



"If" Statements

The first control-flow statement in JavaScript that we're going to go over is the `if` Statement. Those Statements use boolean conditions. By testing the boolean state of an expression, an `if` statement can decide whether or not to execute a particular block of code. This is one of the most basic examples of why you want to use block statements in your JavaScript.

```
var color = "red";

console.log("You chose " + color);
// "You chose red"

if (color == "blue") {
  console.log("Blue is my favorite color");
}
// no output because (color == "blue") is false
```

So here we check if the color equals "blue" and perform action only if this is true.

"If" and "Else" Keywords

`else` statements operate in the context of the `if`, and they let you tell the program what to do if the previous condition is false.

```
var color = "red";

console.log("You chose " + color);
// "You chose red"

if (color == "blue") {
  console.log("Blue is my favorite color");
} else {
  console.log("You have bad taste");
}
// "You have bad taste"
// because (color == "blue") is false
```

So it will perform the first action only if the statement is true, otherwise `else` action will be carried out.

Multiple "Else" Conditions

JavaScript actually supports multiple `else` conditions to a single `if` statement. This way you can branch to different blocks.

```
var color = "red";

console.log("You chose " + color);
// "You chose red"

if (color == "blue") {
  console.log("Blue is my favorite color");
} else if (color == "red") {
```

```
    console.log("That's not as bad as green");  
  } else {  
    console.log("You have bad taste");  
  }  
  // "That's not as bad as green"  
  // because (color == "blue") is false  
  // but (color == "red") is true
```

Using just the power of `if` and `else`, you can create complex programs that branch to multiple conditions, and solve different problems based on the state of the different variables in your program. But there are some other more advanced control flow statements in JavaScript to know about as well.