Lesson 13

Add-ons

Pre-Lesson Ideas:

- The goal of this activity it to have students guide their peers to press an invisible button. First, have students pair up. Let students one be the programmer and have student two be the computer.
- The programmer should draw a rectangle on a piece of graph paper without letting the computer see. Then, set a blank piece of graph paper in front of the computer. The programmer then, needs to verbally use if statements to guide the computer to click the correct space on the blank piece of graph paper.
- Example of what the programmer would say: You have to press in a space where x is greater than 500 and less than 550. The y has to be greater than 50 and less then 300.
- The computer then tries to press the correct space.
- Have the students switch roles and go through the whole process again.

Post-Lesson Ideas:

Reflection Questions

1) What new command did we learn and how does it work?

Possible Answers: mousePressed. It returns true if the mouse is pressed.

2) If you want the button to draw a rectangle that stays on the screen, what method would yield this result?

Possible Answers: Instead of drawing the rectangle inside the if statement for the button, draw the rectangle outside of the if statement with variables for width and height that are originally set to zero. Inside of the if statement, set these two variables equal to some value greater than one.

Further Development

* Try making a button that actually looks like it is 3D and that actually looks like it is being pressed.

Lesson 13

Set Up

On Screen Button:

* Using knowledge of if statements, craft a complex if statement that defines a button being pressed.

Project Goal:

1) Generally, what should the project look like?

A rectangle in the screen that defines a button. This button should initiate some action once it is pressed.

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

3) What concept are students gaining insight on?

Programming/ Math Vocabulary:

Lesson 13

Outline

Introduction to Topic:

“Today we are going to be using conditional statements to make a button. The goal is to have the button initiate some action once it is pressed.

Project Breakdown:

1) Draw a rectangle
2) Write and if statement for the button using mouseX, mouseY, and mouse-Pressed
3) Make something happen once the button is pressed
4) Problem solve and trouble shoot errors

Example Projects/ Basic Source Code:

```
var x = 0;
var speedx = 1;
draw = function( )  {
background(255);
fill(0,0,0);
rect(50, 300, 50, 50);
//write if statement for button
If ( (mouseX > 50 && mouseX < 100) && (mouseY > 300 && mouseY < 350) && mouse-Pressed) {
speedx = 1;
}
//action once button is pressed
ellipse( x, 100, 50, 50);
x = x + speedx;
};
```

Lesson 13

Troubleshooting

Common Mistakes and Confusions:

1) Not closing brackets

Every curly bracket ({ }) has to have and open and close bracket, or else the computer with throw and error.

3) Writing an if statement inside of another one

This is one way to write an if statement to create a button. If you are going to do this, make sure the brackets line up. Also, make sure you want this and not two separate if statements.

FAQ's:

1) Why does the action only happen when the button is pressed?

The action only happens when the if statement is true, which means you would have to hold down the button for it to always be true. To get around this, you can have a variable change when the if statement is true. For example, if you want a shape to move once the button is pressed, you need to write the math expression for movement outside of the if statement. Inside of the if statement is where you change the speed variable to something other than zero.

2) Can I use multiple if statements to make the button instead of one?

Yes! You can, you just have to make sure your syntax is correct.

Example:

```
If ( x > 50 && x < 100 ){
        If ( y > 50 && y < 100 ){
                If ( mousePressed ){
} } }
```