各种console方式

console.log的其他使用方式

console.log()

```
1    let obj = {
2        name:'黄土豆',
3        age:2,
4    }
5    console.log('这是一个对象 %o',obj);
```


常见的占位符 %o (这是字母O，不是0)，它接受对象，%s 接受字符串，%d 表示小数或整数。

```
1    console.log('I am a %cbutton', 'color: white; background-color: orange; padding: 2px 5px; border-radius: 2px');
```


另一个有趣的是 %c，这可能与你所想不太相同，它实际上是CSS值的占位符。使用%c占位符时，对应的后面的参数必须是CSS语句，用来对输出内容进行CSS渲染。

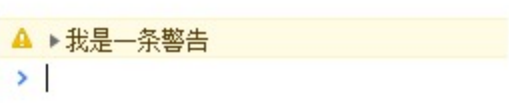## console.dir()

console.log一个元素只能拿到元素内容
而console.dir能拿到元素的详细信息

console.warn()

输出警告信息，更显眼

console.table()

```
1    const data = [{
2        id: "7cb1-e041b126-f3b8",
3        seller: "WAL0412",
4        buyer: "WAL3023",
5        price: 203450,
6        time: 1539688433
7        },
8        {
9        id: "1d4c-31f8f14b-1571",
10       seller: "WAL0452",
11       buyer: "WAL3023",
12       price: 348299,
13       time: 1539688433
14       },
15       {
16       id: "b12c-b3adf58f-809f",
17       seller: "WAL0012",
18       buyer: "WAL2025",
19       price: 59240,
20       time: 1539688433
21       }];
```

这一段数据如果用console.log输出时是这样的

但是如果用console.table输出却是这样的

console.table() 只能处理最多1000行，因此它可能不适合所有数据集。

console.count()

另一个具有特殊用途的计数器，count只是作为一个计数器，或者作为一个命名计数器，可以统计代码被执行的次数。

```
1    for(let i = 0; i < 10000; i++) {
2    if(i % 2) {
3    console.count('odds');
4    }
5    if(!(i % 5)) {
6    console.count('multiplesOfFive');
7    }
8    if(isPrime(i)) {
9    console.count('prime');
10   }
11   }
```

执行后我们会得到一个列表：

```
1    odds: 1
2    odds: 2
3    prime: 1
4    odds: 3
5    multiplesOfFive: 1
6    prime: 2
7    odds: 4
8    prime: 3
9    odds: 5
10   multiplesOfFive: 2
11   ...
```