**If statement with multiple Boolean expressions**:

* Used when we need to have multiple conditions
* Logical OR
  + Made using ||
  + Two vertical bars
  + Pseudocode uses the word OR
  + So either of the expressions can make the IF statement true
  + right way to use this
    - if (number > 10 || number < 1) {
  + need Boolean expression on both sides
* Logical AND
  + Made using &&
  + Two ampersands
  + Pseudocode is AND
  + Both expressions must be true for IF statement to be true
  + Right way
    - If (number > 25 && code === 5) {
* Short Circuiting
  + For OR:
    - If the first part of the IF statement is true, the program doesn’t look at the second part
  + This can be used to set a default value
    - Instead of
      * var defaultState;
        + if(userValue === ""){
* defaultState = "WI";
  + - * + }
  + Use this
    - var defaultState = userValue || "WI";

**Nested IF statements**

* Used to answer more than one question in a series
* It is just adding another IF statement within the previous one
* Can add ELSE to the program as well
* The problem can become rather large and difficult to follow
  + See Examples from Learning Unit 5
* Testing one variable for different values
  + This is done with IF, ELSE IF, ELSE IF, ELSE pattern

**Ternary Conditional**

* Ternary operator is an operator with three operands
* Example
  + (count === 4 ) ? “It’s 4” : “Not 4”;
* This replaces an if/else statement

**The Case Structure**

* This takes in the use of a switch statement
  + Switch (choice) {
    - Case “A”;
      * Do the A things;
      * Break;
    - Default:
      * Do the default;
    - }
* Break
  + Case statements don’t stop when they hit another case
  + It goes onto the next statement
  + So it leads to some nasty bugs
  + Break jumps the program to after the switch statement
* Data types
  + Can use any data type
  + String, Boolean, any
* Just because we can do something doesn't mean we should. Don't use a switch statement to test if something is true or false. Use an IF/Else statement.