Exercise Module 2 (20483): *Guess the Number*

We want to make a program, which generates a secret random number, which the user has to guess. after every guess the output will show whether the secret number is higher or lower compared to the user input. Keep going till the user guesses the number correctly.

Task 1: Flowchart

Try to draw a flow chart of the main flow of the program, starting with a initializing block, followed by a user input block, etc.

Task 2: Generate Random Secret Number

Generate a random secret number using the following code:

```
Random rnd = new Random();

int secret = rnd.Next(100);
```

This gives a random number from 0 to and including 99.

Task 3: User input

Create a method to get the user input.

- What are the parameters needed and what is the return type?
- Console.Readline() provides you with a string, convert it to an int.
- Make sure the exceptions are handled when the user input is not a number and thus can't be converted to an integer.
- If you exception is caught, you want to ask the user a number again.

Task 4: Compare the User Input with the Secret

Compare the two integers and let the user know, whether the secret number is higher or lower. When it is not correct, ask for a new input, etc, etc

Task 5: Success

Congratulate the player when he guesses the number correctly and let him know how many guesses he/she needed.

Exercise Module 3 (20483): Higher/Lower with Cards

We are going to make a similar program to the one we made for module 2. This time we want to have different playing cards and then see if the next card is going to be lower or higher. Order is (A, K, Q, J, 10, 9……, 3, 2). All suits are the same.

For instance, you get Ten of Clubs as the first card, you guess that the next card is lower and then Seven of Hearts gets revealed, giving you 1 point.

Task 1: Card struct

Create a struct for the playing cards, containing a suit and a facevalue. You can use an enum for the facevalue or a string variable, in case of a string, you might need to have a value member as well to be able to compare different cards.

Task 2: Make a Card Deck

Create a deck with the 52 different cards

Task 3: Draw a Card

Draw a card from the deck

Task 4: User input

Ask the user if the next card is going to be higher or lower

Task 5: Putting it all together

If the choice is wrong the game ends and your score is set, if you get it right you keep going.