

Data Types

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

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

boolean var true or false
 myBool = true;

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

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

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

undefined var variable with no value assigned
 myUndefined;

Objects

```
var myObject = {
  "first": "John",
  "last": "Doe",
  "age": 39,
  "sex": "male",
  "salary": 70000,
  "registered": true
};
```

Access object properties

myObject.sex returns "male"
 myObject["age"] returns 39
 myObject[0] returns "John"
 myObject.something returns undefined
 myObject[6] returns undefined

Array of objects

```
var myArray = [
  {
    "first": "John",
    "last": "Doe",
    "age": 39,
    "sex": "male",
    "salary": 70000,
    "registered": true
  },
  {
    "first": "Jane",
    "last": "Smith",
    "age": 42,
    "sex": "female",
    "salary": 80000,
    "registered": true
  },
  {
```

Array of objects (cont)

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

Access array elements

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

Arrays

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



By **Mackan90096**
 (Mackan90096)

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

Sponsored by **Readability-Score.com**
 Measure your website readability!
<https://readability-score.com>

Access array elements

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

Nested objects and arrays

```
var myObject = {
  "ref": {
    "first": 0,
    "last": 1,
    "age": 2,
    "sex": 3,
    "salary": 4,
    "registered": 5
  },
  "jdoe1": [
    "John",
    "Doe",
    39,
    "male",
    70000,
    true
  ],
  "jsmith1": [
    "Jane",
    "Smith",
    42,
    "female",
    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 9th March, 2015.
Page 2 of 2.

Sponsored by **Readability-Score.com**
Measure your website readability!
<https://readability-score.com>