

Logical Operators and JavaScript Practice

CS 115 Computing for the Socio-Techno Web

Instructor: Brian Brubach

Announcements

- Special office hours 2-3pm today for Assignment 3
- Quiz grades will be released later today
 - Will be curved up slightly
- PM2 questions?

Download code for today

- Using Cyberduck, download the folder `ex04` from [cs115/download](#)
- More input/output examples → `fillblank.html`, `fillblank.js`

JavaScript comparisons

- You can compare values by using equality/inequality operators
 - `===` → Returns true if the values and types are equal, false otherwise
 - `!==` → Returns true if the values or types are different, false otherwise
 - `==` and `!=` → Attempt type conversion before comparing values, should usually be avoided due to sometimes unpredicted behavior
- Relational Operators
 - `<` → Less than, returns true if left value is less than right value (e.g., `x < y`)
 - `>` → Greater than
 - `<=` → Less than or equal
 - `>=` → Greater than or equal
- Example files → `comparison_string.html`, `comparison_number.html`

JavaScript if statement

- If statement → Control statement that allows us to make decisions

- First Form → if

```
if (expression)
    statement;           // executed if expression is true
```

- Second Form → if else

```
if (expression)
    statement1;         // executed if expression is true
else
    statement2;         // executed if expression is false
```

- To execute more than one statement use a set of { } around statements
- Example files → if_statement1.html, if_statement2.html

JavaScript logical operators

- Used with comparison operators to create more complex expressions
- Logical “and” (&&) → `exp1 && expr2`
 - Expression is true if and only if both expressions are true otherwise is false
- Logical “or” (||) → `expr1 || expr2`
 - False if and only if both expressions are false otherwise is true
 - True if `exp1` or `exp2` or both are true
 - True if at least one expression is true
- Logical Not (!) – `!expr`
 - Inverts the boolean value of the expression
 - Example: `!(1 === 2)` is true
- Example files → `logical_op1.html`, `logical_op2.html`

Precedence/associativity

- Remember you can use parenthesis to impose a particular order for the evaluation of an expression
- Example $\rightarrow (x === y \ \&\& \ y === z) \ || \ (a === b \ \&\& \ b === c)$
 - True if x, y, and z are equal or a, b, and c are equal, or both
- Example files \rightarrow more_weather.html, more_weather.js