Creating Variables with Types

When we use data in our programs, it's good practice to save them in a *variable*. A variable is basically like a box in our computer memory where we can store values used in our code.

In C#, data types and variables are closely intertwined. Remember how C# is strongly-typed? Every time we declare a variable, we have to specify what kind of data type that variable is going to hold.

There are two ways we can assign variables. We can do it on two lines:

```
// Declare an integer
int myAge;
myAge = 32;
```

Or, we can be more concise and just do it on one:

```
// Declare a string
string countryName = "Netherlands";
```

In each case, we first write the data type, then the variable name, then use the equals sign `=` to assign the variable a value.

Once we've defined a variable, we can use them throughout our program. For example, here's a short program that prints a few math equations to the console:

```
int evenNumber = 22;
int oddNumber = 45;
Console.WriteLine(evenNumber + oddNumber); // Prints 67
Console.WriteLine(oddNumber - evenNumber); // Prints 23
```

If we want to change the values, it's only necessary to change it in one place instead of everywhere it is used.

☑Instructions

1.

To practice creating variables, we're going to write a program that prints information about a dog to the console. We'll be working wth the types `string`, `int`, `double`, and `bool`.

First, create two string variables. The first one is called `name` and has the value `"Shadow"`. The second one is called `breed` and has the value `"Golden Retriever"`.

✓

Hint

The `string` data type is used to represent text. Since these variables contain text, use a `string` data type:

```
string book = "Americanah";
```

2.

Next, create a variable to hold the age. Name the variable `age` and store the value `5`.

✓

Hint

The `int` data type is used to represent whole numbers. Since this value is a whole number, use an `int` data type:

```
int count = 10;
```

3.

Next, create a variable to hold the weight. Name the variable `weight` and store the value `65.22`.

✓

Hint

The `double` data type is used to represent decimal values. Since the value of this variable is not a whole number, use a `double` data type:

```
double distance = 26.2;
```

4.

Next, create a variable that can be either true or false. Name the variable `spayed` and set it to `true`.

✓

Hint

The `bool` data type represents the concept of true or false. Use a `bool` data type for this variable:

```
bool isItOn = false;
```

5.

Use `Console.WriteLine()` to print each variable to the console.

✓

Hint

To print a value to the console:

```
Console.WriteLine(variableName);
```