**Lesson 5** Logical Operators: AND/OR

**How can we test different conditions at the same time?**

| **Overview** | |
| --- | --- |
| This lesson builds on students’ knowledge of conditional statements by using logical operators && and | | to create a hover button in p5. | |
| **Lesson Objectives** | |
| Students will be able to   * Use && and | | to create a compound conditional statement * Explain the difference between the logical AND operator and logical OR operator | |
| **Suggested Duration** | |
| One period (45 minutes) | |
| **Blueprint Foundations Student Outcomes (**https://blueprint.cs4all.nyc/outcomes/) | |
| Abstraction  Analyze | **Describe** how I might use patterns to express an idea. |
| Abstraction  Prototype | **Explain characteristics** or patterns that informed a function or an interface I created. |
| Algorithms  Prototype | **Demonstrate** the benefit of using an event, conditional or loop in my prototype. |
| Algorithms  Communicate | **Compare and contrast** how conditionals or loops were used in classmates’ prototypes. |
| Programming  Communicate | **Discuss** what can and cannot be done with a specific set of commands. |
| **Vocabulary** | |
| * **Logical AND - &&**: Joins two or more conditions such that the code inside the statement will only run if they are all true at the same time. * **Logical OR - | |** : Joins two or more conditions such that the code inside the statement will run as long as one of these conditions is true. | |
| **Planning Notes** | |
| * **Student Activity**: Students will work in pairs to create a hover button. Decide on pairs in advance, preassign first roles, and remember to have students switch roles often. * **Exit Slip**: Print and cut [exit slips](https://docs.google.com/document/d/1GLjdjOmiOQC1cAPQZowCmxaCVBksxbt8ckm5O9OG18A/edit?usp=sharing) before the lesson. | |
| **Resources** | |
| * Video tutorial: [Else, and else if, AND and OR](https://www.youtube.com/watch?v=r2S7j54I68c) * Interaction with p5: Section on [Combining relational expressions with && and | |](https://creative-coding.decontextualize.com/interaction/) | |
| **Assessments** | |
| * Assess the **Student Activity**. Check for the ability to:   + Identify appropriate thresholds for mouseX and mouseY   + Use && to create a compound conditional   + Use pair programming to create a prototype and debug their code * Assess the **Wrap Up**. Check for the ability to:   + Identify the difference between && and || operators. | |

| **Do Now:** |
| --- |
| * [**Design Journal**] Project both situation 1 and situation 2 on the board. Ask students:   + In each example, what has to be true for José to be able to ride the roller coaster?   + Explain how AND is different from OR.   **Situation #1**    **Situation #2**     * Have students share their responses with the class. |
| **Discussion: Logical Operators** |
| * Logical AND:   + Display the following if statement, which draws an ellipse when the condition is true:      * + Assuming the canvas is 400 pixels by 400 pixels, have a student come up to the white board and shade in what area the mouse has to be in for the condition to be **true**. The diagram should look something like this:        * + Explain to students that so far, they have used conditionals to test for one threshold at a time based on mouse position. In this example, the threshold is at x = 200.   + Now tell students to imagine they wanted to create a conditional to test if the mouse is in the following area:      * + To check if mouseX is **between** two areas, two different thresholds are needed. A threshold where x = 200 **and** x = 300. This is written in code using the logical AND operator, which is represented by two ampersands:      * + Now demonstrate that when the mouse passes 300, no ellipse will appear. Even though the first condition (mouseX > 200) is still true, **both conditions must be true** for the code to run. |
| **Teacher Demo: Hover Button Pseudocode** |
| * Tell students that they are going to create a button that changes the background color of a sketch when a mouse hovers over it. Share this [starter code](https://editor.p5js.org/mparker/sketches/Ju85DTag8) with students.    + Begin by creating a diagram. As a class, determine the vertical thresholds for mouseX. Then have students work in pairs to determine the horizontal thresholds for mouseY:      * Next, work as a class to create pseudocode for the program. When complete, it should look something like this:   Inside an if statement:  Step 1: Create FOUR conditions/thresholds:   * + - mouseX is greater than the left edge of the rect     - AND mouseX is less than the right edge of the rect     - AND mouseY is greater than the top edge of the rect     - AND mouseY is less than the bottom edge of the rect   Step 2: Change the bgColor variable to a new color |
| **Student Activity: Hover Button - Pair Programming** |
| Hover Button: Code   * Tell students that they will work in pairs to build out the hover button example. Reiterate that they will need to use the && operator to connect four different conditions. * Early finishers should update the sketch to also change the fill of the button when the mouse hovers over it.   Hover Button: Present   * Get a pair to volunteer and present their sketch. If they have only followed the steps outlined in the class pseudocode, then the background will **never** change back to the original color, regardless of mouse position.   + Ask students to turn to a neighbor and discuss how they can change the if statement to make the background turn lavender again. *Answer: We can add an “else” that changes the bgColor variable to lavender.*   + Here is the [solution](https://editor.p5js.org/mparker/sketches/pemyUavQE) for teacher reference.   Hover Button: Adding OR   * Show students the following [example](https://editor.p5js.org/mparker/sketches/abwxH2Dhr) that has two hover buttons.   + Ask: What is repetitive about this code? *Answer: Both buttons change the background to the same color.*      * Explain that in this scenario, the background will turn gold if you hover over button 1 **or** button 2. If you want two different buttons to perform the **same task**, you can combine them using the logical OR operator. Update the code to combine the conditionals using OR (as with this [example](https://editor.p5js.org/mparker/sketches/CUjYvM3_U)):      * Big Takeaway: Stress the following to your students:   + The logical AND **limits** conditionals - there are fewer situations where the code inside the conditionals will run.   + The logical OR **expands** conditionals - there are more opportunities for the code inside the conditionals to run. |
| **Wrap Up** |
| * Distribute the [exit slip](https://docs.google.com/document/d/1GLjdjOmiOQC1cAPQZowCmxaCVBksxbt8ckm5O9OG18A/edit?usp=sharing) and collect them before students leave. |
| **Extension:** |
| * For each canvas diagram, have students work with a partner to come up with a condition that checks if the mouse is in the gray area. Assume all canvases are 600 by 600 pixels.      * [Solution](https://editor.p5js.org/mparker/sketches/5uBJ4d8Gm) for teacher reference |