How to Determine JavaScript Variable Data Types

Paul D. Sheriff

BUSINESS/TECHNOLOGY CONSULTANT

paul.d.sheriff@gmail.com

Module Goals

Different data types

- Primitives
- Objects

How to determine data types

- typeof operator
- constructor property
- instanceof operator

Primitive Data Types

Data type	Description
boolean	true or false
null	no value
undefined	a variable declared, but has no value
number	integers, decimals, float, etc.
string	a series (array) of characters

Object Data Types

Data type	Description
new Array	A collection of values
new Error	Contains a name and an error message
new Function	A block of code
new Object	A wrapper around any type
new RegExp	A regular expression

Object Data Types

Data type	Description
new Boolean	An object that contains true or false
new Number	An object that contains a numeric value
new String	An object that contains a character(s)

NOTE: Use the primitives 'boolean', 'number' or 'string' instead of these when possible

typeof Operator

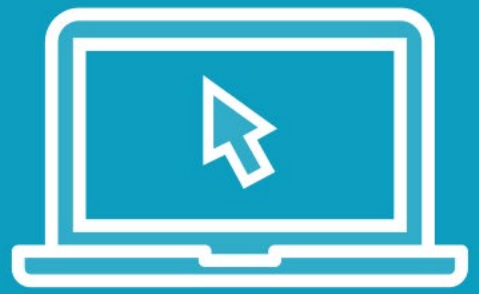
Returns the data type of the passed in expression

A string value is returned such as:
'string', 'number', 'object', etc.

typeof Operator

```
console.log(typeof "Hello");    // prints 'string'
console.log(typeof 4);          // prints 'number'
console.log(typeof (4 * 2));    // prints 'number'
```

Demo



typeof operator

Object Data Type / Constructor

All object data types inherit from Object (not primitives)

Object has constructor property

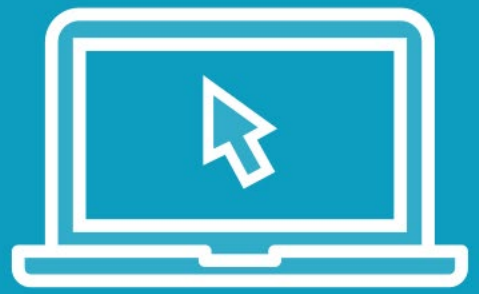
Returns a reference to the object itself

```
_products = function Array() { [native code] }
product = function Object() { [native code] }
product.productID = function Number() { [native code] }
product.productNumber = function String() { [native code] }
strValue = function String() { [native code] }
introDate = function Date() { [native code] }
isActive = function Boolean() { [native code] }
```

Constructor property displayed in console.log()

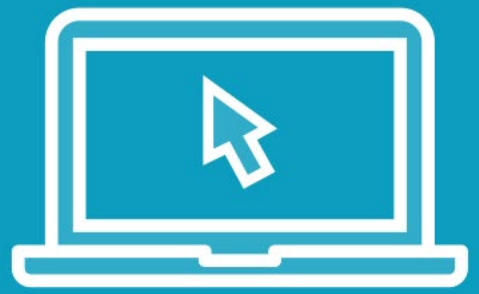
Object literals and primitives are cast to objects for display

Demo



constructor property

Demo



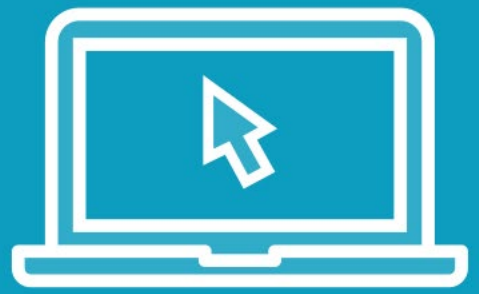
Some helper functions

instanceof Operator

Tests if inherits from Object (not a primitive)

Tests for a specific type of object

Demo



instanceof operator

Summary

Important to understand the difference between primitives and Objects

Use primitives where possible

Detect data types using typeof and instanceof

- typeof for checking type
- Instanceof for checking what type of object

Can use constructor property

- Both on objects and primitives

Coming up in the next module...

Use of 'this' in different scopes

call() and apply() method