Intro to JavaScript Class 2 Reference

Scope: The scope of a variable determines the accessibility of the variables data.
Global and Local Scope: A Global variable is declared outside a function and can be accessed anywhere. A Local scope is only declared within a function and can only be used by that function.

If/Else Statements: used to decide which lines of code to execute based on a condition
```
var age = 28;
if (age >= 16) {
    console.log ('Yay, you can drive!');
} else {
    console.log ('Sorry, but you have ' + (16 - age) +
    ' years until you can drive.');
}
```

Comparison Operators

Example	Name	Result
a == b	Equal	TRUE if a is equal to b (can be different types).
a === b	Identical	TRUE if a is equal to b, and the same type.
a != b	Not equal	TRUE if a is not equal to b (can be different types).
a !== b	Not identical	TRUE if a is not equal to b, or they are not the same type.
a < b	Less than	TRUE if a is strictly less than b.
a > b	Greater than	TRUE if a is strictly greater than b.
a <= b	Less than or equal to	TRUE if a is less than or equal to b.

Logical Operators

Example	Name	Result
a && b	And	TRUE if both a and b are TRUE.
a || b	Or	TRUE if either a or b is TRUE.
! a	Not	TRUE if a is not TRUE.

While Loops: While will repeat the same code over and o**ver until some condition is met.**
```
var bottlesOfBeer = 99;
while (bottlesOfBeer >= 1) {
    console.log (bottlesOfBeer + ' bottles of beer on the wall');
    bottlesOfBeer = bottlesOfBeer - 9;
}
```

For loops: are very similar, but you declare a counter in the statement (see example at bottom of page).
Break Statement: To exit a loop, use the break statement.

Arrays are an **ordered** list of values: var superHeroPowers = ['Telekinesis', 'Agility', 'The Force'];
Change array values by accessing their index: superHeroPowers[0] = 'Reflexes';
Add to an array by pushing values in: superHeroPowers.push('Fighting');
Loop through an array:
```
var rainbowColors = ['Red', 'Orange', 'Yellow', 'Green', 'Blue', 'Violet'];
for (var i = 0; i < rainbowColors.length; i++) {
  console.log(rainbowColors[i]);
}
```