Computer systems quiz

# In today’s lesson you are going to spend some time revising the components and concepts you have covered so far. To do this, you are going to create a quiz using the questions you have written.

Starter files: <https://repl.it/@NCCE/Computer-Systems-Quiz>

Task 1 .

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | import random  questions = []  file = open("questions.txt", 'r')  for line in file:  line = line.rstrip()  questions.append(line)  print(questions) |

Q1. What will this code do when it is executed?

|  |
| --- |
| Print a string containing each line of the file “questions.txt”. |

Task 2 .

For each line in the code, read and discuss with a partner. Then you can write comments here to explain them.

|  |  |
| --- | --- |
| 1 | import random |

|  |
| --- |
| Import the random package |

|  |  |
| --- | --- |
| 3 | questions = [] |

|  |
| --- |
| Create an empty string and define it to the variable questions. |

|  |  |
| --- | --- |
| 5 | file = open("questions.txt", 'r') |

|  |
| --- |
| Open questions.txt in read-only mode, and assign it to the variable “file”. |

|  |  |
| --- | --- |
| 7 | for line in file: |

|  |
| --- |
| Repeat whats in the for loop for every line in the file, assigning the relevant line to the variable “line” each time. |

|  |  |
| --- | --- |
| 8 | line = line.rstrip() |

|  |
| --- |
| Strip all whitespace from each end of the line. |

|  |  |
| --- | --- |
| 9 | questions.append(line) |

|  |
| --- |
| Append the line to the end of the list “questions” |

|  |  |
| --- | --- |
| 11 | print(questions) |

|  |
| --- |
| Print the list of each line |

Task 3 .

Look at the questions.txt file. Use the questions below to investigate how it works.

Q1. What is the pattern in the way each question is written out?

|  |
| --- |
| Each line is first the question, followed by the answer. |

Q2. How could you separate the sections of each question?

|  |
| --- |
| Use the python split function on the comma |

Q3. What is the divider?

|  |
| --- |
| , |

Task 4 .

Make the quiz your own!

Add your questions to the quiz, from the question log you have been working on.

Check them with a partner. You can use the questions below to help.

Copy this symbol - ✔ - to check off the questions.

|  |  |
| --- | --- |
| ✔ | Do your questions match the pattern of the originals? |
| ✔ | Run the program again. Does it still work the same? |

Coding the quiz .

Task 1 .

Replace this line in your program:

|  |
| --- |
| ~~print(questions)~~ |

With this loop to go through each of the questions:

|  |
| --- |
| for q in questions:  print(q) |

Task 2 .

You can split strings in Python using the string.split() method.

|  |
| --- |
| line = “Hello,world”  line = line.split(“,”)  print(line)  > [“Hello”, “world”] |

In order to use this in your program, you need to change it to match your variable. What are your lines called?

|  |
| --- |
| ? = ?.split(“,”)  print(?) |

Change the question marks to match your line variables and add these lines inside of your for loop.

Task 3 .

**Challenge time:** Add code to your program to take an answer from the player.

In Python, you can get input with a specific function:

|  |
| --- |
| variable = input(“message”) |

Test your program after making those changes.

Task 4 .

q[1] - the correct answer from the file

answer - the input from the user

Write an if statement inside the loop that:

1. Compares the correct answer to the user’s input
2. Gives feedback so the player knows if they got the question right, or not.

Playing the quiz .

Feedback .

You are going to ask other learners to give you feedback on your quiz. You should record this feedback here so you can make improvements.

|  |
| --- |
|  |

New questions .

You are also going to add some more questions based on your experience playing other people’s quizzes. Note them down here.

|  |  |
| --- | --- |
| Question | Answer |
|  |  |
|  |  |

Improving the quiz .

Task 1 .

Replace this line in your program:

|  |
| --- |
| ~~if q[1] == answer:~~ |

With this one:

|  |
| --- |
| if q[1].lower() == answer.lower(): |

Task 2 .

You can shuffle the questions before you ask them so that they have a random order.

To do this, you will use the random module:

|  |
| --- |
| questions.shuffle() |

Where should this line go?

Add it to your program.

Task 2 .

**Challenge time:** Add a score to your quiz.

Use these questions to help you add a score to your quiz:

1. Where should you create the score variable?
2. When does it need to increase?
3. When do you show it to the user?

Playing the quiz II .

Feedback .

You are going to ask other learners to give you feedback on your quiz. You should record this feedback here so you can make improvements.

|  |
| --- |
|  |

New questions .

You are also going to add some more questions based on your experience playing other people’s quizzes. Note them down here.

|  |  |
| --- | --- |
| Question | Answer |
|  |  |

This resource is available online at [ncce.io/cpst-9-a1-w](https://docs.google.com/document/d/1qOXsEIRjBbBJ0CQ_8KvVPuwhIo2yImkP6vVW6bJLc0s). Resources are updated regularly — please check that you are using the latest version.

This resource is licensed under the Open Government Licence, version 3. For more information on this licence, see [ncce.io/ogl](http://ncce.io/ogl).