Magic 8 Ball Game

Python Bootcamp: Day One

The Game

The Magic 8 ball game is intended to be a fortune-telling game that uses the magic 8 ball. The actual ball is a large, hollow black sphere with a small icosahedron inside (20 sided polyhedron). On each face of the icosahedron, there is a message that answers a yes-no question. The game is played with the player asking a yes-no question, then rolling the magic 8 ball. Once a message appears, that message is used to answer the player's question.

The Program

You are to write a program that will replicate the Magic 8 Ball Game. The program should prompt the user to type in a question. It should then randomly generate an answer and print that answer to the player.

Requirements

Functionality

When the program starts, there should first be an introduction paragraph that introduces the user on what the program is. After the introduction, the program should ask the user to type in a question. The user should be able to end the game by typing only *enter* key when prompt. If the user inputs a question, randomly generate an answer, and print the answer. The program will then prompt again for a question.

The program should continue asking for the user to input a question until the user terminates the program by entering the *enter* key. Refer to the example below to see what the expected program should output.

Figure 1: Example Output

```
This is a Magic 8-Ball Game
When prompt, give a question that can be answered yes or no.
The magic 8 ball will give it's response


What is your question?
Press enter to exit: Is this example trivial?
Very doubtful.

What is your question?
Press enter to exit: Will I be able to pull my favorite unit in x game?
You may rely on it.

What is your question?
Press enter to exit:
```

Output

The following are the 20 possible answers inside a standard Magic 8 Ball:

- It is certain.
- It is decidedly so.
- Without a doubt.
- Yes - definitely.
- You may rely on it.
- As I see it, yes.
- Most likely.
- Outlook good.
- Yes.
- Signs point to yes.
- Reply hazy, try again.
- Ask again later.
- Better not tell you now.
- Cannot predict now.
- Concentrate and ask again.
- Don't count on it.
- My reply is no.
- My sources say no.
- Outlook not so good.
- Very doubtful.

It is expected that when the program randomly generates an answer, one of the above messages is used.

Other

Various topics was covered in the lecture portion of the bootcamp. For this project, it is required to apply the concept of dictionaries to help organize your output strings. It is also required to use the *random* library.