CSE 3302 Notes 9: JavaScript

(Last updated 11/9/12 12:27 PM)

9.1. HISTORY

Wikipedia (Netscape . . .)

Self/prototype-based/delegation (paper on webpage)

Scheme/functional language elements

9.2. TYPES

Number - 64-bit IEEE 754

Integers lose precision outside of ±2^{53}

Array subscripting and bitwise operations on 32-bit integers

String

No `char` type

Delimited by pair of ' or pair of ". Allows `'Hit the "return" key'`

`String(123)` `string.length` `string.search()`

Boolean

`false`	`undefined`	`null`	`0`	`-0`	`NaN`	`""`
`true`	everything else					

Undefined

Used when a property does not exist for an object.

To access `a.b.c.d` or get `undefined` (to avoid `TypeError`):

```
dCheck = a && a.b && a.b.c && a.b.c.d;
```

Based on short-circuit evaluation, JavaScript uses the last truthy/falsy value for expression. (`!` sanitizes)

Misspelled property name vs. property with `undefined` as value . . .

Object

Set of property names (strings if needed)

Literal - expression to initialize each property (`undefined` is allowed)

Function

First-class: can be constructed as an argument (anonymous)

Four kinds of invocation and `this`:

Function is not a property of an object: the global object

Method invocation for object (e.g. object.function()): the object

Function call prefixed with `new`: a new object (if an object is not returned, then new object will be returned)

`apply` (a method) allows applying a method using 1) an arbitrary object as `this` and 2) an array of arguments.

What is an array?

An array-like object that handles subscripts as integer property names.

`a[i]=` . . . will make `i` a property of `a`

`delete a[i]` removes `i` as a property (similar for removing property of an object)

`length` property

Array contents are unrestricted

Inherits from `Array` prototype

Additional properties may be attached to an array

`objArray.html`

9.3. Functional Approach to Objects (Closures)

Classes at each level in a class hierarchy are simulated by constructor functions with arguments:

Data needed to initialize

Functions for overriding superclasses

Returns an object with the appropriate properties, but not protected data.

Or . . . use

```
shape.func.html
```

How about extending with class members, etc. for counting instances of scalene/isosceles/equilateral triangles and other shapes?

9.4. Simulated Class Approach to Objects

Everything public?

```
shape.class.html
```

9.5. Prototypal Approach to Objects