

CSCI 1301 – Lab 20

October 24, 2018

1 User Input Validation

1.1 Integer Validation

Consider the code we just studied:

```
Console.WriteLine("Please enter a positive number");
int n = int.Parse(Console.ReadLine());
while (n < 0)
{
    Console.WriteLine($"You entered {n}, I asked you for a positive number. Please try
    ↪ again.");
    n = int.Parse(Console.ReadLine());
}
```

1. As always, start by creating a blank project, copy-and-paste that “snippet” into the `Main` method, compile it and execute it.
2. Then, copy and, comment it out, and adapt your copy so that the user will be asked to enter an integer between 0 and 100, and asked again as long as (s)he does not comply.
3. Re-do the previous step, but change the condition, so that the user has to enter an even number.

1.2 String Validation

Adapt the code above to perform string validation: ask the user to enter a string, and as long as the user does not enter “Yes” or “No”, ask him/her again to enter a value.

2 First While Loops

1. Write a `while` loop that display the “*” character 100 times at the screen.
2. Modify your previous loop, so that a new line character is displayed on the screen every time 10 “*” has been displayed on the screen. That is, your program should display on the screen:

```
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

3 TryParse Method

3.1 Getting Familiar With It

Consider the code we just studied:

```
Console.WriteLine("Please, enter an integer.");
string message = Console.ReadLine();
int a;
bool res = int.TryParse(message, out a);
if (res)
{
    Console.WriteLine($"The value entered was an integer: {a}.");
}
else
{
    Console.WriteLine("The value entered was not an integer, so 0 is assigned to
↪ a.");
}
Console.WriteLine(a);
```

What happen if:

- The user enters an integer?
- The user enters a floating-point value?
- The user enters nothing?
- The user enters a string that contains alphabetical characters?

3.2 Using It

1. Write a code that ask the user to enter an integer, and ask the user again as long as the user entered something that isn't an integer.
2. Actually, there is a `TryParse` method in other classes as well: there is for instance a `Double.TryParse` and a `Decimal.TryParse` method. Write a small program that uses one of them.