JavaScript: Objects, BOM, and DOM

CS 4640
Programming Languages for Web Applications

Objects group variables and functions to create a model representing something you would recognize from the real world

Object type: Hotel

Event	Happens when
Reserve	reservation is made
Cancel	reservation is cancelle

Events are things or interactions that can happen to the objects

Method	What it does
makeReservation()	increases value of *bookings* property
cancelReservation()	decreases value of *bookings* property
checkAvailability()	subtracts value of *bookings* property from value of *rooms* property and returns number of rooms available

Methods represent tasks that are associated with the objects (or things we can do with the objects)

Properties

Name:	Awesome
Rating:	5
Rooms:	70
Bookings:	56
Pool:	true
Gym:	true

Properties tell us the characteristics of the objects

Object type: Car

Event	Happens when	
Accelerate	driver speeds up	changeSpeed()

Method	What it does
changeSpeed()	increases or decreases value of *currentSpeed* property

Properties

Make:	UVA1
currentSpeed:	30
Color:	yellow
Fuel:	gasoline

JavaScript Objects

revisit

- JavaScript is an object-based language
  - It supports for object-oriented programming but not at the same level as other languages (ES6: introduced `class` – still lacks private property)
- Objects are represented as property-value pair
  - The property values can be data or functions (methods)
- A property is something that can be modified :
  - Data properties : primitive values or references to objects
  - Method properties : can be executed
- Objects can be created and their properties can be changed dynamically
  - JS is not really typed .. If it doesn't care between a number and a string, why care between two kinds of objects?

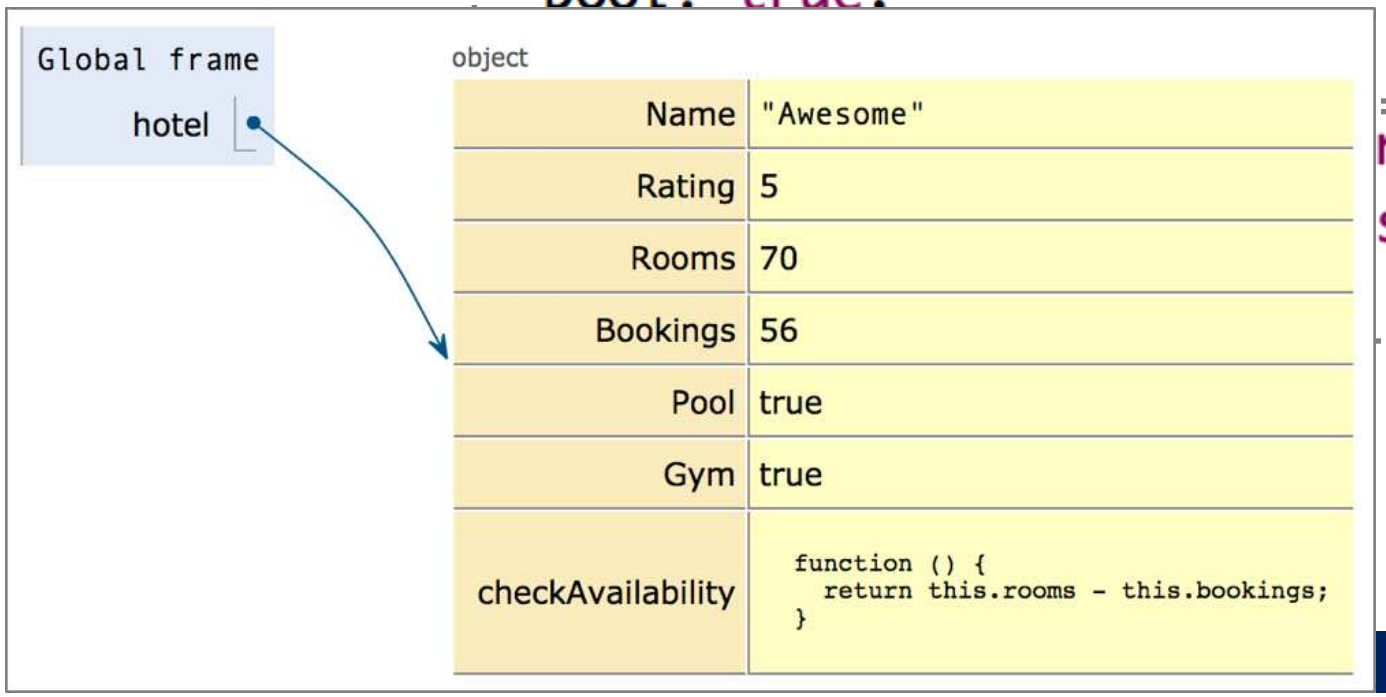
Creating Objects

revisit

Create an object and assign variables and functions directly by using { } syntax

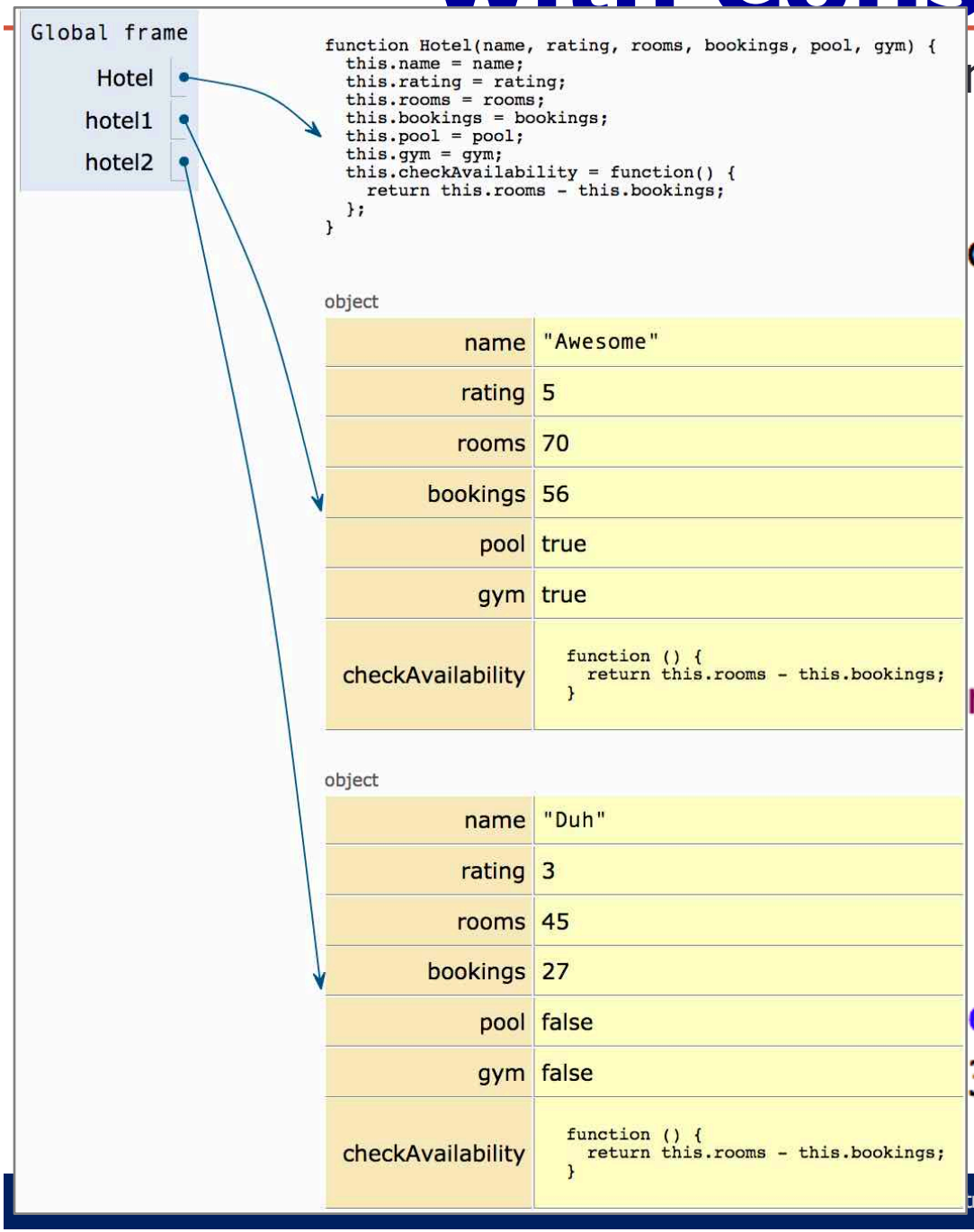
```
var hotel = {
  name: "Awesome",
  rating: 5,
  rooms: 70,
  bookings: 56,
  pool: true,
```

```
n() {
s.bookings;
```



Creating Objects with Constructors

revisit



ng the constructor function and

```
ooms, bookings, pool, gym) {
```

```
nction() {
.bookings;
```

```
e', 5, 70, 56, true, true);
3, 45, 27, false, false);
```

revisit

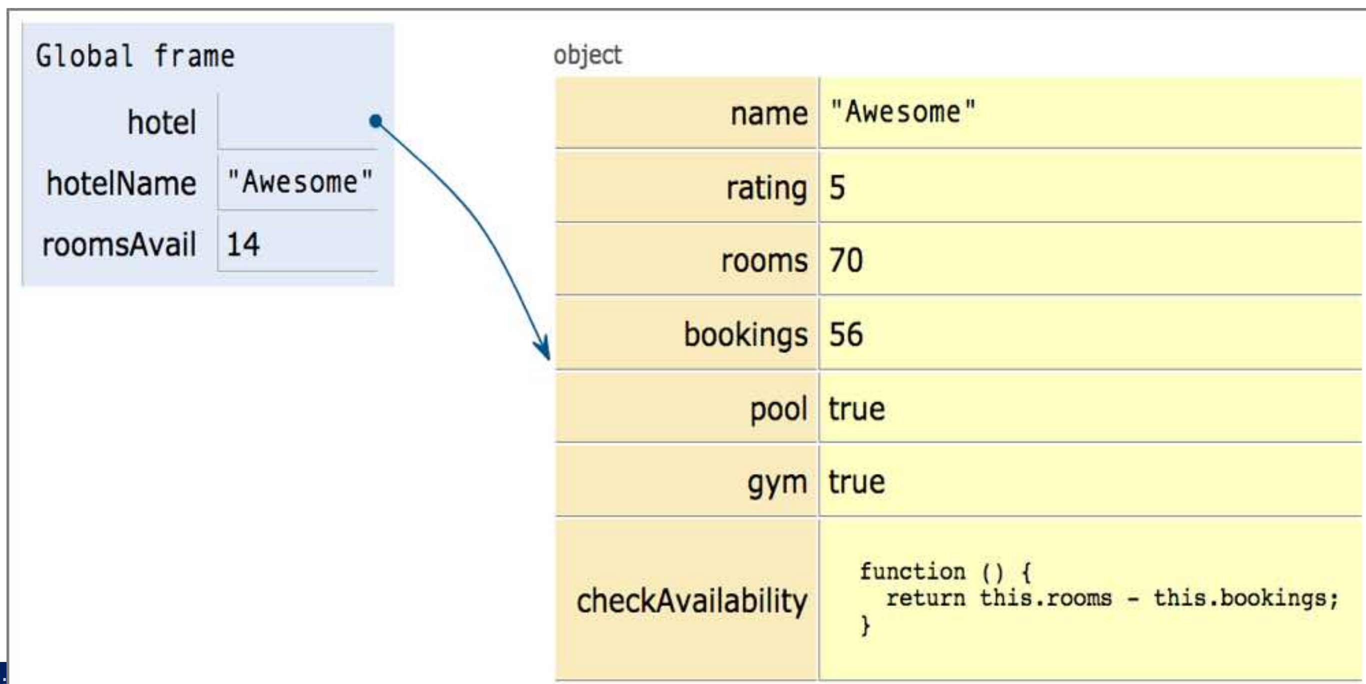
Accessing Objects

- Access properties or methods of an object using dot notation

```
var hotelName = hotel.name;
var roomsAvail = hotel.checkAvailability();
```

- Access properties or methods using square brackets

```
var hotelName = hotel['name'];
var roomsAvail = hotel['checkAvailability']();
```



Updating Properties

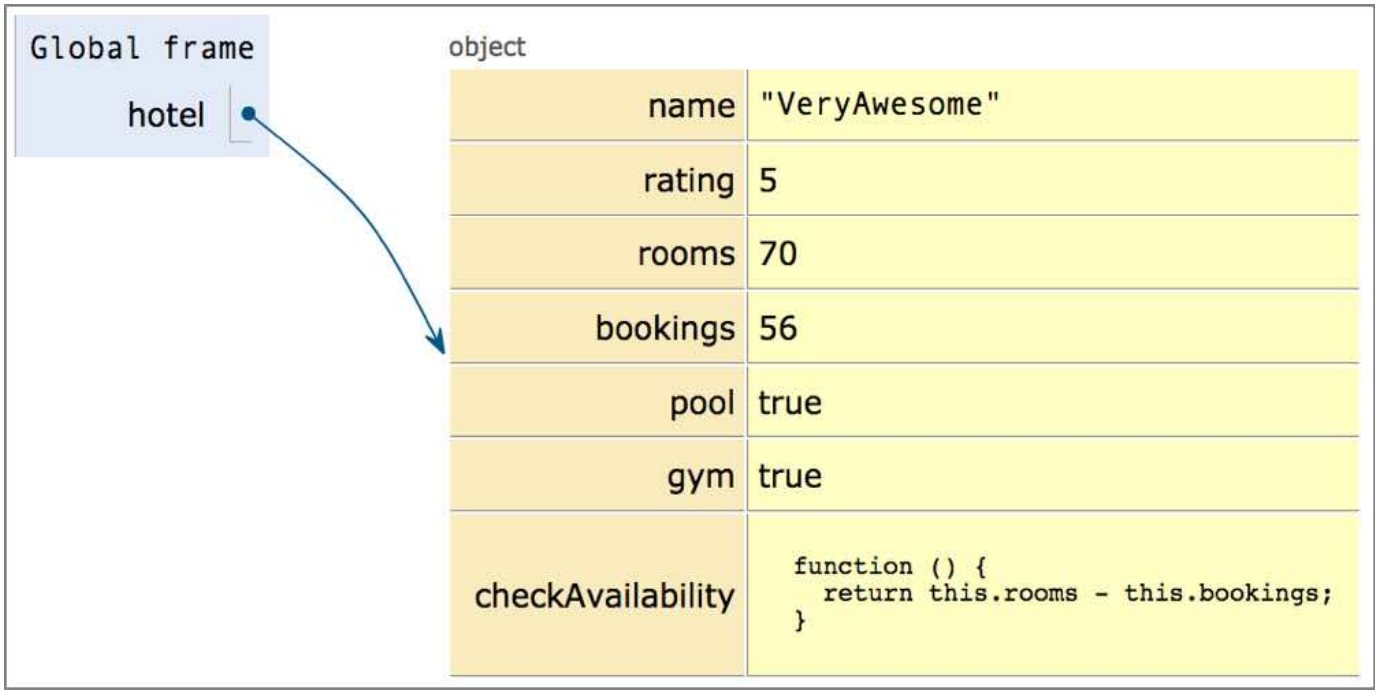
revisit

- Update properties using dot notation

```
hotel.name = 'VeryAwesome';
```

- Update properties using square brackets

```
hotel['name'] = 'VeryAwesome';
```

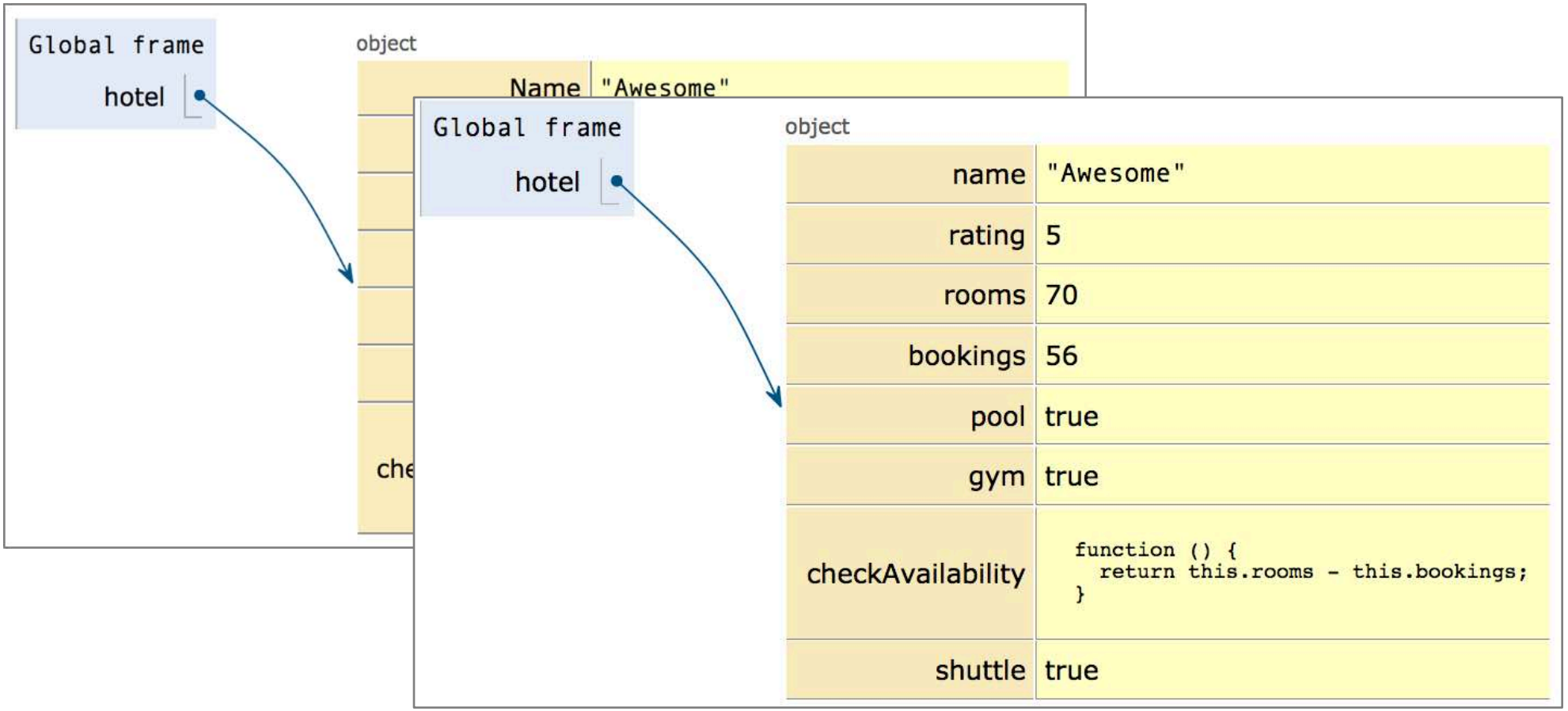


Adding Properties

revisit

- Add a property using a dot notation

```
hotel.shuttle = true;
```

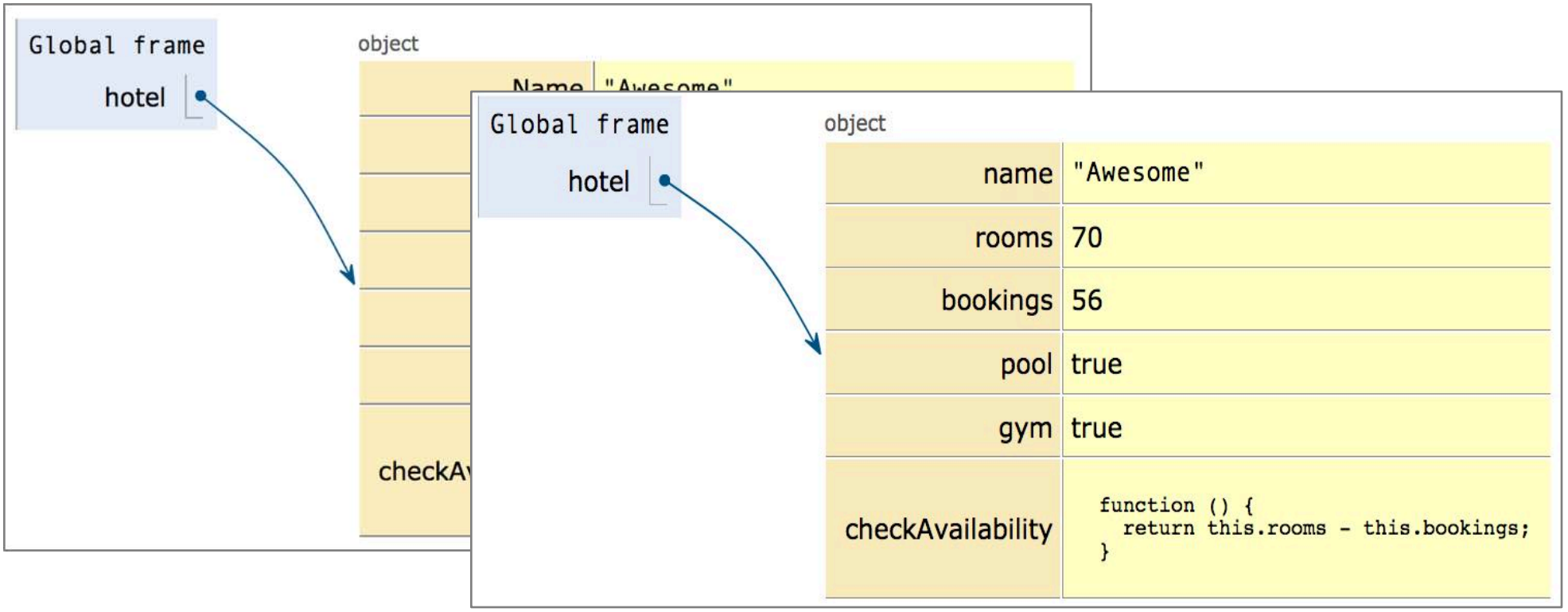


Deleting Properties

revisit

- Delete a property using the delete keyword

```
delete hotel.rating;
```

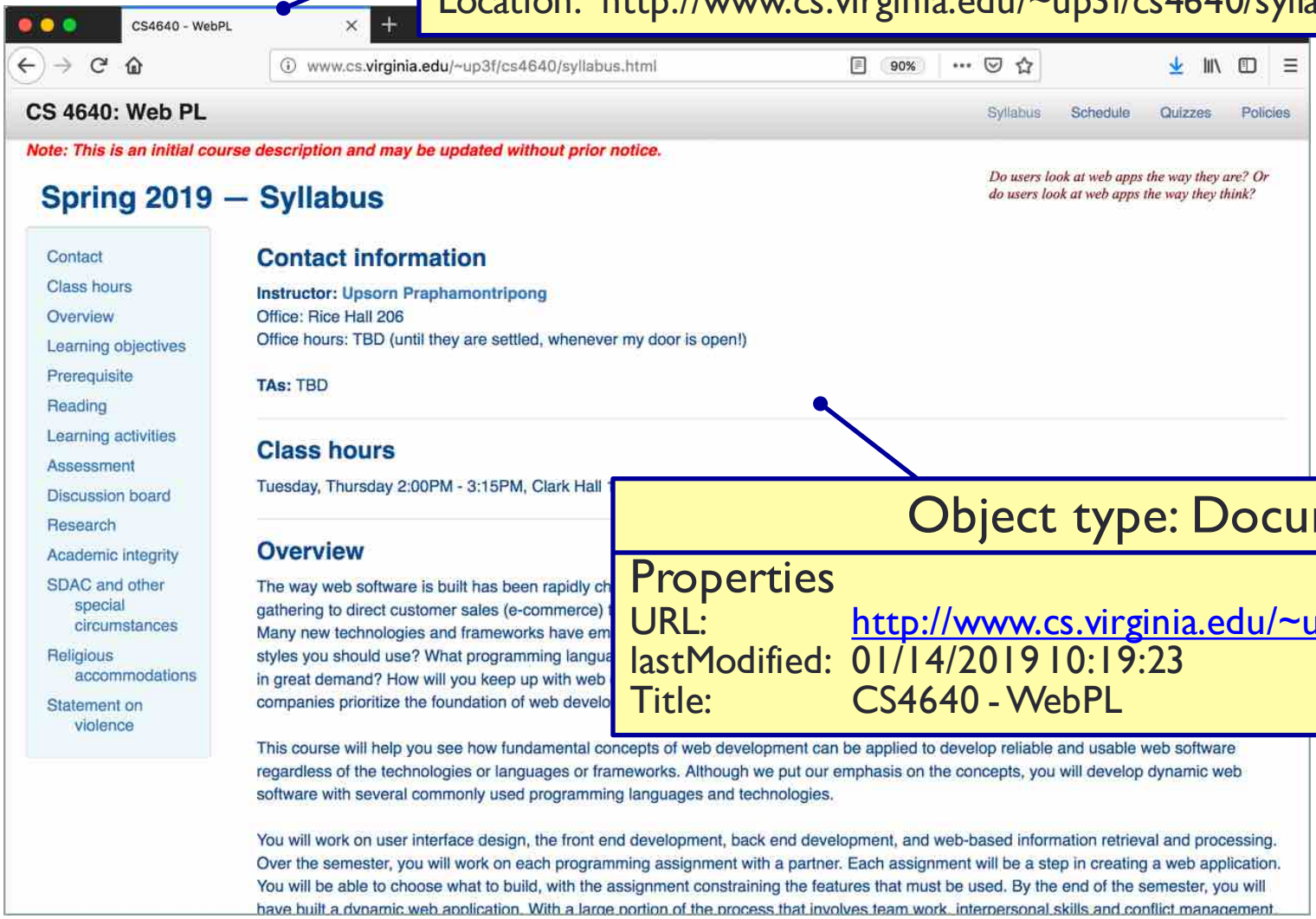


Web Browsers and Objects

revisit

Object type: Window

Properties
Location: http://www.cs.virginia.edu/~up3f/cs4640/syllabus.html



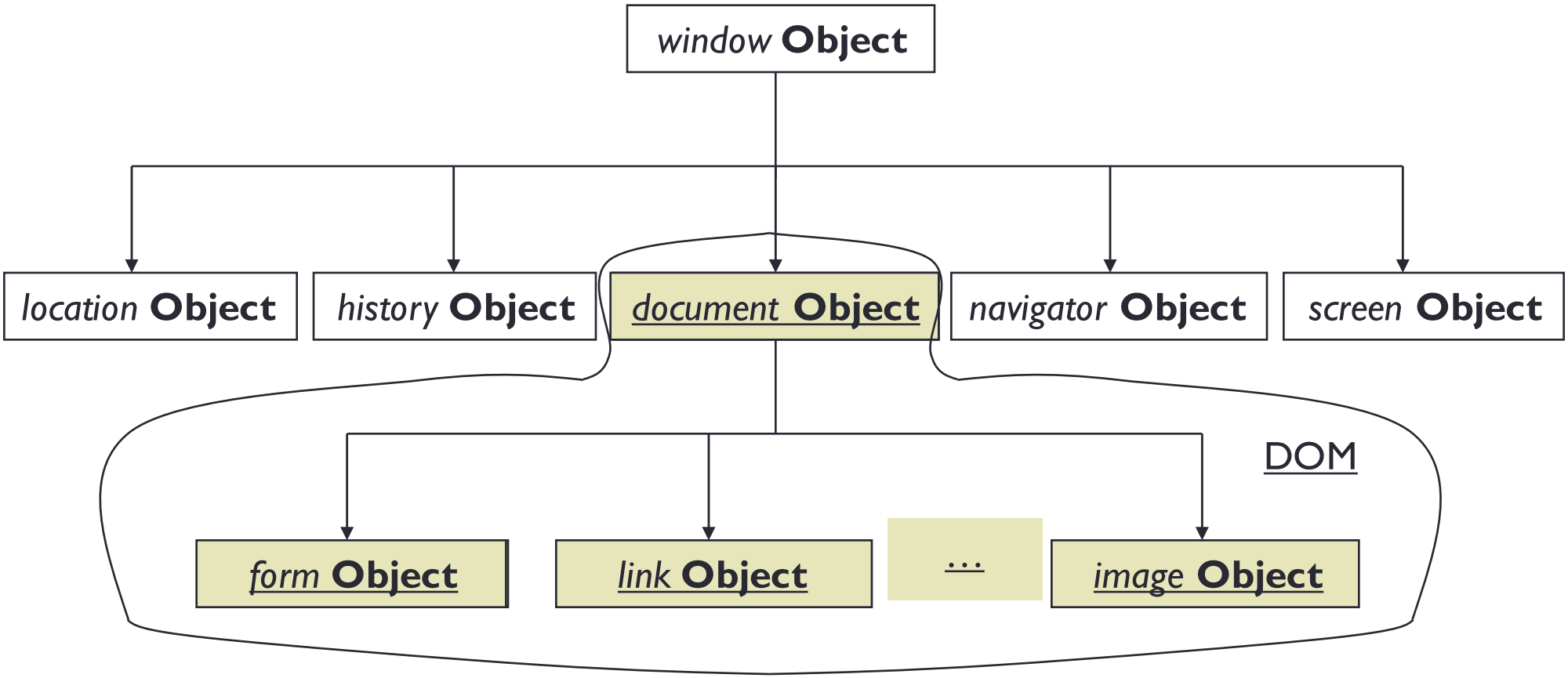
Object type: Document

Properties
URL: http://www.cs.virginia.edu/~up3f/cs4640/syllabus.html
lastModified: 01/14/2019 10:19:23
Title: CS4640 - WebPL

BOM: Browser Object Model

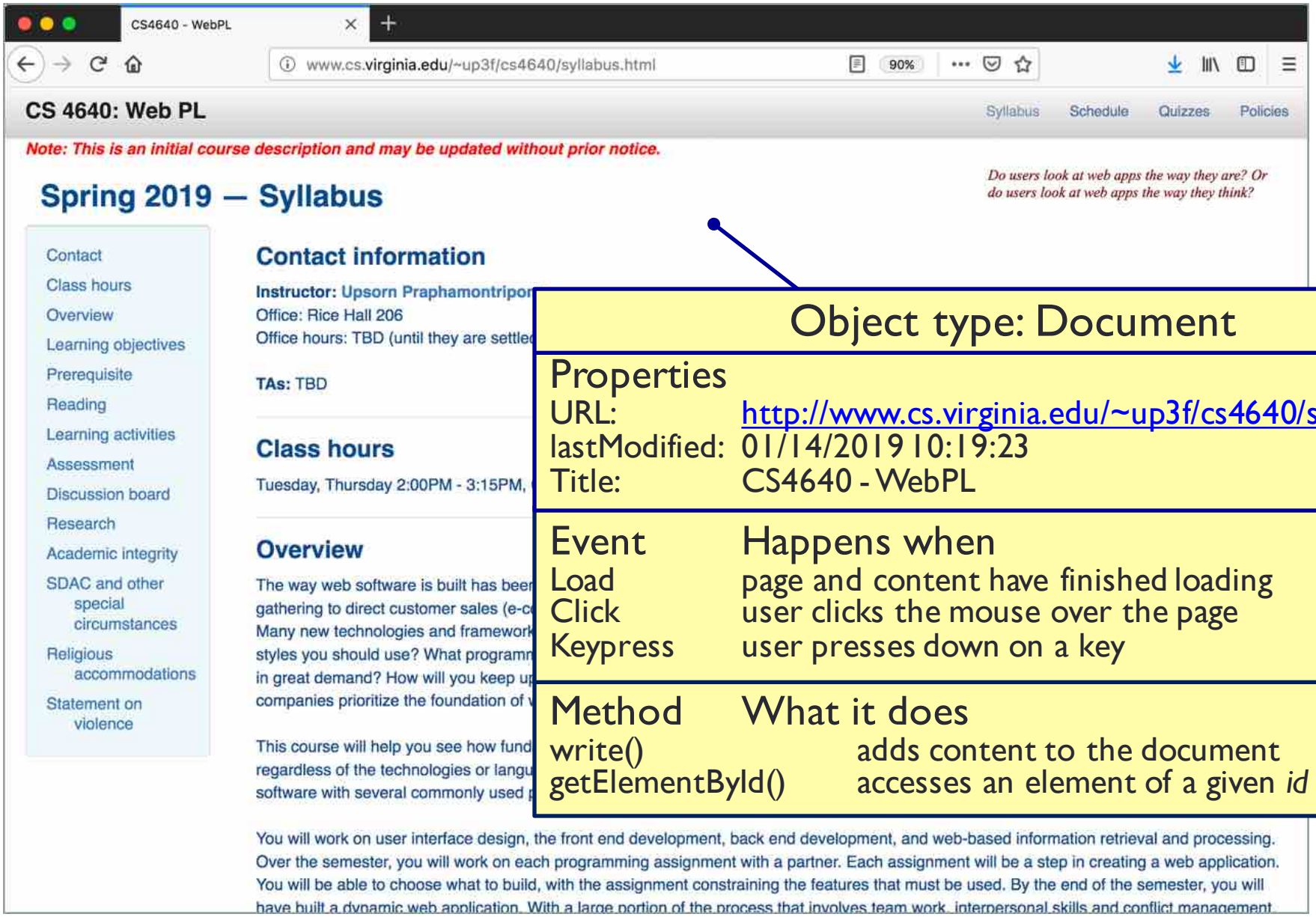
revisit

- BOM – collection of objects that the browser makes available to us for use with JavaScript



DOM: Document Object Model

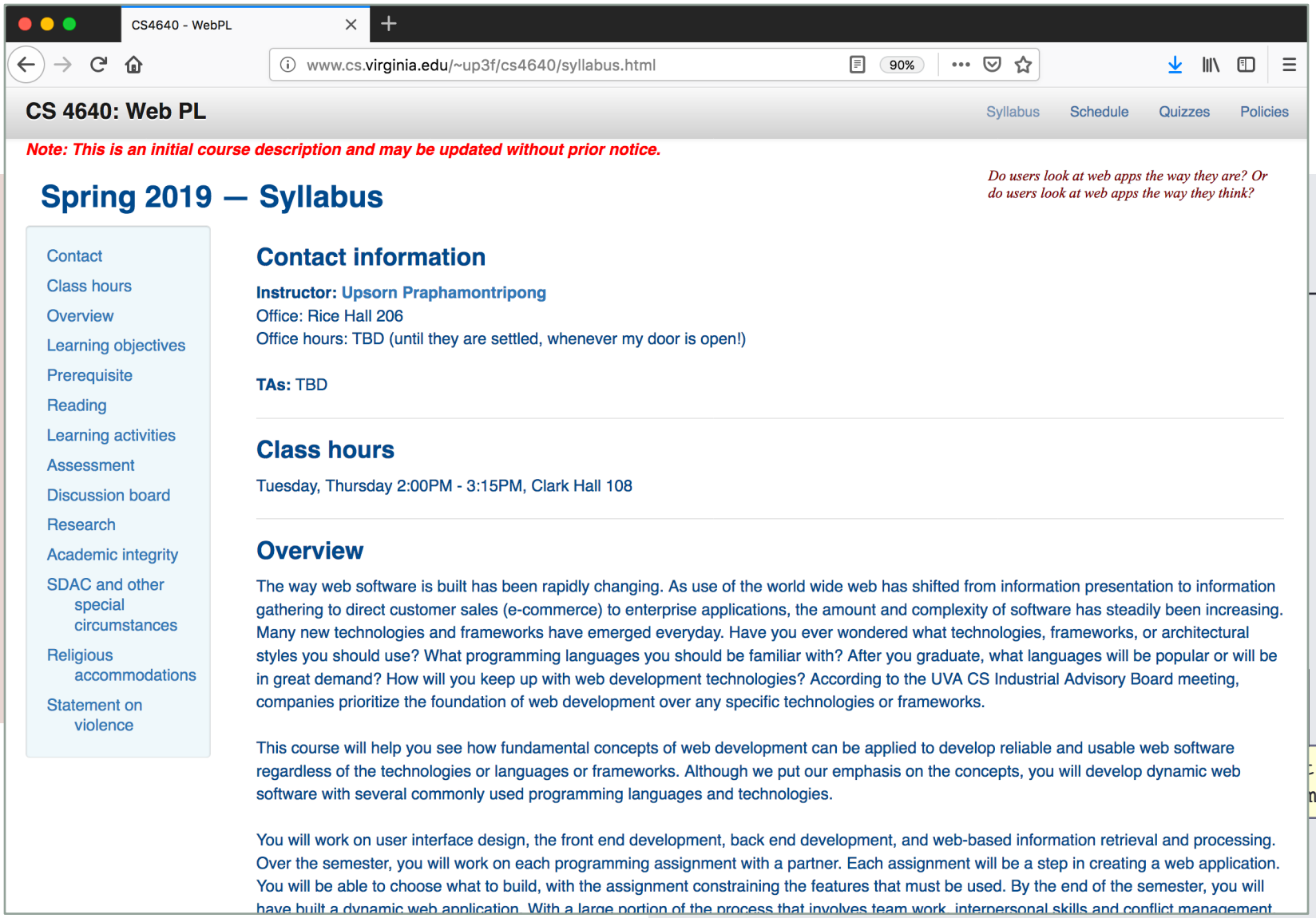
revisit



Object type: Document	
Properties	
URL:	http://www.cs.virginia.edu/~up3f/cs4640/syllabus.html
lastModified:	01/14/2019 10:19:23
Title:	CS4640 - WebPL
Event	**Happens when**
Load	page and content have finished loading
Click	user clicks the mouse over the page
Keypress	user presses down on a key
Method	**What it does**
write()	adds content to the document
getElementById()	accesses an element of a given *id* attribute

How A Browser See A Web Page

revisit



The browser receives an HTML page

It creates a model of the page and stores it in memory

It shows the page on screen using a rendering engine

Using BOM Objects (Some Properties)

Property	Description
window.screenX	X-coordinate of pointer, relative to top left corner of screen (in pixels)
window.screenY	Y-coordinate of pointer, relative to top left corner of screen (in pixels)
window.location	Current URL of window object
window.document	Reference to document object
window.history	Reference to history object for browser window or tab, which contains details of the pages that have been viewed in that window or tab
window.history.length	Number of items in history object
window.screen	Reference to screen object
window.screen.width	Accesses width property of screen object
window.screen.height	Accesses height property of screen object

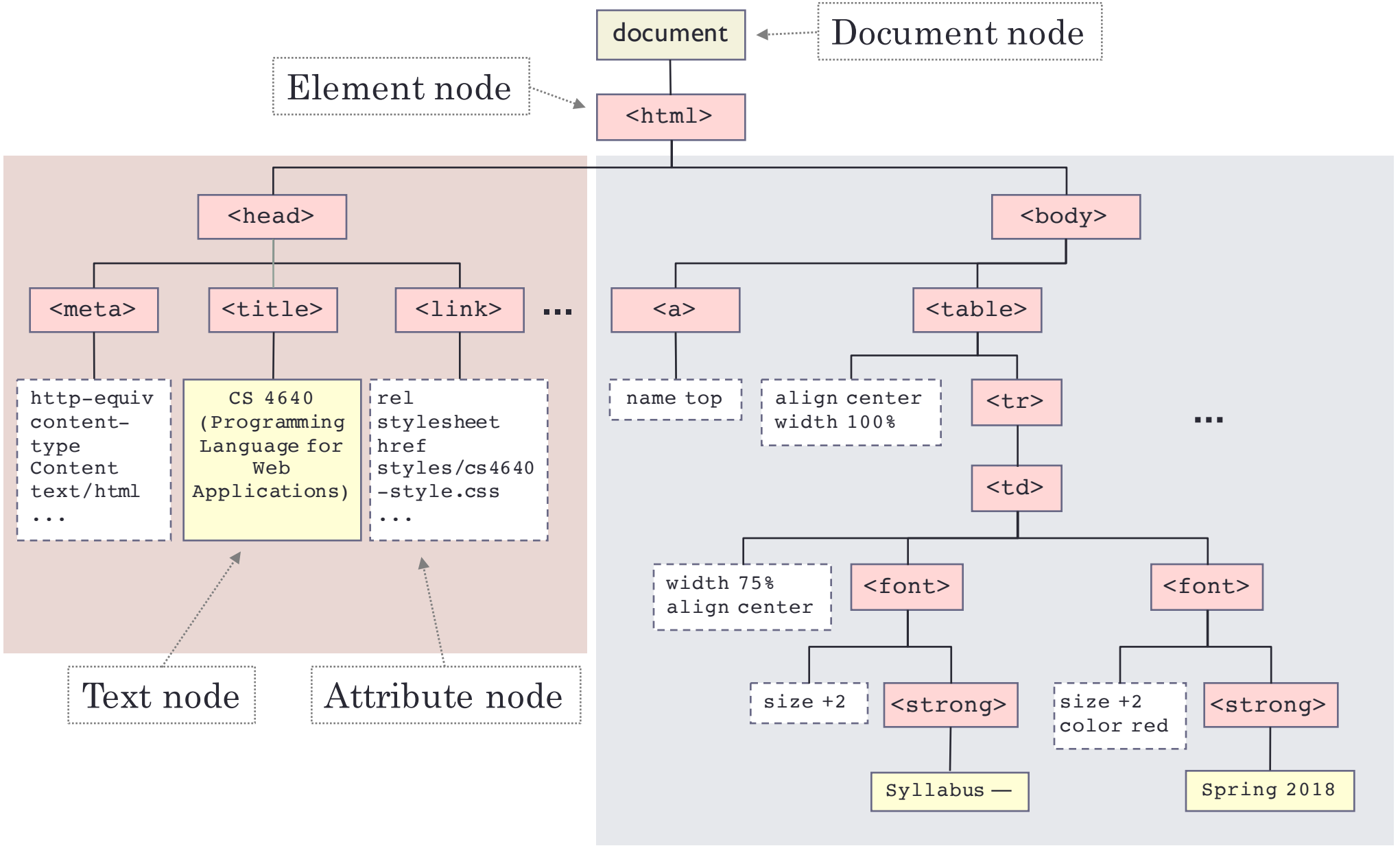
Using BOM Objects (Some Methods)

Method	Description
window.alert( )	Create modal dialog box with message (user must click OK button to close it)
window.open(url)	Open new browser window with the specified URL
window.print( )	Tell browser that user wants to print contents of current page (act like user has clicked a print option)
window.history.back( )	Move backward through history
window.history.forward( )	Move forward through history
window.history.go(step)	Move to specific page from session history (step specifies the number of pages, forward or backward)
history.pushState(state, title, url)	Create a new entry (or add a URL) at the top of the browser history
history.replaceState(state, title, url)	Modify the current entry (current URL at the top) of the browser history

Using DOM Objects

- Not part of HTML or JS
- Separate rules implemented by all major browser markers
- Two primary purposes:
  - Making a model of the HTML page
    - Specifies how browsers should create a model of an HTML page
  - Accessing and changing the HTML page
    - Specifies how JS can access and update the contents of a web page

DOM: Four Types of Nodes



Using DOM Objects
(Some Properties and Methods)

Property	Description
document.title	Title of current document
document.lastModified	Date on which document was last modified
document.URL	String containing URL of current document
document.domain	Domain of current document

Method	Description
document.write( )	Write text to document
document.getElementById(id)	Return element whose id attribute matches the specified id
document.querySelectorAll(selector)	Return list of elements that match the specified CSS selector
document.createElement(element)	Create new element
document.createTextNode(text)	Create new text node