

Lesson 1: Variables Explore

Overview

To begin the lesson students explore sample apps similar to the ones they'll be able to build by the end of the unit. Then students complete an unplugged activity with plastic baggies and sticky notes to build a mental model of how variables are used to move and store information. The lesson ends with a synthesizing discussion and students adding key vocabulary to their journal.

Purpose

The warm up activity is designed to provide context for the coming unit and motivate the reasons students will want to learn the concepts covered in Unit 4. The subsequent activity provides students a physical mental model they will be able to use when they start programming with variables in the subsequent lessons.

Standards

Full Course Alignment

CSP Conceptual Framework

- ▶ **AAP-1** - To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways.
- ▶ **AAP-2** - The way statements are sequenced and combined in a program determines the computed result. Programs incorporate iteration and selection constructs to represent repetition and make decisions to handle varied input values.
- ▶ **DAT-1** - The way that the computer represents data is different from the way that the data are interpreted and displayed for the user. Programs are used to translate data into a representation that is more easily understood by people.

CSTA K-12 Computer Science Standards (2017)

- ▶ **AP** - Algorithms & Programming

Agenda

Lesson Modifications

Set Up

Warm Up (5 minutes)

Objectives

Students will be able to:

- Evaluate expressions that include numbers, strings, and arithmetic operators.
- Trace simple programs that use variables, expressions, and variable assignment.
- Use appropriate vocabulary to describe variables, expressions, and variable assignment.

Preparation

- Collect for each pair of students:
 - 3 sandwich baggies
 - packs of red and yellow stickies
 - pens / pencils
 - 1 dry erase marker per four students (pairs can share)
- Review the sample apps shown in the warm up
- Review the rules and vocabulary used in the slides

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the teachers

- **CSP Unit 4 - Variables, Conditionals, and Functions** - Slides
- **EIPM: A Short Introduction** - Resource
- **Guide to EIPM Lessons** - Video
- **Guide to Explore Lessons** - Video

Explore Sample Apps

Sample Apps

Preview the Unit

Activity (30 minutes)

Variables

Wrap Up (10 minutes)

Check For Understanding

Teaching Guide

Lesson Modifications



Attention, teachers! If you are teaching virtually or in a socially-distanced classroom, please read the full lesson plan below, then click **here** to access the modifications.

Set Up

This is the first official "Explore" lesson in the EIPM model. Review the EIPM model in the **EIPM One Pager**.

Explore Lessons:

Overview: Students explore the new concept through a teacher-led hands-on group activity.

- Typically uses physical manipulatives
- Teacher leads with the support of slides and activity guides

Goal: Students begin to develop a shared mental model and understand the main ideas of the new concept.

Explore



Watch the following videos:


- **Guide to EIPM Lessons**
- **Guide to Explore Lessons**

Warm Up (5 minutes)

Explore Sample Apps

Group: Place students in pairs

Sample Apps

 **Do This:** Give students 3-5 minutes to explore sample apps that they'll find on Code.org at the beginning of this lesson (in Levels 1-3). Students can explore one or more depending on time.




1-3

Sample Apps

1

2

3

 **Discuss:** *These are samples of the kinds of apps you'll be able to build by the end of this unit. As you go through them, write down at least two examples where the app seems to be keeping track of a piece of information or using it to make decisions.*

Discussion Goal: Have volunteers share what they noticed with the room. Some ideas include:

- The Pet Rock App keeps track of clicks and uses it to decide when the pet rock will "evolve"
- The Poem App keeps track of the poem as you write it
- The Thermostat App keeps track of temperature and uses it to change the color of the text

Preview the Unit

Remarks

Last unit we built apps that mostly focus on input and output. For example

- When button clicked → show picture
- When button clicked → change screen
- When button clicked → play a sound

But we all want our programs to do more than that! In this unit we're going to learn how to build apps that keep track of information and use to make decisions and perform calculations. This is going to let us build much more powerful apps! So let's get started by diving in deep on what's involved with programming with information.

Activity (30 minutes)

Variables

Group: Group students in pairs.

Distribute: Give each pair of students:

- A small stack of red and yellow sticky notes
- A pen / pencil
- 3 plastic baggies
- A dry erase marker to share with another group

 Teaching Tip



Supplies Substitutions: There's no need to use stickie notes if you have other scraps of colored paper. Also consider cutting stickies in 4 to make them go further. If you don't have dry erase markers handy consider using pieces of masking tape on the baggies.

Display: Use the activity slides for this lesson to guide the unplugged activity on Variables.

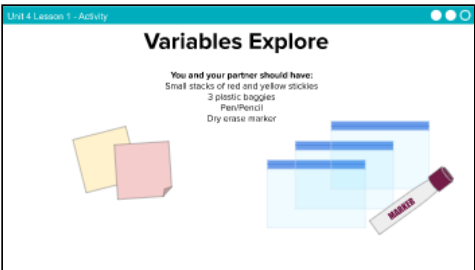
Teaching Tip

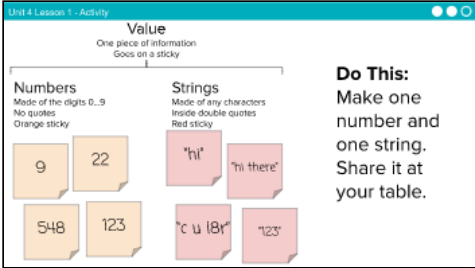
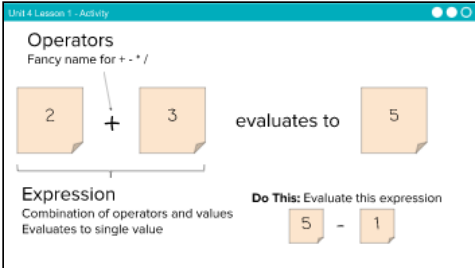
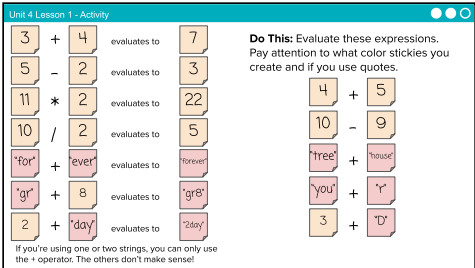
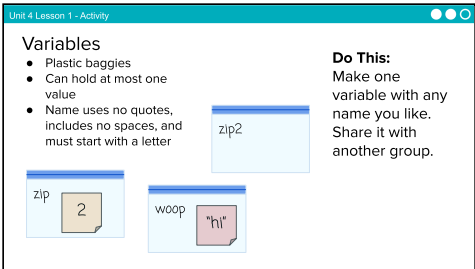
Running the Activity: This activity asks students to follow along as a number of core concepts for programming are introduced. The model is typically that a term or concept is introduced and modeled and then afterwards students are encouraged to try it out on their own. Trying it out typically means they are writing information on a sticky note and sharing it with another group before discussing the results with the whole class.

Slides with animations have an icon in the bottom left corner to let you know you need to click to reveal more of the slide's content.

To help you more easily prepare the activity and keep track of your instructions, detailed instructions have been included as speaker notes in the presentation. Here are some tips to help you throughout the presentation.

- There are opportunities throughout the presentation for students to actively engage. At these moments students should be making things with their manipulatives or using them to answer questions. Use these opportunities to check progress.
- There is a fair amount of new vocabulary introduced but it is introduced gradually and with intentional repetition. Make a point of actively modeling the use of new terms.
- The most important goal here is building a mental model. It is ok if students have some open questions that will get resolved over the subsequent conditional lessons.
- Both you and students can use the "Key Takeaways" to check your understanding at the end.

Slides	Speaker Notes
	<p>Say: Today we are going to explore Variables.</p>

Slides	Speaker Notes
 <p>Value One piece of information Goes on a sticky</p> <p>Numbers Made of the digits 0-9 No quotes Orange sticky</p> <p>Strings Made of any characters Inside double quotes Red sticky</p> <p>Do This: Make one number and one string. Share it at your table.</p>	<p>Say: We're going to be thinking about how computers work with information. We're going to call one "piece" of information a "value". Right now there are two different types of values: numbers and strings. There are a few ways to tell them apart. Numbers work the way you normally think of numbers. They are made of the digits 0 through 9. We are going to put numbers on yellow sticky notes. When we write numbers you don't need quotes. Strings are made of any characters you can see on the keyboard which means all these examples count as strings. Strings go inside double quotes. You can see how it would be important to tell the number 123 apart from the string "123".</p> <p>Do This: Make one string and one number and hold it up at your table.</p>
 <p>Operators Fancy name for + - * /</p> <p>Expression Combination of operators and values Evaluates to single value</p> <p>Do This: Evaluate this expression</p>	<p>Say: Operator is just a fancy name for the plus, minus, multiply, and divide symbols. An expression is a combination of two sticky notes and an operator. When I ask you "what's 2 plus 3" you say "5"! You just "evaluated" the expression 2+3, or figure out what the value is. Since 5 is a number, we put it on a yellow sticky. When evaluating an expression, we follow order of operations - just like in math class.</p> <p>Do This: Evaluate the expression 5-1 and make a sticky note. Make sure you pick the correct color and decide if you need quotes or not!</p>
 <p>Do This: Evaluate these expressions. Pay attention to what color stickies you create and if you use quotes.</p>	<p>Say: This table shows the different ways we can create expressions. We can use all four operators with numbers the normal way. When you want to use strings, you can only use the + operator and it connects the two words together. If you're connecting a number and a string, the number first gets converted to a string.</p> <p>Do This: With your partner make a sticky note for each of these 5 expressions and we'll share as a class.</p> <p>Click for animation: Click through to see the answers.</p>
 <p>Variables</p> <ul style="list-style-type: none"> Plastic baggies Can hold at most one value Name uses no quotes, includes no spaces, and must start with a letter <p>Do This: Make one variable with any name you like. Share it with another group.</p>	<p>Say: We're going to call the plastic baggies on your table "variables". Variables can hold at most one value, or sticky note. They have names that use no quotes, include no spaces, and must start with a letter. For now just practice making a variable with your partner.</p> <p>Do This: Make one variable with any name you like. Share it with another group. Make sure you use a whiteboard marker so we can reuse the baggies later.</p>

Slides

Speaker Notes

Unit 4 Lesson 1 - Activity

Variables and Expressions

Replace variable name with a copy of the value it holds
Evaluate the expression as normal

Say: We can evaluate expressions that include variable names. To do that, first make a copy of the sticky note inside the variable. Then evaluate the expression the way you normally would. These two examples show you how.



****Click for animation: **** Click through to see the answers.

Unit 4 Lesson 1 - Activity

Do This: Evaluate these expressions. Make sure you pay attention to whether it evaluates to a string or a number.

Do This: Evaluate these expressions. Make sure to pay attention to whether it evaluates to a string or a number.

Note: Have students share their answers by holding up the sticky notes for each solution.



****Click for animation: **** Click through to see the answers.

Unit 4 Lesson 1 - Activity

Let's start writing programs that control our variables.

We're going to stop using stickies but will highlight **strings** and **numbers** to help you remember the difference.

Say: Now let's write programs. We're going to stop using sticky notes, but will highlight strings and numbers to help you remember the difference.

Unit 4 Lesson 1 - Activity

var

Creates a new variable
Grab a new baggie
Write the variable's name on the baggie

Say: Here's what a program looks like. The var command tells the computer to create a variable.

Unit 4 Lesson 1 - Activity

Do This: Run this program

← "Assignment operator"

"Assign": a fancy name for putting a value inside the baggie.

Variables can only hold one stickie. If there's already a sticky note in there, throw it away.

"pow gets 3" and "pow gets 5"

Say: The left arrow is called the "assignment operator". That's just a fancy word for "put this value in the baggy". If we wanted to read line 01 we would say "pow gets 3". We know that variables can only hold one sticky note or value. So if we try to assign a variable that already has a value in it, we just throw the old one away.



Click for animation: ** Click through to run the program.

****Say:** In a computer you don't actually put the old number in a trash can, it's totally deleted! The numbers are stored as electrical charges somewhere in your computer's memory. When you assign a new number to a variable it actually erases the old number, and stores the new number in its place.

Slides

Speaker Notes

Unit 4 Lesson 1 - Activity

Do This:
Run this program. Compare your result with another group.

```
00 var pizza
01 pizza ← 3
02 var tacos
03 pizza ← "yum"
04 tacos ← "the best"
```

Do This: Run this program. Compare your results with another group.

Note: Walk around the room making sure students are following the rules correctly. No baggies should have more than one sticky note in them.



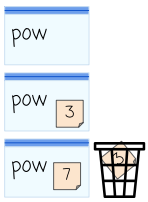
**Click for animation: ** Click through to see the answers.

Unit 4 Lesson 1 - Activity

Assign a Variable with Expression
Evaluate the expression first to get one value.
Assign the value as normal

```
00 var pow
01 pow ← 1 + 2
02 pow ← 3 + 4
```

Evaluate expression first



Say: Now let's combine what we've learned. You can use assignment with expressions. In order for this to work you need to evaluate first, then assign. This makes sense because we know we can only put one sticky note in the variable baggy.

Unit 4 Lesson 1 - Activity

Do This:
Run this program. Compare your result with another group.

```
00 var zow
01 var fly
02 fly ← "to" + "day"
03 zow ← 4 - 1
04 fly ← 3 * 3
05 zow ← 4 + "now"
```

Do This: Run the program. Compare your results with another group.

Note: Walk around the room making sure students are evaluating before they assign. Reinforce the language "gets" for the assignment operator. For example, line 02 would read "fly gets two plus day".



**Click for animation: ** Click through to see the answers.

Unit 4 Lesson 1 - Activity

We're not going to highlight our strings and numbers anymore. We can just use double quotes to tell the difference.

Say: We're not going to highlight our strings and numbers anymore. We can just use double quotes around the strings to tell the difference.

Slides

Speaker Notes

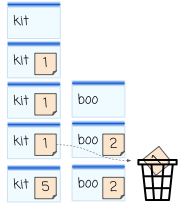
Unit 4 Lesson 1 - Activity

Assign a Variable: Expressions with Variables

Evaluate the expression on the right first to get one value.
Assign the value as normal

```
00 var kit
01 kit ← 1
02 var boo
03 boo ← kit + 1
04 kit ← 5
```

Note: Variables aren't "connected". Changing kit doesn't change boo.



Do This: Run through this program together as a class.

Note: Reinforce:

Evaluate, then assign

As shown on line 04, there's no special "connection" made between variables. All we're doing is moving information around.

Variables only hold one value, the old one is "thrown away" (again, what's happening is that assignment replaces the electric charges somewhere inside the computer so the old value is actually "erased" or "deleted")




**Click for animation: ** Click through to run the program.

Unit 4 Lesson 1 - Activity

Do This:
Run this program. Compare your result with another group.

```
00 var fuzz
01 var clip
02 fuzz ← 5
03 clip ← fuzz + 2
04 fuzz ← clip + 1
05 clip ← "gr" + fuzz
06 fuzz ← fuzz + 1
07 fuzz ← fuzz + 1
08 fuzz ← fuzz + 1
```



Do This: Run this program. Compare your results with another group.

Note: Have students hold up their two baggies when they're done. Make sure students are evaluating and then assigning. Call out that last three lines which can be a little nutty to think about. You're using a variable's value as part of the assignment. This lets it "count up by one".

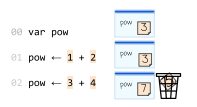


**Click for animation: ** Click through to see the answers.

Unit 4 Lesson 1 - Activity

Key Takeaways

- Numbers and strings are two different types of values
- Expressions evaluate to a single new value
- When variables are in the expression just make a copy, don't change the actual variable.
- Variables are "assigned" a new value
- Evaluate first, then assign
- Old values are deleted forever.
- Assignment just moves information around. It does not "connect" variables.



Do This: Review the key takeaways with students.

Unit 4 Lesson 1 - Activity

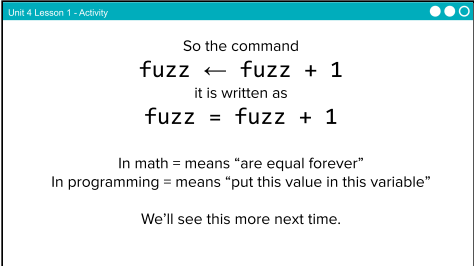
In some languages (including Javascript) the assignment operator is not written

←


it is written as

=

Say: In some programming languages the assignment operator is not written with an arrow but is written as an equal sign.

Slides	Speaker Notes
	<p>Say: Here's how the assignment operator looks in Javascript. We'll see more of this in the next lesson!</p> <p>Note: In math = means "are equal forever". In programming = means "put this value in this variable".</p>

Wrap Up (10 minutes)

 **Journal:** Have students add the following words to their journals: Expression, Variable, Assignment Operator.

- **Expression:** a combination of operators and values that evaluates to a single value.
- **Variable** an abstraction inside the program that can hold a value. Each variable has associated data storage that represents one value at a time.
- **Assignment operator:** allows a program to change the value represented by a variable.

Remarks

This isn't the last time we'll look at these definitions and it's fine to update them as you get more experience. In fact, I'm sure you'll have a lot more to add once we see all of these concepts used inside of apps when we meet next time!

Check For Understanding

Check For Understanding Question(s) and solutions can be found in each lesson on Code Studio. These questions can be used for an exit ticket.

Question: What will the value of score be at the end of the program?

```
var score
score <- 3
score <- score + 1
score <- "The score is: " + score
```

 4

 Check For Understanding