

Array.prototype.reduce() 用法

`reduce()` 方法对数组中的每个元素按序执行一个提供的 reducer 函数，每一次运行 reducer 会将先前元素的计算结果作为参数传入，最后将其结果汇总为单个返回值。

```
// 用法
reduce(callbackFn);
reduce(callbackFn, initialValue);
```

1. 求数值数组中值的总和

```
// 无初始值
const array = [15, 16, 17, 18, 19];
function reducer(accumulator, currentValue, index) {
  const returns = accumulator + currentValue;
  console.log(`accumulator: ${accumulator}, currentValue: ${currentValue}, index: ${index}, returns: ${returns}`);
  return returns;
}
array.reduce(reducer);
```

	accumulator	currentValue	index	返回值
第一次调用	15	16	1	31
第二次调用	31	17	2	48
第三次调用	48	18	3	66
第四次调用	66	19	4	85

无初始值时，index 从 1 开始，因为第一次调用函数第一个参数使用数组中第一个成员的值，函数第二个参数使用数组中第二个成员的值，调用次数比数组长度少一。array 参数在整个过程中始终不会改变——它始终是 [15, 16, 17, 18, 19]。reduce() 返回的值将是最后一次回调返回值（85）。

```
// 有初始值
const array = [15, 16, 17, 18, 19];
function reducer(accumulator, currentValue, index) {
  const returns = accumulator + currentValue;
  console.log(`accumulator: ${accumulator}, currentValue: ${currentValue}, index: ${index}, returns: ${returns}`);
  return returns;
}
array.reduce(reducer, 10);
```

	accumulator	currentValue	index	返回值
第一次调用	10	15	0	25
第二次调用	25	16	1	41
第三次调用	41	17	2	58
第四次调用	58	18	3	76
第五次调用	76	19	4	95

2. 按属性对对象进行分组

```
let list = [
  {
    "meetingId": 718,
    "meetingName": "测试注销",
    "beginTime": "2023-09-08",
  },
  {
    "meetingId": 698,
    "meetingName": "测试会议 1",
    "beginTime": "2023-08-25",
  },
  {
    "meetingId": 680,
    "meetingName": "测试新会议",
    "beginTime": "2023-08-03",
  },
  {
    "meetingId": 676,
    "meetingName": "测试连屏会议",
    "beginTime": "2023-07-31",
  },
  {
    "meetingId": 668,
    "meetingName": "刚回家成华",
    "beginTime": "2023-07-27",
  }
]

function groupByList(objectArray, property) {
  return objectArray.reduce((acc, obj) => {
    const key = obj[property];
    const curGroup = acc[key] ?? [];

    return { ...acc, [key]: [...curGroup, obj] };
  }, {});
}

const groupedList = groupByList(people, "beginTime");
console.log(groupedList);
```

```
// {  
//   "2023-09-08": [  
//     {  
//       "meetingId": 718,  
//       "meetingName": "测试注销",  
//       "beginTime": "2023-09-08"  
//     }  
//   ],  
//   "2023-08-25": [  
//     {  
//       "meetingId": 698,  
//       "meetingName": "测试会议 1",  
//       "beginTime": "2023-08-25"  
//     }  
//   ],  
//   "2023-08-03": [  
//     {  
//       "meetingId": 680,  
//       "meetingName": "测试新会议",  
//       "beginTime": "2023-08-03"  
//     }  
//   ],  
//   "2023-07-31": [  
//     {  
//       "meetingId": 676,  
//       "meetingName": "测试连屏会议",  
//       "beginTime": "2023-07-31"  
//     }  
//   ],  
//   "2023-07-27": [  
//     {  
//       "meetingId": 668,  
//       "meetingName": "刚回家成华",  
//       "beginTime": "2023-07-27"  
//     }  
//   ]  
// }
```