

# Lesson 5

## Add-ons

### Pre-Lesson Ideas:

#### Move an ellipse:

- \* Have younger students only draw an ellipse. Then, have them define one variable,  $x$ . Set  $x$  equal to the number they have for the  $x$  coordinate in their ellipse command.
- \* Use one math expression to add one to  $x$ .
- \* There should be a trail of ellipses. Use this visualization to explain to students that the computer draws a new ellipse at 1 plus wherever the  $x$  coordinate was.

### Post-Lesson Ideas:

#### Reflection Questions

- 1) What did you do with variables?

Possible Answers: I made my vehicle move at an angle.

- 2) What was most difficult about learning to use variables?

Possible Answers: It was hard to remember to define my variables before the draw function.

- 3) Is there an error or problem you keep running into?

Possible Answer: I keep forgetting to add my variables to every shape command. As a result, some shapes get left behind and my vehicle gets messed up.

#### Further Development

- \* Have students try to get another vehicle to move on the same screen. Challenge them to have the vehicle move in the opposite direction. This would require two new variables and a math expression that uses subtraction, instead of addition.

# Lesson 5

## Set Up

### Vehicles that Move:

- \* Learn how to declare and use variables in order to create values that can be changed.
- \* Manipulate variables with math expressions to move shapes.

### Project Goal:

1) Generally, what should the project look like?

A vehicle that moves across the screen. **A single shape that moves across the screen.**

2) What skill(s) are being learned/ practiced?

3) What concept are students gaining insight on?

Manipulating variables to move objects.

### Programming/ Math Vocabulary:

**Variable** - A declared symbol that represents a value that can change

**Execute** - To put into effect : perform or carry out. (Run)

**Loop/Looping** - When something occurs more than once

**Math Expression** - **Addition Subtraction Multiplication or Division**

**Declare** - **To define the name and data type of a variable or other programming construct.**

# Lesson 5

## Outline

### Introduction to Topic:

“Today we are going to be (making our vehicles move).

### Project Breakdown:

- 1) Plan out your vehicle
- 2) Program a vehicle in the advanced grid
- 3) Declare variables for the x and y coordinates of your shapes.
- 4) Add the variables to each of the x and y coordinates.
- 5) Use a math expression to change one or all of your variables
- 6) Problem solve and trouble shoot errors

### Example Projects/ Basic Source Code:

```
// define variables for x and y
var x = 0;
var y = 0;
draw = function( ) {
  //draw an ellipse
  fill(0,0,0);
  ellipse(x+50, y+300, 50, 50);
  //use math expression to add 5 to the x and y coordinates
  x = x + 5;
  y = y + 5;
};
```

# Lesson 5

## Troubleshooting

### Common Mistakes and Confusions:

#### 1) Declaring variables inside of the draw function

This causes the variable value to be reset to the original value everytime the function is looped through.

#### 2) Using names for variables that are already defined in JavaScript

You should not create variable names like size.

#### 3) Putting code outside of the draw function

Anything you want to change has to be inside of the draw function. Generally, everything you draw is written inside of the draw function.

#### 4) Writing math incorrectly

If you want to change one variable, that variable has to equal the same variable plus or minus some number.

### FAQ's:

#### 1) Why do I need to put var in front of my variables?

Even though it works regardless, it is good programming practice to tell the computer the type of thing you are creating. You also open yourself up to more errors in the future.

#### 2) Can I have multiple variables in a single shape or design command?

Yes! You can use the same variable multiple times and/or have many different variables in one command.

#### 3) Why does a math expression cause a variable to change?

The draw function is a loop that loops through all of your code about 60 times per second. Each time the computer loops through you code, the math expression changes the variable. For example, 5 is added to the variable representing the x coordinate.