Introduction to Data Types and Variables in C#

When we write programs, we’re telling the computer how to process pieces of information, like the numbers in a calculation, or printing text to the screen. So how does a computer distinguish between a number and a character? Why is it important to be able to tell the difference?

Languages like C# tell a computer about the type of data in its program using *data types*. Data types represent the different types of information that we can use in our programs and how they should be used.

Without data types, computers would try and perform processes that are impossible, like squaring a piece of text or capitalizing a number. That’s how we get bugs!

C# is *strongly-typed*, so it requires us to specify the data types that we’re using. It is also *statically-typed*, which means it will check that we used the correct types before the program even runs. Both language features are important because they help write scalable code with fewer bugs.

Using types will come up in several different places when learning C#. To start, we’ll examine how types impact variable declaration and the usage of different data types in a program.

In this lesson, we’ll look at:

- Common C# data types
- How to create variables that are statically typed
- How to convert variables from one data type to another

☑Instructions

Computers can do different things with different kinds of data. This computer will process data according to the function that you give it. If the function matches the data type, you will get an answer! If it doesn’t, you’ll see an error.

The functions do the following:

capitalize: will turn lowercase characters into uppercase characters

square: will square a number

evaluate: will determine if an input is true or false

The data types include:

int (4637): whole integer number

string (kangaroo): a piece of text

bool (true): represents the logical idea of true or false