

常见值类型

```
const a // undefined
const s = 'abc'
const n = 100
const b = true
const s = Symbol('s')
```



www.imooc.com



常见引用类型

```
const obj = { x: 100 }
const arr = ['a', 'b', 'c']

const n = null // 特殊引用类型，指针指向为空地址

// 特殊引用类型，但不用于存储数据，所以没有“拷贝、复制函数”这一说
function fn() {}
```



www.imooc.com



typeof 运算符

- ◆ 识别所有值类型
- ◆ 识别函数
- ◆ 判断是否是引用类型（不可再细分）



typeof 运算符

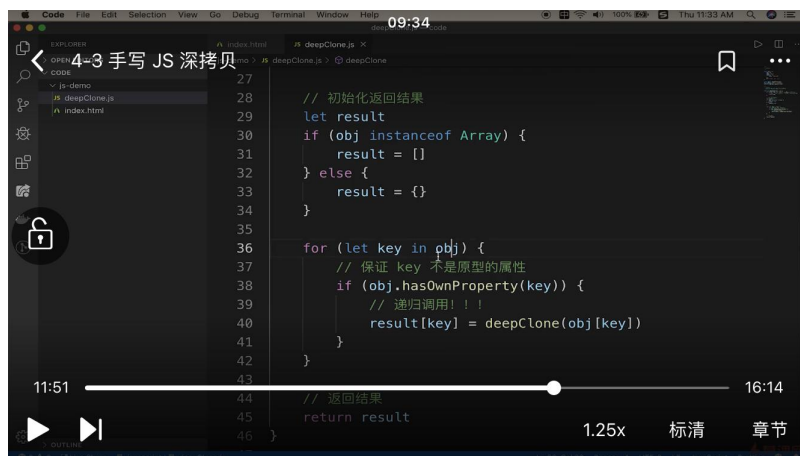
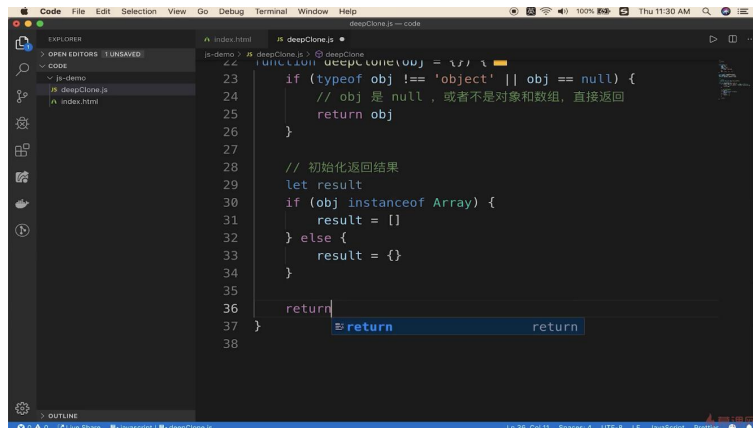
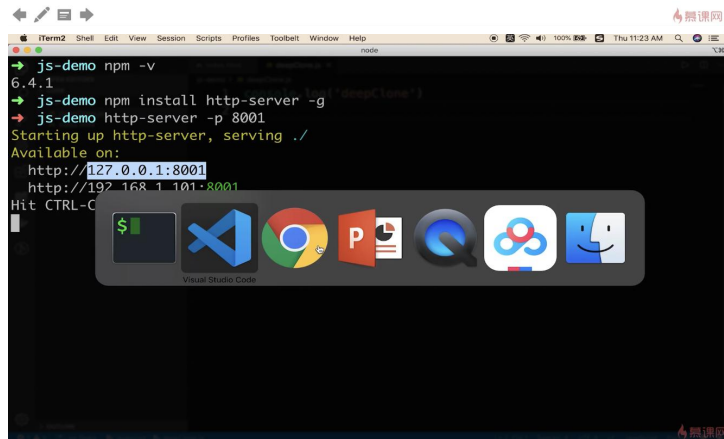
```
// 判断所有值类型
let a;           typeof a // 'undefined'
const str = 'abc';  typeof str // 'string'
const n = 100;     typeof n // 'number'
const b = true;    typeof b // 'boolean'
const s = Symbol('s');  typeof s // 'symbol'
```



typeof 运算符

```
// 能判断函数
typeof console.log // 'function'
typeof function () {} // 'function'

// 能识别引用类型 (不能再继续识别)
typeof null // 'object'
typeof ['a', 'b'] // 'object'
typeof { x: 100 } // 'object'
```



变量计算 – 类型转换

- ◆ 字符串拼接
- ◆ ==
- ◆ if 语句和逻辑运算

字符串拼接

```
const a = 100 + 10    // 110
const b = 100 + '10'  // '10010'
const c = true + '10' // 'true10'
```

== 运算符

```
100 == '100'    // true
0 == ''         // true
0 == false      // true
false == ''     // true
null == undefined // true
```

== 运算符

```
// 除了 == null 之外，其他都一律用 === ， 例如：

const obj = { x: 100 }
if (obj.a == null) { }
// 相当于：
// if (obj.a === null || obj.a === undefined) { }
```

if 语句和逻辑运算

- ◆ truly 变量：!!a === true 的变量
- ◆ falsely 变量：!!a === false 的变量

if 语句和逻辑运算

```
// 以下是 falsely 变量。除此之外都是 truly 变量
!!0 === false
!!NaN === false
!!'' === false
!!null === false
!!undefined === false
!!false === false
```

逻辑判断

```
console.log(10 && 0) // 0
console.log('' || 'abc') // 'abc'
console.log(!window.abc) // true
```