Programming Lab 2 – Magic 8-Ball
PHYS 2511 – Prof. Matthew Newby – Spring 2019

Goal:	Create a random “Fortune Teller” program in Python
Requirements:	Take a question as input, repeat the question and at least a yes or no answer.
Inputs:	• A yes or no question as text
Outputs:	• The question, repeated • A time delay (for dramatic tension) • A randomly-chosen answer to the question
Optional:	• Include a large number of possible answers (20+) • Create a graphical interface

Background:

The “Magic 8-Ball” is a popular toy that pretends to answer yes or no questions wuth what amounts to a physical random answer chooser. We will make a python program that answers yes or no questions with a randomly-chosen answer.

We will need the *random* and *time* python modules (libraries). Specifically we will need the “randint” and “sleep” functions, respectively. To learn what these do, search (online) the python random and time modules for these functions.

There are several ways to get what we need into our program:

Option	What to do to import the module	How to use functions in your program
1	`import random` `import time`	`random.randint(0,1)` `time.sleep(0.5)`
2	`import random as rn` `import time as tm`	`rn.randint(0,1)` `tm.sleep(0.5)`
3	`from random import randint` `from time import sleep`	`randint(0,1)` `sleep(0.5)`
4	`from random import *` `from time import *`	`randint(0,1)` `sleep(0.5)`

- Option 1 imports the entire module, but requires you to name the module every time you call functions from it.
- Option 2 imports the entire module, but with a user-defined nickname for the module.
- Option 3 imports only the specific function, and makes it available without naming its home module.
- Option 4 imports the entire module, and makes it available without naming its home module.

Option 4 is the most “dangerous,” as different modules may have functions with the same name, in which case only the function from the last-imported module will be used. Option 2 is recommended; good nicknames will be quick to type, it is clear where every function is coming from (making reading and debugging your code easier), and there’s no chance of accidentally using the wrong function (as in Option 4).

The user experience for you program should look something like this:

```
“What is your question?” <user types and enters a question>

“Thinking...”  <time delay>

“Your question was ” <repeat question>

“My answer is ” <the text associated with a random number>
```

You are free to change the wording and/or add additional delays as you would like.