

Announcement

- Gradebook has been down
- So we haven't been able to transfer your quiz scores over to MyUW
- Supposed to be fixed this morning



Announcement

- Marc's Friday office hour
 - * Has miraculously transformed into
 - 2 hours
 - -In a computer classroom
 - » MGH 030 from 4-6pm on Fridays



FIT100

Announcement

- Project 2A
- * Available Friday
- * Due on Wednesday
 - 2 paragraph story
 - 2 images
 - Copyright information
 - Choose words in story to replace



Keepin' on with the Program:

Fundamental Programming Concepts Expressed in JavaScript (continued)

© 2004 Lawrence Snyder



Exercise Part 4,

 You'll understand more as we work through the next few slides.



Right side in the assignment statement

EXPRESSIONS



An Expression and its Syntax

- Algebra-like formula called an expression
 - * Describe the means of performing the actual computation
 - * Built out of values and operators
 - Standard *arithmetic operators* are symbols of basic arithmetic



Arithmetic Operators

- Multiplication requires an asterisk (*), the multiply operator
- Multiply and divide are performed before add and subtract
 - * Anything within parentheses is done first
 - * Any multiplication or division within parentheses is performed first
- No operator for exponents
- Modulus or mod (%) divides two integers and returns the remainder



Relational Operators

- Make comparisons between numeric values
- Outcome is a **Boolean** value, true or false
- < less than
- <= less than or equal to</p>
- == equal to

(Note difference between = and = =)

- != not equal to
- >= greater than or equal to
- > greater than



Logical Operators

- To test two or more relationships together
- * Teenagers are older than 12 and younger than 20
- · Logical AND
 - * Operator is &&
 - Outcome of a && b is true if both a and b are true; otherwise it is false
- Logical OR
 - * Operator is | |
 - * Outcome of a | | b is true if either a is true or b is true
- Logical NOT
 - * Operator is!
 - * Outcome is opposite of value of comparison



Operators (cont'd)

- Operator Overload
 - * Use of an operator with different data types
 - * Case of interest in JavaScript is +
- Addition
 - * When used with numbers, + adds
 - 4 + 5 produces 9
- Concatenation
 - * When + is used with strings, + concatenates or joins the strings together
 - "four" + "five" produces "fourfive"



A Conditional Statement

if (<Boolean expression>) <then-statement>;

- Boolean expression is a relational expression;
 - * Evaluates as either True or False
- then-statement is any JavaScript statement



If Statements and Their Flow of Control

- The Boolean statement, called a predicate, is evaluated, producing a true or false outcome
- If the outcome is true, the then-statement is performed
- If the outcome is false, the thenstatement is skipped
- Then-statement can be written on the same line as the Boolean or on the next line



Compound Statements

- Sometimes we need to perform more than one statement on a true outcome of the predicate test
- You can have a sequence of statements in the then clause
- Group these statements using curly braces { }
 - * They are collected as a compound statement



if/else Statements

To execute statements if a condition is false

- The Boolean expression is evaluated first
 - If the outcome is true, the then-statements are executed and the else-statements are skipped
 - * If the outcome is false, the then-statements are skipped and the else-statements are executed



Nested if/else Statements

- The then-statement and the elsestatement can contain an if/else
- The else is associated with the immediately preceding if
- Correct use of curly braces ensures that the else matches with its if



Nested if/else Statements



Exercise Part 4

Replacing Variables with values	Result	Number	String	Boolean
"Donald" + "Duck"	"DonaldDuck"		Х	
"Donald " + "Duck"	"Donald Duck"			
"Donald" + 10	"Donald10"		Χ	
75 /25 < 100 = 3 < 100	true			х
(25 > 100) OR (75 < 100)	If either is true, it's true			х
h = 75		Χ		
75==75?	True			Х
	with values "Donald" + "Duck" "Donald" + "Duck" "Donald" + 10 75 /25 < 100 = 3 < 100 (25 > 100) OR (75 < 100) h = 75	with values Result "Donald" + "Duck" "DonaldDuck" "Donald " + "Duck" "Donald Duck" "Donald" + 10 "Donald10" 75 /25 < 100	"Donald" + "Duck" "DonaldDuck" "Donald " + "Duck" "Donald Duck" "Donald " + 10 "Donald10" 75 /25 < 100 true = 3 < 100 (25 > 100) OR (75 < If either is true, it's true) h = 75 X	"Donald" + "Duck" "DonaldDuck" X "Donald " + "Duck" "Donald Duck" "Donald " + 10 "Donald10" X 75 /25 < 100 true = 3 < 100 (25 > 100) OR (75 < If either is true, it's 100) true X



Working Together

HTML, CSS, AND JAVASCRIPT



Purposes of Each

- Three separate types of coding
 - * HTML—for content
 - * CSS—for appearance
 - * JavaScript—for action



Examples

- HTML—static page
- CSS—add styling to the page
- JavaScript—adds action!





JavaScripts and HTML

Types of JavaScripts are based on location in the HTML page:

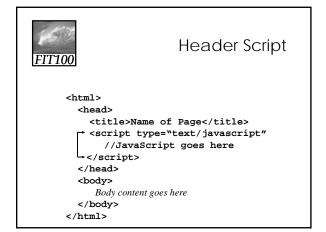
- * Body scripts—body section
- * Header scripts—head section
- * External scripts—links to a .js page
 - Similar to .css pages



Body Script



Body Script





External Script

- Linked in the <head>
- src gives pathname



External JavaScripts

- Make changes to scripts in one place
- Reusable
 - * Can be linked to any page, every page in a site, or many sites





Summary

- Programming is the exact specification of an algorithm
- JavaScript is typical ... with many rules
 - * Learning strategy
 - Do the reading first
 - Practicing is better than memorizing for learning the rules
 - Use the program-save-reload-test plan
 - Precision is your best friend