

# Introduction

1 minute

Many applications you'll build in C# will require you to work with data. Sometimes that data will be hard-coded in your application. For example, you may need to print a message to the user when some operation succeeds. A "success" message would likely be the same every time the application is executed.

But more often, you'll work with data that is unknown as you build your application. Perhaps the data is captured via keyboard input, or comes from a file or the internet. Therefore, you'll need to build structures called variables that can store data in memory as your application runs.

Suppose you want to display a formatted message to the end user containing different types of data. The message would include hard-coded strings, but may combine it with information your application collects from the user. To display a formatted message, you'll need to create both hard-coded values and define variables that can store data of a certain type, whether numeric, alphanumeric, and so on.

In this module, you'll create hard-coded literal values that contain different data types. You'll create variables that can hold certain data types, set those variables with a value, then retrieve those values later in code. And finally, you'll learn how you can simplify your code by allowing the compiler to shoulder some of the work.

By the end of this module, you'll be able to create literal values and store and retrieve data in variables.

## Learning objectives

In this module, you will:

- Create literal values for five basic data types
- Declare and initialize variables
- Retrieve and set values in variables
- Allow the compiler to determine the data type for your variable when initializing

# Prerequisites

- Must understand basic C# syntax rules
  - Must understand how to use `Console.WriteLine()`
- 

## Next unit: Exercise - Literal values

Continue >

---

How are we doing? ☆ ☆ ☆ ☆ ☆