new hand, r to replay the last hand, or e to end
Invalid command.
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
```

Hints about the output

Be sure to inspect the above sample output carefully - very little is actually printed out in this function specifically. Most of the printed output actually comes from the code you wrote in `playHand` - be sure that your code is modular and uses function calls to the `playHand` helper function!

You should also make calls to the `dealHand` helper function. You shouldn't make calls to any other helper function that we've written so far - in fact, this function can be written in about 15-20 lines of code.

Here is the above output, with the output from `playHand` obscured:

```
Loading word list from file...
```

```
83667 words loaded.
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
You have not played a hand yet. Please play a new hand first!
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
<call to playHand>
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
<call to playHand>
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
<call to playHand>
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
Invalid command.
```

```
Enter n to deal a new hand, r to replay the last hand, or e to end
```

Hopefully this hint makes the problem seem a bit more approachable.

Entering Your Code

Be sure to only paste your definition for `playGame` in the following box. Do not include any other function definitions.



A Cool Trick about 'print'

A cool trick about `print`: you can make two or more print statements print to the same line! Try out the following code. It will separate the first and second line with a space, and the second and third line with a "?" rather than putting each on a new line.

```
print('Hello', end = " ")
print('world', end="?")
print('!')
```

```
1 def playGame(wordList):
2     """
3     Allow the user to play an arbitrary number of hands.
4
5     1) Asks the user to input 'n' or 'r' or 'e'.
6         * If the user inputs 'n', let the user play a new (random) hand
7         * If the user inputs 'r', let the user play the last hand
8         * If the user inputs 'e', exit the game.
9         * If the user inputs anything else, tell them their input
10
11     2) When done playing the hand, repeat from step 1
12     """
13     # TO DO ... <-- Remove this comment when you code this function
14
```

Press ESC then TAB or click outside of the code editor to exit

Unanswered

Note: the `input` function on Spyder may print an extra newline. That's ok. Do not try to move text backwards using `end='\b'` in a print statement

Submit

You have used 0 of 30 attempts

Problem 6 - Playing a Game

Topic: Problem Set 4 / Problem 6

Show Discussion





© 2012-2017 edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

