

Variables

var	Same as ES5 (function-scoped).
let	Block-scoped (like ES6).
const	Same as let, but readonly.

Base types

Boolean	boolean : true false
Number	number : decimal hex binary octal
String	string : '...' "..." `...\${(...)}`
Array	type [] type [][] Array< type >
Tuple	[type , type , ...]
Enum	enum X { A = 1, B, C, ... }
Any	any
Void	function f(): void
Null	let n: null = null;
Undefined	let u: undefined = undefined;
Never	Use for functions , which never ends.

--strictNullChecks - check for assignment.

Destructuring

Array	<pre>i = [1, 2]; [f, s] = i; [f, s] = [s, f]; f([f, s]: [number, number]) [f, s, ...r] = [1, 2, 3, 4]; [f] = [1, 2, 3, 4]; [, s, , f] = [1, 2, 3, 4];</pre>
Object	<pre>o = { a: " f", b: 1, c: " b" } { a, b } = o; ({ a, b } = { a: " b", b: 1 }); { a, ...p } = o;</pre>
Property renaming	<pre>{ a: n1, b: n2 } = o; { a, b }: { a: string, b: number } = o;</pre>
Default values	<pre>{ a, b = 1 } = o;</pre>

Destructuring (cont)

Function declarations	f(o: { a: string, b?: number })
Spread	<pre>let f = [1, 2]; let s = [3, 4]; let b = [0, ...f, ...s, 5]; > [0, 1, 2, 3, 4, 5]</pre>

Interface

```
interface LabelledValue {
  ...
}
```

Optional parameters

```
name?: <type>;
```

Readonly properties

```
interface Point {
  readonly x: number;
  ...
}
```

ReadonlyArray<T>

Variables use **const** whereas properties use **readonly**.



By Denis Bednov
(denisbednov)

Not published yet.

Last updated 16th December, 2016.

Page 1 of 1.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>