

LOGICAL OPERATORS

PROF. DAVID ROSSITER

AFTER THIS PRESENTATION

- You'll appreciate Boolean values
- You'll know more about logical operators and how to use them

WE WILL LOOK AT

Boolean values	true
----------------	------

	false
--	-------

Logical operators	&&
-------------------	----

--	--

	!
--	---

BOOLEAN

- A *Boolean* value is either true or false
- A variable which has a Boolean value is called a Boolean variable

LOGICAL OPERATORS

- Logical operators work with Boolean values
- JavaScript has these logical operators:
 - Logical And - the && operator
 - Logical Or - the || operator
 - Logical Not - the ! operator

AND - &&

- && - the result is true if both inputs are true, otherwise the result is false

AND - &&

a	b	a && b
false	false	false
false	true	false
true	false	false
true	true	true

```
<html><body><script>
  var you_are_rich = false;
  var you_have_partner = true;
  var you_have_flat = true;
  var life_is_fantastic = you_are_rich
    && you_have_partner && you_have_flat;
  alert("life is fantastic is " +
        life_is_fantastic);
  you_are_rich = true;
  life_is_fantastic = you_are_rich
    && you_have_partner && you_have_flat;
  alert("life is fantastic is now " +
        life_is_fantastic);
</script></body></html>
```


SHORT-CIRCUIT IN AND

- JavaScript is clever
- When it evaluates an And it checks the first input
- If the value is `false` it knows the result must be `false`
- So it doesn't bother checking the next input

```
<!doctype html>
<html>
  <body><script>
    function first_function() {
      alert("first_function() is running!");
      return true;
    }
    function second_function() {
      alert("second_function() is running!");
      return false;
    }
    var test_function =
      first_function() && second_function();
  </script></body>
</html>
```

AFTER SWAPPING THE FUNCTIONS

```
<html><body><script>
  function first_function() {
    alert("first_function() is running!");
    return true;
  }
  function second_function() {
    alert("second_function() is running!");
    return false;
  }
  var test_function_swapped =
    second_function() && first_function();
</script></body></html>
```

OR - ||

- || - the result is false if both inputs are false, otherwise the result is true

OR - ||

a	b	a b
false	false	false
false	true	true
true	false	true
true	true	true

```
<!doctype html>
<html>
  <body><script>
    var you_are_rich = false;
    var you_have_partner = true;
    var you_have_flat = false;
    var life_is_good = you_are_rich
      || you_have_partner || you_have_flat;
    alert("life is good is " + life_is_good);
    you_have_partner = false;
    life_is_good = you_are_rich
      || you_have_partner || you_have_flat;
    alert("life is good is now " + life_is_good);
  </script></body>
</html>
```

SHORT-CIRCUIT IN OR

- If JavaScript is evaluating Or and the first input is true, it knows the result must be true
- So it doesn't bother checking the second input

```
<!doctype html>
<html>
  <body><script>
    function first_function() {
      alert("first_function() is running!");
      return true;
    }
    function second_function() {
      alert("second_function() is running!");
      return false;
    }
    var test_function =
      first_function() || second_function();
  </script></body>
</html>
```


NOT - !

- ! - the result is the opposite of the input

NOT - !

a	!a
false	true
true	false

```
<!doctype html>
<html>
  <head>
    <title>Not Operator Example</title>
  </head>
  <body>
    <script>
      var you_are_male = true;
      var you_are_female = !you_are_male;
      alert("you_are_male is " + you_are_male);
      alert("you_are_female is " + you_are_female);
    </script>
  </body>
</html>
```