

Data Types

number `var myNum = 123.456;` Series of numbers; decimals ok; double-precision floating-point format

string `var myString = "abc-def";` Series of characters (letters, numbers, or symbols); double-quoted UTF-8 with backslash escaping

boolean `var myBool = true;` true or false

array `var myArray = ["a", "b", "c", "d"];` sequence of comma-separated values (any data type); enclosed in square brackets

object `var myObject = {"id": 7};` unordered collection of comma-separated key/value pairs; enclosed in curly braces; properties (keys) are distinct strings

null `var myNull = null;` variable with null (empty) value

undefined `var myUndefined;` variable with no value assigned

Objects

```
var myObject = {
  "fir st": " Joh n",
  " las t": " Doe ",
  " age ": 39,
  " sex ": " mal e",
  " sal ary ": 70000,
  " reg ist ere d": true
};
```

Access object properties

<code>myObject.sex</code>	returns "male"
<code>myObject["age"]</code>	returns 39
<code>myObject[0]</code>	returns "John"
<code>myObject.something</code>	returns undefined
<code>myObject[6]</code>	returns undefined

Array of objects

```
var myArray = [
  {
    "fir st": " Joh n",
    " las t": " Doe ",
    " age ": 39,
    " sex ": " mal e",
    " sal ary ": 70000,
    " reg ist ere d": true
  },
  {
    "fir st": " Jan e",
    " las t": " Smi th",
    " age ": 42,
    " sex ": " fem ale ",
    " sal ary ": 80000,
    " reg ist ere d": true
  },
  {
```

Array of objects (cont)

```
> "first": "Amy",
  "last": "Burnquist",
  "age": 29,
  "sex": "female",
  "salary": 60000,
  "registered": false
}
];
```

Access array elements

<code>myArray[0]</code>	returns { "first": "John", "last": "Doe" ... }
<code>myArray[1]</code>	returns { "first": "Jane", "last": "Smith" ... }
<code>myArray[1].first</code>	returns "Jane"
<code>myArray[1][2]</code>	returns 42
<code>myArray[2].registered</code>	returns false
<code>myArray[3]</code>	returns undefined
<code>myArray[3].sex</code>	error: "cannot read property..."

Arrays

```
var myArray = [
  " Joh n",
  " Doe ",
  39,
  " M",
  70000,
  true
];
```



By **Mackan90096**
(Mackan90096)

cheatography.com/mackan90096/

Published 9th March, 2015.
Last updated 12th May, 2016.
Page 1 of 2.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Access array elements

myArray[1]	returns "Doe"
myArray[5]	returns true
myArray[6]	returns undefined

Nested objects and arrays

```
var myObject = {
  " ref ": {
    " fir st": 0,
    " las t": 1,
    " age ": 2,
    " sex ": 3,
    " sal ary ": 4,
    " reg ist ere d": 5
  },
  " jdo e1": [
    " Joh n",
    " Doe ",
    39,
    " mal e",
    70000,
    true
  ],
  " jsm ith 1": [
    " Jan e",
    " Smi th",
    42,
    " fem ale ",
    80000,
    true
  ]
};
```

Access nested elements

myObject.ref.first	returns 0
myObject.jdoe1	returns ["John", "-Doe", 39 ...]
myObject[2]	returns ["Jane", "-Smith", 42 ...]
myObject.jsmith1[3]	returns "female"
myObject[1][5]	returns true
myObject.jdoe1[myObject.ref.last]	returns "Doe"
myObject.jsmith1[myObject.ref.age]	returns 42



By **Mackan90096**
(Mackan90096)

cheatography.com/mackan90096/

Published 9th March, 2015.
Last updated 12th May, 2016.
Page 2 of 2.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>