Code Exercise 1:

In a language of your choice write a program that will take a random integer as a parameter, then have the computer (your program) try to guess it.

After each guess, you will tell the computer either **h** (too high), **l** (too low) or **c** correct.

The program should use the best possible strategy to minimize the number of guesses (it may be wise to use an existing search algorithm).

It should never need more than 1 + $\log_2$ n guesses. For example, if n = 1000, then it should always be correct by the tenth guess. If n is any number 4 through 7 then it should take at most 3 guesses.
The program will start by printing a brief description of the rules of the game and asking for a number n.

Then it will guess at most 1 + $\log_2$ n numbers, after which you enter the letter h, l, or c and press ENTER. On the last guess, if there is only one possible correct answer, the program will simply print "your number is *x*" rather than ask for a response.

After the program has guessed your number, it will print the number of guesses it took for this game (including the correct guess at the end), the number of games played, the average number of guesses per game, and ask if you want to play again. If you enter **y** (and ENTER) then the game repeats (with the same value of n). If you enter anything else, then the program exits.

For example (Sample Input/Output):

```
In this game, you think of a number from 1 through n and I will try to guess what it is.  After each guess,
enter h if my guess is too high, l if too low, or c if correct.

Please enter a number n: 100
50? h
25? l
37? l
43? h
40? h
38? l
Your number is 39.
It took me 7 guesses.
I averaged 7 guesses per game for 1 game(s).
Play again (y/n)? y
50? l
75? l
88? l
94? c
Your number is 94.
It took me 4 guesses.
I averaged 5.5 guesses per game for 2 game(s).
Play again (y/n)? N
```

Code Exercise 2:

One of the biggest part of ensuring that data is ready for use for analytics, applications and reporting is ensuring certain data quality standards are met.

Using the provided provided CSV file about library books and Python Pandas and NumPy libraries as needed, do the following:

1. Clean the provided CSV file, removing the following columns:
    a. Edition Statement
    b. Corporate Author
    c. Corporate Contributors
    d. Former Owner
    e. Engraver
    f. Contributors
    g. Issuance type
    h. Shelfmarks
2. Change the Pandas Index to use the Identifier field.
    a. Demonstrate if the index is unique
    b. Show the difference between label-based indexing vs. position-based indexing
3. Data cleanup:
    a. Enforce the Date of Publication field to Numeric value, removing any extra characters. If there are multiple dates, use the first date present.
    b. Standardize the Place of Publication so that the each place is standardized, removing any extra characters as needed
4. Once cleaned up, output the file with standardized column names for the columns that are left and provide the output.

Please be ready to show all of your code and answer questions about it from the panel.