

题目 - 8

- ◆ 关于作用域和自由变量的场景题 - 1
- ◆ 判断字符串以字母开头，后面字母数字下划线，长度 6-30
- ◆ 关于作用域和自由变量的场景题 - 2



慕课网

关于作用域和自由变量的场景题 - 1

```
let i
for(i = 1; i <= 3; i++) {
  setTimeout(function(){
    console.log(i)
  }, 0)
}
// what?
```



慕课网

字符串 字母开头，后面字母数字下划线，长度 6-30

- ◆ `const reg = /^[a-zA-Z]\w{5,29}$/`
- ◆ 学习正则表达式的规则
- ◆ 手写常见的正则表达式

关于作用域和自由变量的场景题 - 2

```
let a = 100
function test() {
  alert(a)
  a = 10
  alert(a)
}
test()
alert(a)
// what?
```

题目 - 9

- ◆ 手写字符串 trim 方法，保证浏览器兼容性
- ◆ 如何获取多个数字中的最大值
- ◆ 如何用 JS 实现继承？

imooc

慕课网

手写字符串 trim 保证浏览器兼容性

```
String.prototype.trim= function (){  
    return this.replace(/^\s+/, '').replace(/\s+$/, '')  
}  
// （原型、this、正则表达式）
```

← ✎ ☰ →

慕课网

获取多个数字中的最大值

```
function max() {  
  const nums = Array.prototype.slice.call(arguments) // 变为数组  
  let max = 0  
  nums.forEach(n => {  
    if (n > max) {  
      max = n  
    }  
  })  
  return max  
}
```

慕课网

获取多个数字中的最大值

```
Math.max(10, 30, 20, 40)  
// 以及 Math.min
```

← ✎ ☰ →

慕课网

如何用 JS 实现继承

- ◆ class 继承
- ◆ prototype 继承



题目 - 10

- ◆ 如何捕获 JS 程序中的异常？
- ◆ 什么是 JSON ？
- ◆ 获取当前页面 url 参数



如何捕获 JS 中的异常？

```
try {  
    // todo  
} catch (ex) {  
    console.error(ex) // 手动捕获 catch  
} finally {  
    // todo  
}
```

```
// 自动捕获  
window.onerror = function (message, source, lineNom, colNom, error) {  
    // 第一，对跨域的 js，如 CDN 的，不会有详细的报错信息  
    // 第二，对于压缩的 js，还要配合 sourceMap 反查到未压缩代码的行、列  
}
```

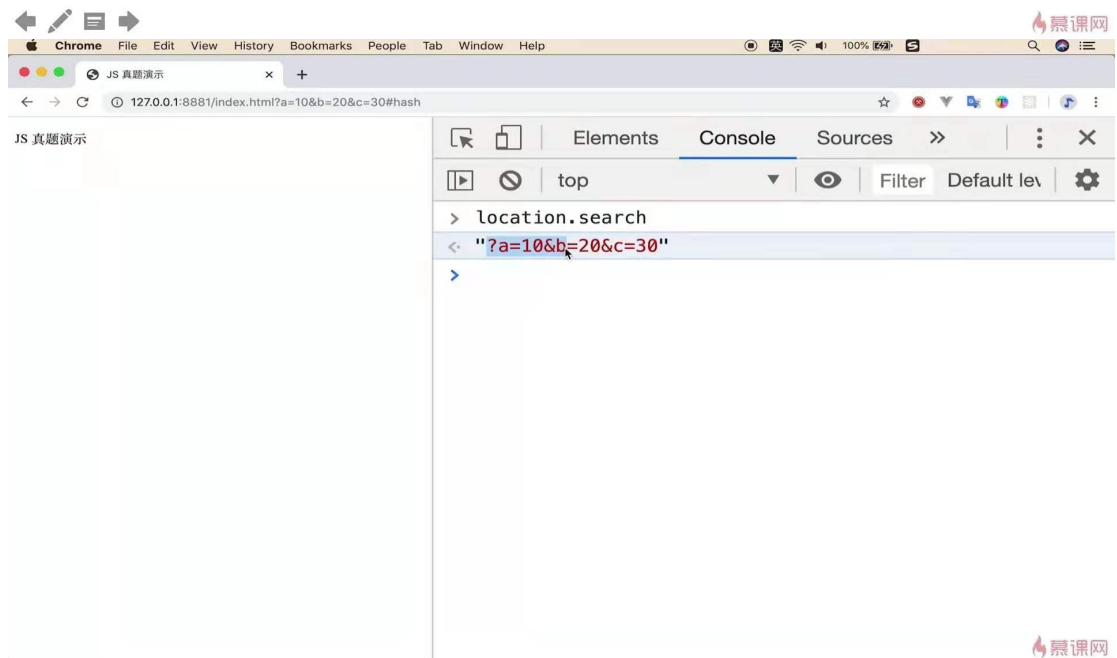
慕课网

什么是 JSON ？

- ◆ json 是一种数据格式，本质是一段字符串。
- ◆ json 格式和 JS 对象结构一致，对 JS 语言更友好
- ◆ window.JSON 是一个全局对象：JSON.stringify JSON.parse

获取当前页面 url 参数

- ◆ 传统方式，查找 location.search
- ◆ 新 API，URLSearchParams



EXPLORER

index.html JS query.js

code > questions-demo > JS query.js > query > res

```
1 // 传统方式
2 Complexity is 4 Everything is cool!
3 function query(name) {
4     const search = location.search.substr(1) // 类似 array.slice(
5     // search: 'a=10&b=20&c=30'
6     const reg = new RegExp(`(^&){name}=(^&*)(&|$)`,'i')
7     const res = search.match(reg)
8     if (res === null) {
9         return null
10    }
11    console.log(res)
12    return res[2]
13 }
14 query('b')
```

JS 真题演示

127.0.0.1:8881/index.html?a=10&b=20&c=30#hash

Elements Console

top

Filter Default lev

query.js:10

(4) ["a=10&", "", "10", "&", index: 0, input: "a=10&b=20&c=30", groups: undefined]

0: "a=10&"

1: "" a=10&

2: "10"

3: "&"

groups: undefined

index: 0

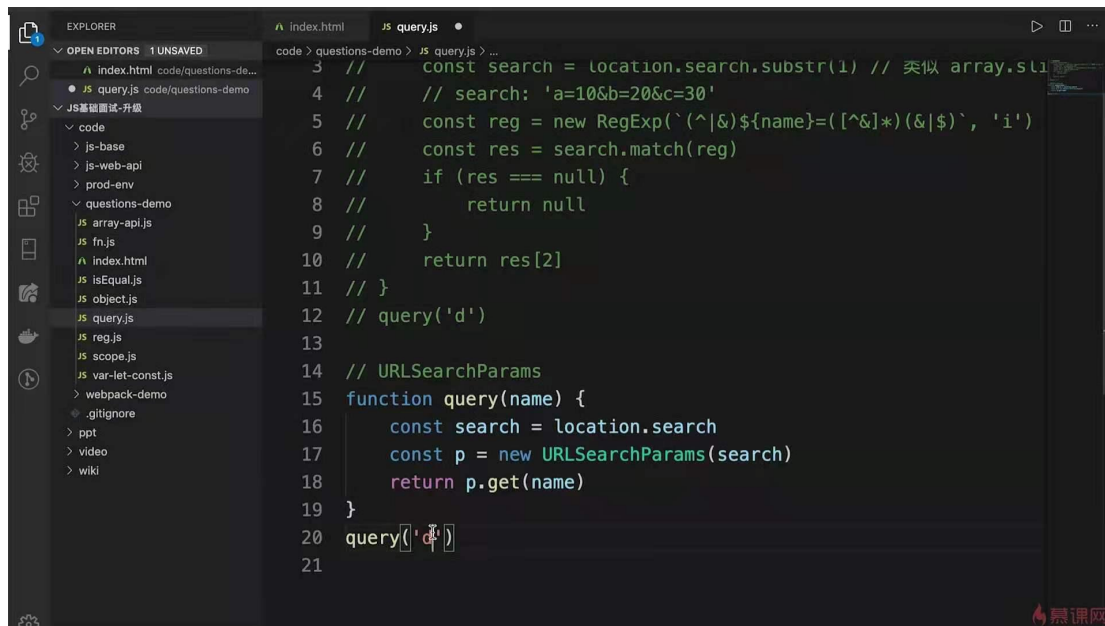
input: "a=10&b=20&c=30"

length: 4

__proto__: Array(0)

GET http://127.0.0.1:8881/favicon.ico:8881/favicon.ico:1 on.ico 404 (Not Found)

慕课网



```
3 // const search = location.search.substr(1) // 类似 array.slice
4 // // search: 'a=10&b=20&c=30'
5 // const reg = new RegExp(`(^|&){name}=(^&)*(&|$)` , 'i')
6 // const res = search.match(reg)
7 // if (res === null) {
8 //     return null
9 // }
10 // return res[2]
11 // }
12 // query('d')
13
14 // URLSearchParams
15 function query(name) {
16     const search = location.search
17     const p = new URLSearchParams(search)
18     return p.get(name)
19 }
20 query('d')
21
```