JavaScript Flash Cards

Data Types

Basic JavaScript Creation		
Basic JavaScript Terms		
Basic JavaScript Terms		
Concatenation		
Arithmetic Operators		
Arrays		
Naming Variable Rules		
Subject 4		
IfÉlse Statements		
Logical Operators		
Comparison Operators		
Subject 4		
Truthy and Falsy Values		
Equality & Existence		
Short Circuit Values		

Weak or Strong Typed Content 5

222233334

Switch Statements
Subject 4
Functions - declaring a function
Immediately Invoked Function
Expressions (IIFE) p 97
Loop Types
Loops
Subject 4
Subject 1
Event Flow (chapter 6 Events
continued)` ·
Events (chapter 6)
, ,

```
Basic JavaScript Terms
                                           Basic JavaScript Creation
Object
                                           var name = variable value:
                                           code (parameters) { code
  Property
    Name: Value (Value Pairs)
                                           } {code
Method()
                                           } {code
Event.
JavaScript scripts reside alongside html
                                           Document.methodname(parameter, parameter)
document, recommended in a separate
folder for organization.
                                           Data Types
Basic JavaScript Terms
JavaScript appears in html as
                                           Numeric - numbers, including floating
                                             values
<script src="dir/filename.js"></script>
                                           String - 'letters and other characters.'
                                           Boolean - will return true or false
```

Naming Variable Rules 1. Name must begin with letter, \$, or .

2. Name cannot use a - or . 3. Cannot be a keyword or reserved word.

4. All variables are case sensitive.

5. Be correctly descriptive.

6. Use camelCase for new words, and -(dashes) are also acceptable.

Arithmetic Operators + Addition - Subtraction

/ Divison * Multiplication ++ increment - +1 from current number

-- Decrement - -1 from current number % Modulus - divides by value and returns the remainder.

Arrays

OR

Create by - (below is an array literal) var colors;

colors = ['white', 'black', 'custom'; Call by -

colors.item(0) (returns 'white') colors[2] (returns 'custom')

Arrays began at 0.

Concatenation

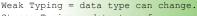
+ combines values to a string if a string value is present in the concatenation.

```
Comparison Operators
                                          If...Else Statements
      equal to
      is not equal to
                                          keyword (condition) {
      strict equal to
                                               code to execute if true();
!==
      strict not equal to
      greater than
                                          else {
      less than
                                               code to execute if false();
>= greater than or equal to
<=
      less than or equal to
Logical Operators
                                          Subject 4
                                          Content
&& logical and (tests more than one
               condition)
|| logical or (tests at least one
               condition)
! Logical not (takes a single Bool and
   inverts it)
```

```
Switch Statements
switch (var being evaluated) {
 case (var result):
    var if true:
    break:
 case (next until choose to stop):
    var if true;
    hreak
 default:
    var if none resolve true;
    Break:
element.element = var resolved as true
Weak or Strong Typed Content
Strong Typing = data type for a variable
is specified prior.
```

See p 167 Equality & Existence See p 168 Short Circuit Values See p 169

Truthy and Falsy Values



Subject 4

Content

Loops	Functions - declaring a function
For (var i = value; I < 10; i++) {	Function function name(argument1, argument2) { code; return var; } Immediately Invoked Function Expressions (IIFE) p 97 A way to "brute force" a piece of code. "Wrappers" to make things happen.
Loop Types	Subject 4
for while do	Content

Events (chapter 6)	Subject 1
Events are raised or fired (likely by a user), that then triggers a function or script. Pp 246-247 outline many of the events. How you script an event uses the DOM and is like other Javascript DOM manipulation Element.onevent - functionName; ^^Event handler format; element.addEventListener('event', functionName [,Boolean]);	Content
Event Flow (chapter 6 Events continued)	Subject 4
event bubbling: starting at most specific to least. Event capturing: event starts at least specific and flows inward.	Content
User Interface Events (pp 272-282) These pages address the details of having JavaScript interact with what is going onscreen, what the user is doing. Events like focus and blur, mouse clicks, keyboard, events all controllable by events and event listeners via DOM manipulation.	