

PRACTICE 4: MORE VARIABLES



Using Variables to Measure Something

Build on our last lesson with more variables! You'll create a tool that measures how many of something it takes to reach a destination. Here's a preview of what you'll build:

```
What is your height in feet?
5.41
It would take 5778401922.365988909426987061 of you stacked on top of one another to reach the moon!
Press any key to continue...
```

Variable Assignment

You can use the values that variables contain by typing the name of the variable.

```
Cats = 5
```

```
BoxesNeeded = Cats*2
```

First, the value 5 is assigned to the variable Cats.

For the second line, the right-hand side is evaluated first: take the value of Cats (5) and multiply it by two. Then, assign this value to the variable **BoxesNeeded**.

So what's the value of **BoxesNeeded**? Use `TextWindow.WriteLine(BoxesNeeded)` in Small Basic to find out!

Now try reading this code sample:



Note: the right-hand side of the equation is evaluated first

```
Length = 4
```

```
Width = 10
```

```
Area = Length*Width
```

Here, Area ends up with the value 40: first, you take the values of Length and Width, you multiply them, then you assign the result to Area.

Now try this next code sample. What value do you think Area ends up with?

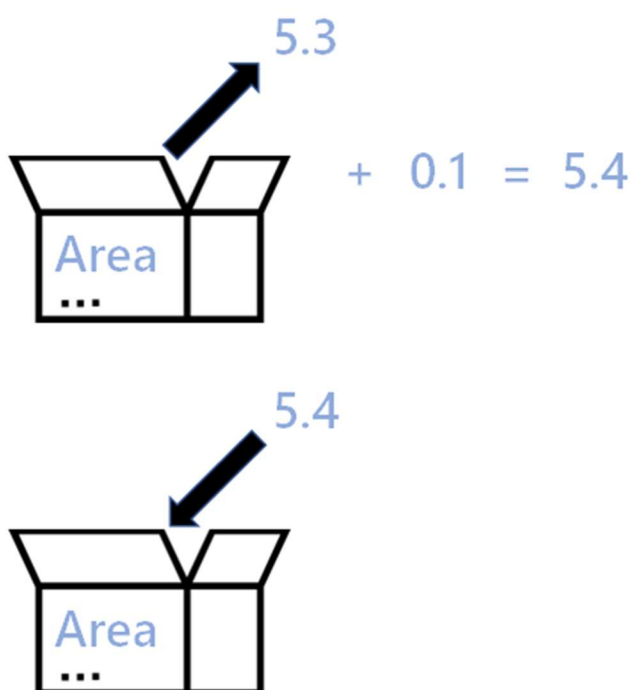
```
Area = 5.3
```



`Area = Area+0.1`

If you said 5.4, you are right.

You take the value of Area, which is 5.3, add 0.1 to it, then assign this value back to Area.



It might be confusing to see the equals sign, but keep in mind that it means assignment, not equality. So the last statement isn't saying that Area is equal to `Area+0.1`, but that Area's new value is now Area's old value + 0.1.

Naming Variables

You can name your variables anything you want:

`Tacos = 2.54*Hamburgers`

But it makes things easier to read (especially for future you) if you name things well:

`Centimeters = 2.54*Inches`



Challenge: Build a Measurer

We're now going to build a measurer. First, ask for the height of something using `TextWindow.Read()`: it could be your height, the height of an animal, or the length of a bus. To keep it less complicated (for you), ask for the height or length in inches.

Next choose a long distance: maybe the distance from your house to your school, from New York to London, or from Earth to the Moon. You might have to look that up. Finally, output how many of that object it would take to reach the long distance.

A sample of what your program might look like after you run it can be found at the beginning of this lesson.

You could also try other measurements: how many cups of water would it take to fill an Olympic sized pool (using volume), how many of you would it take to be as heavy as an elephant (using weight), or anything else you can come up with!

If you want to try a more complicated formula, you could build a converter, such as a Fahrenheit to Kelvin converter.

Tip: You might not want a long trailing decimal point. To round a number to the nearest whole number, type `Math.Round(number)`. To round down, type `Math.Floor(number)`. To round up, type `Math.Ceiling(number)`. The floor is down and the ceiling is up!

Challenge 2: Solve a Math Formula

Pick a math formula and write a program to solve it. If you need ideas, there's a list of basic formulas in the resources. Use one `TextWindow.Read()` statement for each piece of information you need to solve the formula, then output the answer.

For example, write a program that calculates the perimeter of a triangle by adding side a plus side b plus side c. You'll need to ask for a, b, and c.



Tip: You might want to print out the formula before asking for values a, b, and c, so they know what types of values to enter each time!

Discussion Questions

1. Look at your life today. What is something silly that you could measure? Examples: How many times someone looked at your eyes when you were talking to them. How many times a cheerleader repeated an overused word. How long it takes for a student to leave a classroom when the bell rings.
2. Give a real-life situation where you could code using variables.
3. How would you write a program using 3 variables that gives the result of 26.7?
4. Now that you know about variables, how do you think calculators work?
5. When would be a good time to use a rounded number and when would be a good time to use decimal points?
6. TextWindow has Read and ReadNumber. Use ReadNumber, but try typing in a word. What happens? Why do you think that is?

Additional Resources

Basic Math Formulas

<https://www.basic-mathematics.com/basic-math-formulas.html>

Small Basic: Variable

<https://social.technet.microsoft.com/wiki/contents/articles/24151.small-basic-variable.aspx>

Small Basic Getting Started: Introducing Variables

<https://social.technet.microsoft.com/wiki/contents/articles/16063.small-basic-getting-started-guide-chapter-3-introducing-variables.aspx>

Small Basic TextWindow Object

<https://social.technet.microsoft.com/wiki/contents/articles/23724.small-basic-reference-documentation-textwindow-object.aspx>

