**City guessing**

Task .

You will start with an **existing program** that randomly picks a European capital out of a list and asks the user to guess it. You will **extend** the program so that it provides the user with hints.

**Step 1**

**Open** this [program](https://ncce.io/py-cities-1) (ncce.io/py-cities-1) in your development environment.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | from data import cities  from random import choice   |  |  | | --- | --- | | city = choice(cities) | Pick a city . |   done = False  while not done:   |  |  | | --- | --- | | print("Guess the capital:")  guess = input() | Ask the user . |  |  |  | | --- | --- | | if guess == city:  print("You've got it!")  done = True  elif guess == "":  print("It was", city)  done = True  else:  print("Try again") | Give feedback . | |

Line 1 imports the list of cities that the program will use. Note that this is **not a standard Python component**. The list has been created specifically to allow you to perform this activity.

The choice function (imported from the random module in line 2) is used to randomly select an item out of the list of cities (line 3).

**Step 2**

**Run** the program a couple of times to see how it currently works. You will find it is very hard for the user to guess the city, without any additional information.

**Tip:** Look at lines 11 to 13. While testing the program you can simply press Enter (without entering a city) and the city will be revealed, terminating the game.

**Step 3**

**Add** the incomplete lines below to your program, right above the else-block.

**Complete** the gaps so that the program reveals the first letter of the city to the user, in the case where the city’s first letter is different from the first letter of the user’s guess.

**Tip:** You can access a specific character in a string just like you can access a specific item in a list, by using the string’s name followed by an index in square brackets.

**Tip:** You must **indent** these lines properly, as they are part of the while-block.

|  |  |
| --- | --- |
| +  + | elif :  print("The first letter is", )  else:  print("Try again") |

|  |  |
| --- | --- |
| **Example** |  |
| Note: This example illustrates how your program should work. The output of your program will depend on the randomly selected city and the user’s input, so it will be different each time you execute it. | |
| The program displays a prompt and waits for keyboard input. | Guess my European capital: |
| The user types a reply. | Prague |
| The program displays a hint. The selected city is actually Moscow (different first letter). | The first letter is M |

**Step 4**

**Add** the incomplete lines below to your program, right above the else-block.

**Complete** the gaps so that the program reveals the length (number of characters) in city to the user, in the case where city’s length is different from the length of the user’s guess.

**Tip:** Use the len function to retrieve the number of characters in city and guess.

**Tip:** You must **indent** these lines properly, as they are part of the while-block.

|  |  |
| --- | --- |
| +  + | elif :  print("It has", , "letters")  else:  print("Try again") |

|  |  |
| --- | --- |
| **Example** |  |
| Note: This example illustrates how your program should work. The output of your program will depend on the randomly selected city and the user’s input, so it will be different each time you execute it. | |
| The program displays a prompt and waits for keyboard input. | Guess my European capital: |
| The user types a reply. | Prague |
| The program displays a hint. The selected city is actually Paris (same first letter but different length). | It has 5 letters |

**Step 5**

**Add** the incomplete lines below to your program, right above the else-block.

**Complete** the gaps so that the program reveals the second letter of city to the user, in the case where city’s second letter is not contained in the user’s guess.

**Tip:** Use the in operator to check if a character is contained in a string.

**Tip:** You must **indent** these lines properly, as they are part of the while-block.

|  |  |
| --- | --- |
| +  + | elif :  print("It contains letter", )  else:  print("Try again") |

|  |  |
| --- | --- |
| **Example** |  |
| Note: This example illustrates how your program should work. The output of your program will depend on the randomly selected city and the user’s input, so it will be different each time you execute it. | |
| The program displays a prompt and waits for keyboard input. | Guess my European capital: |
| The user types a reply. | London |
| The program displays a hint. The selected city is actually Lisbon (same first letter, same length, but second letter of city is not contained in guess). | It contains letter i |

Explorer task . How many attempts

**Extend** the program so that it counts the number of attempts the user makes and displays them after the game is over.