

Random class

<https://csci-1301.github.io/about#authors>

June 23, 2021 (09:31:25 PM)

Contents

| | | |
|---|--|---|
| 1 | Manipulating Two Arrays at the Same Time | 1 |
| 2 | A Guessing Game | 2 |

1 Manipulating Two Arrays at the Same Time

This problem combines random number generation with arrays. The `Random` class from the C# standard library can be used to generate random numbers in any given range. Here is an example of how to use it to generate 15 random numbers:

```
Random myRandomObject = new Random();    //Instantiate a Random object
int randNum, i = 0;
while (i < 15)
{
    randNum = myRandomObject.Next(1, 11); //Generate a random number between 1 and 10
    Console.WriteLine(randNum);           //Display the random number
    i++;
}
```

Using a `Random` object like this, write a program that

1. declares two arrays of `int` of size 8,
2. initializes the values of the first array with random numbers between 0 and 9,
3. initializes the values of second array with random numbers between 0 and 9,
4. displays the contents of the two arrays in a table, and, for each index, a letter indicating whether the first array “won” or “lost” a contest with the second array: “W” if the value in the first array is greater than the value in the second array, “T” if they are equal, and “L” if it is lesser.

An example execution of this program would display:

| | | |
|---|---|---|
| 0 | 8 | L |
| 5 | 3 | W |
| 3 | 3 | T |
| 1 | 2 | L |
| 3 | 1 | W |
| 9 | 0 | W |
| 9 | 0 | W |
| 1 | 5 | L |

In this example, the first array contains “0 5 3 1 3 9 9 1” and the second contains “8 3 3 2 1 0 0 5”.

2 A Guessing Game

Write a program that

1. Stores a random number in a variable.
2. Asks the user to enter a numerical value, and stores the user’s answer in a variable.
3. With an **if** statement, displays on the screen “You guessed correctly” if the number entered by the user is your favorite number.
4. Once this part of the program works, add an **if** statement that displays on the screen “Too high!” if the number entered by the user is strictly greater than your favorite number.
5. Once this part of the program works, add an **if** statement that displays on the screen “Too low!” if the number entered by the user is strictly lower than your favorite number.
6. Once this part of the program works, add an **if** statement that displays on the screen “You found a multiple of my favorite number!” if the number entered by the user is a multiple of your favorite number, but different from it.

You can adjust your program so that e.g. if the user enters a number that is at the same time higher and a multiple of your favorite number, only one message is displayed.

A solution is proposed in this archive¹.

¹GuessingGame.zip