# CIS 500 – Fundamentals of Software Practice
## Weekly Exercise #3

## Topic: Decisions and Loops

**Number Guessing Game**: Write a program that randomly selects a number from 1 to 100 when the user starts the game and asks the user to guess that number. If the user's guess is correct, the program congratulates the user for the perfect answer and then make it possible for the user to start a new game; otherwise, it will tell the user that it is too high or too low and let the user continue to play.

In your program, you may use the following code to generate a random number from 1 to 100 as the secret number each time user starts a new game.

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

In addition, your program will keep track of the number of guesses the user made and display the number of guesses when a game is over. If a user is successful by more than 7 tries, display an additional message to tell the user to consider a different strategy in order to improve one's performance.

Given below is a typical dialog between a user and your program (text in **red** is user's input):

Hello, what is your name? **John**
John, I am thinking of a number between 1 and 100 (both included). Can you guess what it is…
Guess a number (1-100): **50**
Your guess is too high. Try again.
Guess a number (1-100): **25**
Your guess is too low. Try again.
Guess a number (1-100): **35**
Your guess is too low. Try again.
Guess a number (1-100): **43**
John, you won in 4 tries. Congratulations!
Do you want to continue to play? **no**
Thank you for playing this game. Bye.

In the above execution, the user plays the game only once. If the user says 'yes' at the second to last step, one would be able to play the game again.

Due to the complexity of this program (which involves both branching and loop statements), it would be beneficial to write pseudocode/algorithm on paper and work out its processing logic before writing Python code. Do not hurry to write code in your programming environment immediately.

## Exercise Instructions:
- Download the file **WEX-3.py** from Blackboard and edit the file to place the code to solve the problem.
- Run and test your code to make sure it works correctly.
- Upload the file **WEX-3.py** on Blackboard by midnight of due date.