Table of Contents

Introduction

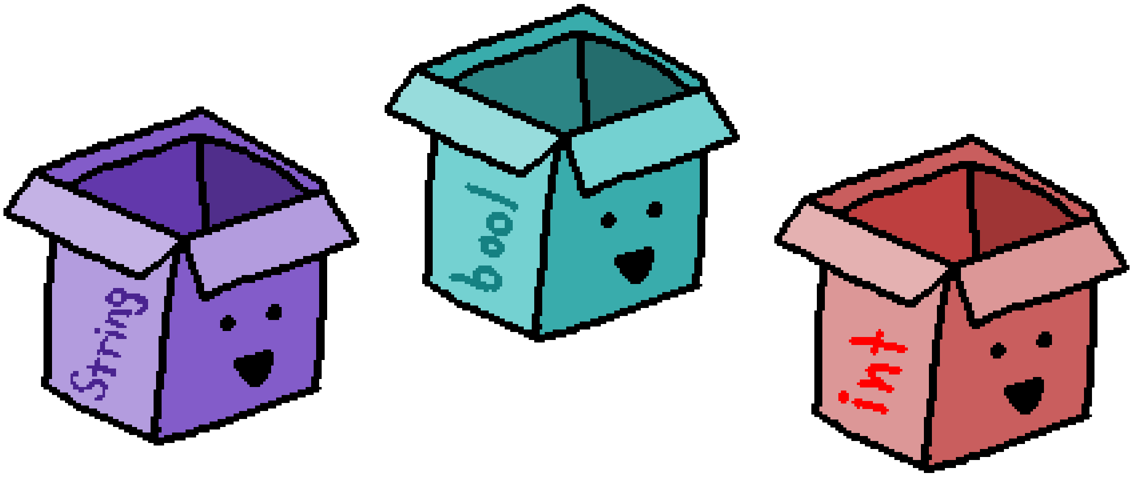
Remember in math we used to have equations that looked like this:

$$v = \frac{d}{t}$$

In programming, we have variables as well. They can store different *types* of data (not just numbers), and they can be given longer names than simply *v*, *d*, or *t*.

When we create a variable, we have to specify what **data type** it is (Integer? Number with a decimal? Text?), and we have to specify the **variable name** that we're giving it.

Variables are an important part of programming, so make sure you learn the appropriate vocabulary to discuss them with!



Vocabulary

It is important to be acquainted with programming terminology in order to effectively communicate with other programmers, as well as your instructor in class. Make sure you know these terms.

Data Types	Variable Declaration	Variable Name	Initialize

Data Types

In math, variables just store numbers. In programming, variables can store any type of data, which is why there are different **data types**.

The most common data types we will use are:

Integer

An integer is a whole number (no fraction or decimal), and can be positive, negative or zero.

C++/Java/C# Example:

```
int a = 2;
int b = 100;
int c = -5000;
```

Double, Float

Doubles and floats are numbers that contain a decimal point. These can also be positive, negative, or 0.0.

C++/Java/C# Example:

```
double price = 9.99;
float velocity = -9.8;
double price = 9.99;
float velocity = -9.8;
```

Character

You can store a single character in a char data type. A character can be a number, a letter, a symbol, etc. Character values must be contained within single-quotes.

C++/Java/C# Example:

```
char direction = 'N';
char moneyType = '$';
char space = ' ';
```

String

A string can contain a series of characters. Within a string, you can store words, sentences, text, etc. String literals must be contained within double-quotes.

C++/C# Example:

```
string name = "Ada";
string noun = "Cat";
string label = "Amount: 3";
```

Java Example:

```
String name = "Ada";
String noun = "Cat";
String label = "Amount: 3";
```

Boolean

*A boolean can store either true or false. We can use it to ask questions within the program, to decide what path to take. Note that **true** and **false** are keywords, so you simply type out these words for values. You do not put them within double-quotes like a string.*

C++/C# Example:

```
bool isDone = true;
bool votingAge = false;
```

Java Example:

```
boolean isDone = true;
boolean votingAge = false;
```

Variable Declaration

In C++, Java, and C# (as well as some other programming languages), you must **declare** a variable before you can begin using it.

At minimum, a variable declaration must include the **data type** and **variable name**:

```
int totalPizzas;
```

int is the data type, *totalPizzas* is the variable name.

You can also declare multiple variables at once, as long as they're of the same data type:

```
int studentCount, classCount;
```

When declaring a variable, you can also initialize the variable to some value:

```
int totalPizzas = 10;
```

And if you're declaring multiple variables, you can initialize each one:

```
int studentCount = 20, classCount = 5;
```

Once a variable is **declared**, you can use it by-name throughout your program, without reiterating its data type. In the following code, line 3 references previously declared variables.

```
int totalPizzas = 10;
int people = 2;
int pizzasPerPerson = totalPizzas / people;
```

What is the value of **pizzasPerPerson** at the end of this snippet of code?

Valid variable names

Variable names in C++, Java, and C# must follow these rules:

1. A variable must begin with a letter (a-z, A-Z), underscore (_), or dollar sign ($).
2. You can include numbers in a variable name (0-9), but not as the first character.
3. The variable name can have virtually any length.
4. You can use upper case and lower case variables, but the convention is to only name final/const variables in all-upper-case. Remember that variable names are case-sensitive.
5. Reserved keywords cannot be used as variable names. (example, **for**, **if**, or **class**)

Assigning variables to values

When assigning a value to a variable, the variable name **must** be on the left-hand side of the expression (also known as “LHS” for short), and the value or computation **must** be on the right-hand side (RHS).

Variable declarations with assignments:
1. Data-type
2. Variable name
3. Equal sign
4. Value

```
String food = "Taco";
int quantity = 10;
double pricePerTaco = 1.24;
```

You can assign values to variables that have already been declared or overwrite variables that already have values.

```
String food = "Churro";
food = "Bibim Bap";
food = "Rasgulla";
```

You can do computations and assign them to variables, but the computations must be on the RHS.

```
double pi = 3.14159;
double twoPi = 2 * pi;
int sum = 5 + 6 + 7;
double price = 9.99 + tax;
```