Model 1   Guessing Game

Create a new file named `guess.py` and enter the following code. Replace the name in Line 2 with your own name. Be careful to type the code *exactly* as shown.

```
1  name = raw_input("What is your name? ")
2  if name == "Taylor":
3      print name, "is a great name!"
4  else:
5      print name, "is an okay name."
```

Note: `raw_input` is a **function** that displays a **prompt** on the screen and reads a line from the keyboard. In this program, the result of `raw_input` is stored in the **variable** `name`.

Questions (15 min)   Start time: _____

1. What is the prompt? Why is there a space at the end of it?

The prompt is "`What is your name?   `". The space at the end makes it so that the input isn't "touching" the question mark when the user types.

2. Run the program a few times, entering a different name each time. Feel free to modify the messages as you see fit.

3. Enter each of these lines into the IDLE shell, and explain where the syntax error occurs.

a) `name? = raw_input("What is your name?")` question mark

b) `your name = raw_input("What is your name?")` space between your and name

c) `1st_name = raw_input("What is your name?")` the word 1st

d) `from = raw_input("Where were you born?")` the equals sign (from is a keyword)

4. Based on the errors in the previous question and the following correct examples, describe three rules that need to be followed when naming a variable.

```
name2 = raw_input("What is your name?")
your_name = raw_input("What is your name?")
firstName = raw_input("What is your name?")
```

5. At the end of your `guess.py` program, create two new variables named `number` and `guess`. Set the value of `number` to be an integer between 1 and 100 (of your choice). Ask the user to guess your number, and store the result in `guess`. When asking for numbers, use `input` instead of `raw_input`. Write your two statements in the space below.

```
number = 74   # or some other value
guess = input("Guess my number: ")
```

6. Add the following logic to your program: If the guess is too high, display the message `"Too high!"`; if the guess is too low, display the message `"Too low!"`; if the user guessed the number, display `"You got it!"`. Write your statements in the space below.

```
if guess < number:
    print "Too low!"
if guess > number:
    print "Too high!"
if guess == number:
    print "You got it!"
```

7. What is the difference between = and == in the programs you have written today?

8. At this point, you should have a program that allows the user to make only one guess. Rather than run this program over and over again, you can use a `while` loop to make it repeat the guessing part. Insert the following two lines before the `input` line you wrote in #5.

```
guess = -1
while guess != number:
```

9. What did you have to do after inserting the `while` loop to make it work? In other words, how did you make the `input` and `if` statements part of the `while` loop?

10. Rather than guess the same number every time, you can have the computer select a random number for you:

- At the top of your program, add the line "`import random`" (without the quotes).
- Then change the line where you set value of `number` to use this example instead:

```
number = random.randint(1, 100)
```