Lesson 2
Add-ons

Pre-Lesson Ideas:

* Let students do research about different logos.

* Use printed grid paper and have students draw and label a sketch of what they want to draw.

* Then, require them to get approval of their hand drawn sketches, before they can start programming.

Post-Lesson Ideas:

Reflection Questions

1) What logo did you pick and why?

Possible Answers: I chose the Windows logo because I wanted a logo with different colors.

2) Did anything unexpected happen? What did you think would happen?

Possible Answers: I tried to use the text command, but at first it did not show up. I thought the text would be black.

3) Does anybody want to share a custom logo? Why did you make this logo?

Possible Answer: I used an E and O combined to represent my initials.

Further Development

* Have students who made company logos create their own unique logos.

* Push students to create unique logos that represent them.

Lesson 2

Outline

Introduction to Topic:

"Today we are going to be (making our own logo/ refining an old project).

Project Breakdown:

1) Plan out your drawing/ do research (use graph paper and label each shape)

2) Program basic shapes

3) Use new commands to add details to the shapes

4) Problem solve and trouble shoot errors

Example Projects/ Basic Source Code:

fill(r, g, b); // percent of red, green, and blue from 0-255

background(r, g, b); // same as fill, but fills in the background with color

stroke(r, g, b); // same as fill, but colors lines

noStroke( ); // takes away the outline of a shape

strokeWeight(#); // changes the thickness of lines

text(" ", x, y); // prints the text that you put into the string (" ") and places it at the x and y coordinated indicated

textSize(#); // adjusts the size of text

Lesson 2
Set Up

The Logo Project:

* Teach students visual programming logic by having them use more advanced design commands to either duplicate or create a logo.

Project Goal:

1) Generally, what should the project look like?

A duplicated or unique logo that is more advanced than just shapes. There should be color and some variation in stroke size or color.

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

Navigating the coordinate grid

Learning basic color and design commands and syntax in JavaScript

Understanding R,G,B logic

3) What concept are students gaining insight on?

Communicating with the computer to translate an idea into a product with code.

How a few design commands can be combined to create a complex and realistic piece of art.

Programming/ Math Vocabulary:

Comment - A piece of specially tagged explanatory text within (a program). It is used to assist other users and organize the code.

String - A set of characters that can also contain spaces and numbers enclosed in quotation marks.

Integer - A whole number that can be either positive, negative, or zero.

Lesson 2
Troubleshooting

Common Mistakes and Confusions:

1)   Misspelling Commands

It is really easy for students to make silly mistakes, when they are working with a subject completely unfamiliar to them. Pay special attention to background, students often forget the g.

2)   Missing capitals where appropriate

 In JavaScript, the second word in commands that are two words long are capitalized.

3)   Putting a number in noStroke( )

noStroke is a function with no parameters, so nothing needs to go inside of the parentheses.

4)   Background covering up their project

Background should be written at the top of all the code.

FAQ's:

1)   Are the commands case sensitive?

Yes! The S in noStroke must be capitalized - same with the W in strokeWeight( );

2)   How does strokeWeight work?

You put a number inside the parenthesis to represent how thick you would like the lines around the shapes. In beginner, you have to use a decimal for this number.

3)   Why wont fill color points?

The stroke command is used to color lines and points.