



黑马程序员™
www.itheima.com

传智播客旗下
高端IT教育品牌

JavaScript 数组

目录 Contents

- ◆ 数组的概念
- ◆ 创建数组
- ◆ 获取数组中的元素
- ◆ 遍历数组
- ◆ 数组中新增元素
- ◆ 数组案例

1. 数组的概念

问：之前学习的数据类型，只能存储一个值。如果我们想存储班级中所有学生的姓名，那么该如何存储呢？

答：可以使用数组(Array)。数组可以把一组相关的数据一起存放，并提供方便的访问(获取)方式。

问：什么是数组呢？

答：数组是指**一组数据的集合**，其中的每个数据被称作**元素**，在数组中可以**存放任意类型的元素**。数组是一种将**一组数据存储在单个变量名下**的优雅方式。

```
// 普通变量一次只能存储一个值
var num = 10;

// 数组一次可以存储多个值
var arr = [1,2,3,4,5];
```

目录 Contents

- ◆ 数组的概念
- ◆ 创建数组
- ◆ 获取数组中的元素
- ◆ 遍历数组
- ◆ 数组中新增元素
- ◆ 数组案例

2. 创建数组

2.1 数组的创建方式

JS 中创建数组有两种方式：

- 利用 new 创建数组
- 利用数组字面量创建数组



2. 创建数组

2.2 利用 new 创建数组

```
var 数组名 = new Array() ;  
var arr = new Array();    // 创建一个新的空数组
```

- 这种方式暂且了解，等学完对象再看
- 注意 Array () ，A 要大写

2. 创建数组

2.3 利用数组字面量创建数组

```
//1. 使用数组字面量方式创建空的数组  
var 数组名 = [];  
  
//2. 使用数组字面量方式创建带初始值的数组  
var 数组名 = ['小白', '小黑', '大黄', '瑞奇'];
```

- 数组的字面量是方括号 []
- 声明数组并赋值称为数组的初始化
- 这种字面量方式也是我们以后**最多使用的方式**

2. 创建数组

2.4 数组元素的类型

数组中可以存放**任意类型**的数据，例如字符串，数字，布尔值等。

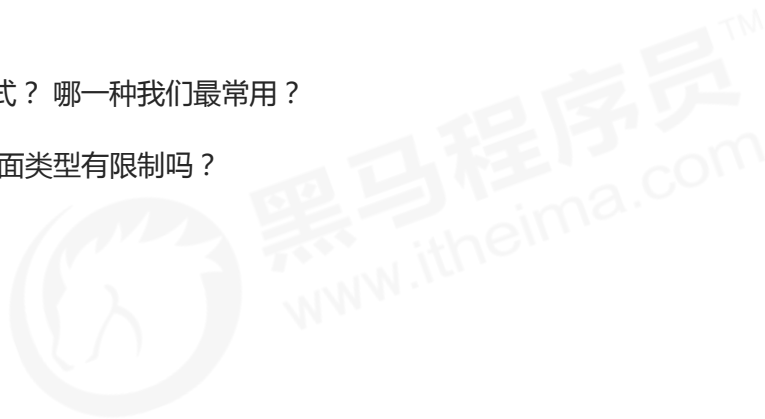
```
var arrStus = ['小白',12,true,28.9];
```



■ 2. 创建数组

2.5 pink老师提问

1. 数组的作用是什么？
2. 创建数组我们有哪两种方式？哪一种我们最常用？
3. 什么是数组元素？数组里面类型有限制吗？



目录

Contents

- ◆ 数组的概念
- ◆ 创建数组
- ◆ 获取数组中的元素
- ◆ 遍历数组
- ◆ 数组中新增元素
- ◆ 数组案例



3. 获取数组元素

3.1 数组的索引

索引 (下标) : 用来访问数组元素的序号 (数组下标从 0 开始)。

```
var arr = ['小白', '小黑', '大黄', '瑞奇'];
```

索引号: 0 1 2 3

数组可以通过**索引**来访问、设置、修改对应的数组元素，我们可以通过“**数组名[索引]**”的形式来获取数组中的元素。

这里的**访问**就是获取得到的意思

```
// 定义数组
var arrStus = [1,2,3];
// 获取数组中的第2个元素
alert(arrStus[1]);
```

3. 获取数组元素



课堂练习：数组练习

定义一个数组，里面存放星期一、星期二.....直到星期日（共7天），在控制台输出：星期日，请同学们自己动手完成。



目录

Contents

- ◆ 数组的概念
- ◆ 创建数组
- ◆ 获取数组中的元素
- ◆ 遍历数组
- ◆ 数组中新增元素
- ◆ 数组案例

4. 遍历数组

问：数组中的每一项我们怎么取出来？

答：可以通过“**数组名[索引号]**”的方式一项项的取出来。

```
var arr = ['red', 'green', 'blue'];  
console.log(arr[0]) // red  
console.log(arr[1]) // green  
console.log(arr[2]) // blue
```

问：怎么把数组里面的元素全部取出来？

规律：

从代码中我们可以发现，从数组中取出每一个元素时，代码是重复的，有所不一样的是**索引值在递增**

答案就是 循环

4. 遍历数组

遍历: 就是把数组中的每个元素从头到尾都访问一次（类似我们每天早上学生的点名）。

我们可以通过 **for** 循环索引遍历数组中的每一项

```
var arr = ['red', 'green', 'blue'];  
for (var i = 0; i < arr.length; i++) {  
    console.log(arr[i]);  
}
```

4.遍历数组

4.1 数组的长度

使用“**数组名.length**”可以访问数组元素的数量（数组长度）。

```
var arrStus = [1,2,3];  
alert(arrStus.length); // 3
```

注意：

- ① 此处数组的长度是**数组元素的个数**，不要和数组的**索引号**混淆。
- ② 当我们数组里面的元素个数发生了变化，这个 length 属性跟着一起变化。

4. 遍历数组



课堂案例 1：遍历数组

请将 ["关羽", "张飞", "马超", "赵云", "黄忠", "刘备", "姜维"]; 数组里的元素依次打印到控制台。

```
var arr = ["关羽", "张飞", "马超", "赵云", "黄忠", "刘备", "姜维"];  
// 遍历 从第一个到最后一个  
for(var i = 0; i < arr.length; i++ ) {  
    console.log( arr[i] );  
}
```

4. 遍历数组

Pink 老师提问 ?

1. 什么是遍历 ?
2. 我们通过什么来遍历数组里面的元素 ?
3. for 里面的 i 是什么 ? 当什么使用 ? for里面的数组元素怎么写 ? **i是计数器，当索引号使用，arr[i] 是数组元素 第i个数组元素**
4. 怎么获取数组的长度 ?
5. 数组索引号和数组长度有什么关系 ? **索引号从0开始， 数组长度是 元素个数**

4. 遍历数组



课堂案例 2：数组求和及平均值

求数组 [2,6,1,7, 4] 里面所有元素的和以及平均值。

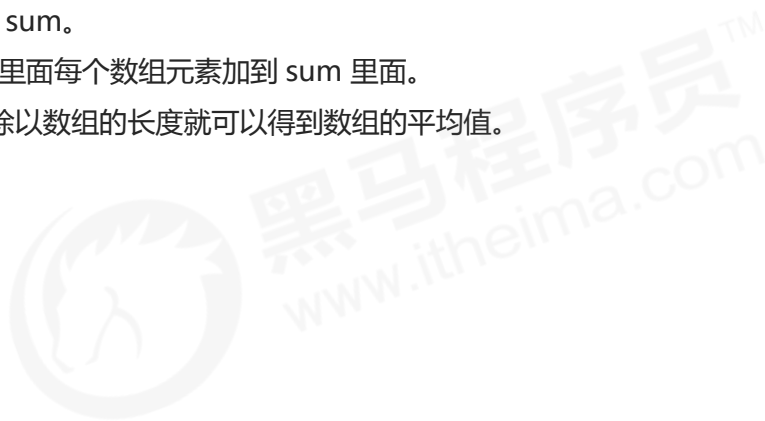


4. 遍历数组



案例分析

- ① 声明一个求和变量 sum。
- ② 遍历这个数组，把里面每个数组元素加到 sum 里面。
- ③ 用求和变量 sum 除以数组的长度就可以得到数组的平均值。



4. 遍历数组



实现代码

```
var arr = [2, 6, 1, 7, 4];  
var sum = 0;  
var average = 0;  
for (var i = 0; i < arr.length; i++) {  
    sum += arr[i];  
}  
average = sum / arr.length;  
console.log('这组数的和是:' + sum);  
console.log('这组数的平均值是:' + average);
```

4. 遍历数组



课堂案例 3：数组最大值

求数组[2,6,1,77,52,25,7]中的最大值。



4. 遍历数组



案例分析

- ① 声明一个保存最大元素的变量 max。
- ② 默认最大值可以取数组中的第一个元素。
- ③ 遍历这个数组，把里面每个数组元素和 max 相比较。
- ④ 如果这个数组元素大于max 就把这个数组元素存到 max 里面，否则继续下一轮比较。
- ⑤ 最后输出这个 max。

4. 遍历数组



实现代码

```
var arrNum = [2,6,1,77,52,25,7];  
var maxNum = arrNum[0]; // 用来保存最大元素,默认最大值是数组中的第一个元素  
// 从0 开始循环数组里的每个元素  
for(var i = 0;i< arrNum.length; i++){  
    // 如果数组里当前循环的元素大于 maxNum , 则保存这个元素和下标  
    if(arrNum[i] > maxNum){  
        maxNum = arrNum[i]; // 保存数值到变量 maxNum  
    }  
}
```


4. 遍历数组



课堂案例 4：数组转换为字符串

要求：将数组 ['red', 'green', 'blue', 'pink'] 里面的元素转换为字符串

输出：'redgreenbluepink'

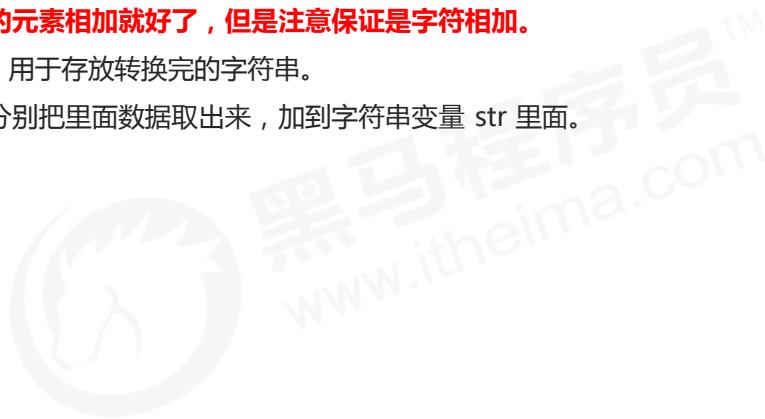


4. 遍历数组



案例分析

- ① 思路：**就是把里面的元素相加就好了，但是注意保证是字符相加。**
- ② 需要一个新变量 str 用于存放转换完的字符串。
- ③ 遍历原来的数组，分别把里面数据取出来，加到字符串变量 str 里面。



4. 遍历数组



实现代码

```
var arr = ['red', 'green', 'blue', 'pink'];  
var str = '';  
for (var i = 0; i < arr.length; i++) {  
    str += arr[i];  
}  
console.log(str);
```

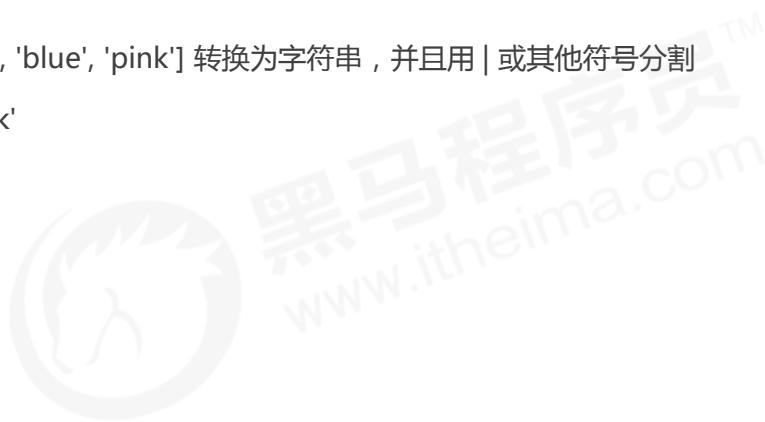
4. 遍历数组



课堂案例 5：数组转换为分割字符串

要求：将数组 ['red', 'green', 'blue', 'pink'] 转换为字符串，并且用 | 或其他符号分割

输出：'red|green|blue|pink'

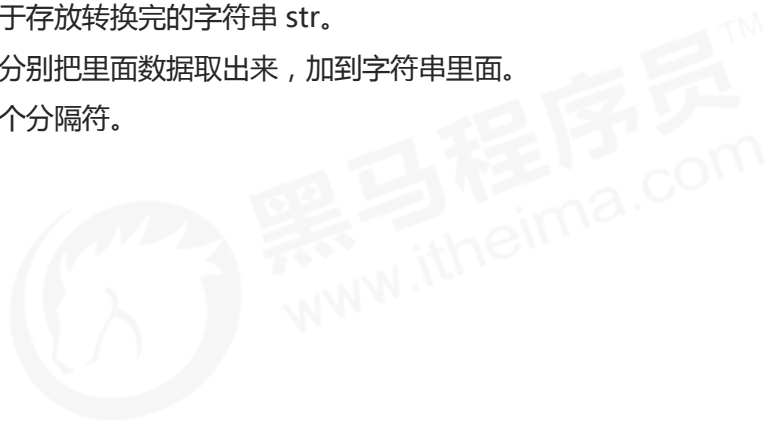


4. 遍历数组



案例分析

- ① 需要一个新变量用于存放转换完的字符串 str。
- ② 遍历原来的数组，分别把里面数据取出来，加到字符串里面。
- ③ 同时在后面多加一个分隔符。



4. 遍历数组



实现代码

```
var arr = ['red', 'green', 'blue', 'pink'];  
var str = '';  
var separator = '|'  
for (var i = 0; i < arr.length; i++) {  
    str += arr[i] + separator;  
}  
console.log(str);
```

目录

Contents

- ◆ 数组的概念
- ◆ 创建数组
- ◆ 获取数组中的元素
- ◆ 遍历数组
- ◆ 数组中新增元素
- ◆ 数组案例

5. 数组中新增元素

5.1 通过修改 length 长度新增数组元素

- 可以通过修改 length 长度来实现数组扩容的目的
- length 属性是可读写的

```
var arr = ['red', 'green', 'blue', 'pink'];  
arr.length = 7;  
console.log(arr);  
console.log(arr[4]);  
console.log(arr[5]);  
console.log(arr[6]);
```

其中索引号是 4, 5, 6 的空间没有给值, 就是声明变量未给值, 默认值就是 **undefined**。

```
► (7) ["red", "green", "blue", "pink", empty × 3]
```

```
undefined
```

```
undefined
```

```
undefined
```




5. 数组中新增元素

5.2 通过修改数组索引新增数组元素

- 可以通过修改数组索引的方式追加数组元素
- 不能直接给数组名赋值，否则会覆盖掉以前的数据

```
var arr = ['red', 'green', 'blue', 'pink'];  
arr[4] = 'hotpink';  
console.log(arr);
```

这种方式也是我们最常用的一种方式。

5. 数组中新增元素



课堂案例 1：数组新增元素

新建一个数组，里面存放10个整数（1~10），要求使用循环追加的方式输出：[1,2,3,4,5,6,7,8,9,10]



5. 数组中新增元素



案例分析

- ① 使用循环来追加数组。
- ② 声明一个空数组 arr。
- ③ 循环中的计数器 i 可以作为数组元素存入。
- ④ 由于数组的索引号是从0开始的，因此计数器从 0 开始更合适，存入的数组元素要+1。

5. 数组中新增元素



实现代码

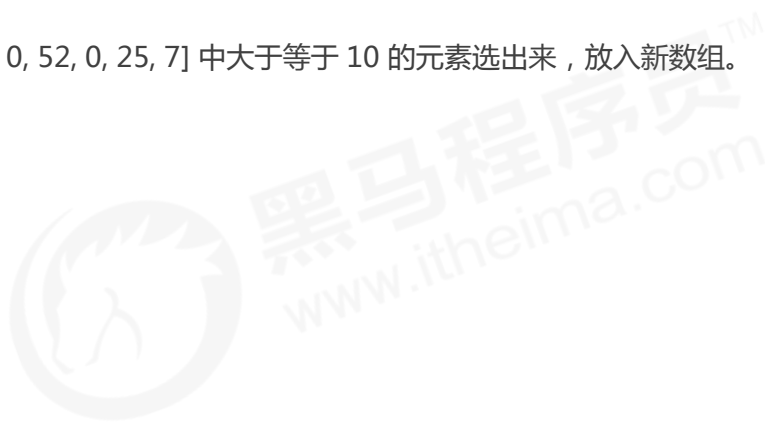
```
var arr = [];  
for (var i = 0; i < 10; i++) {  
    arr[i] = i + 1;  
}  
console.log(arr);
```

5. 数组中新增元素



课堂案例 2：筛选数组

要求：将数组 [2, 0, 6, 1, 77, 0, 52, 0, 25, 7] 中大于等于 10 的元素选出来，放入新数组。



5. 数组中新增元素



案例分析

- ① 声明一个新的数组用于存放新数据。
- ② 遍历原来的数组，找出大于等于 10 的元素。
- ③ 依次追加给新数组 newArr。



5. 数组中新增元素



实现代码 1

```
var arr = [2, 0, 6, 1, 77, 0, 52, 0, 25, 7];  
var newArr = [];  
// 定义一个变量 用来计算 新数组的索引号  
var j = 0;  
for (var i = 0; i < arr.length; i++) {  
    if (arr[i] >= 10) {  
        // 给新数组  
        newArr[j] = arr[i];  
        // 索引号 不断自加  
        j++;  
    }  
}  
  
console.log(newArr);
```

5. 数组中新增元素



实现代码 2

```
var arr = [2, 0, 6, 1, 77, 0, 52, 0, 25, 7];
var newArr = [];
for (var i = 0; i < arr.length; i++) {
    if (arr[i] >= 10) {
        // 给新数组
        newArr[newArr.length] = arr[i];
    }
}
console.log(newArr);
```


目录

Contents

- ◆ 数组概念
- ◆ 创建数组
- ◆ 获取数组元素
- ◆ 遍历数组
- ◆ 数组中新增元素
- ◆ 数组案例

6. 数组案例



课堂案例 1：删除指定数组元素

要求：将数组[2, 0, 6, 1, 77, 0, 52, 0, 25, 7]中的 0 去掉后，形成一个不包含 0 的新数组。

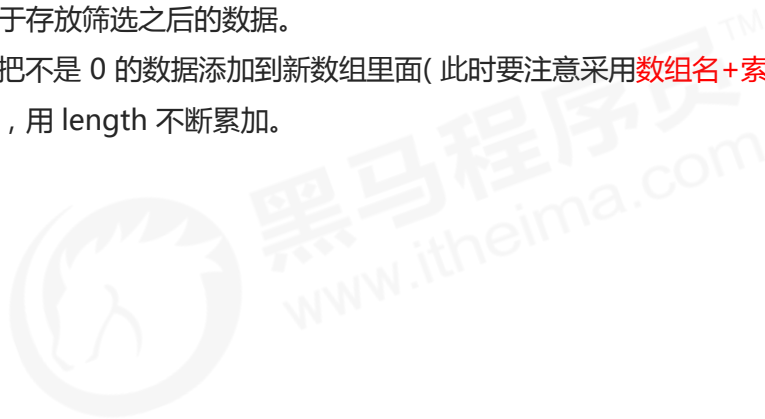


6. 数组案例



案例分析

- ① 需要一个新数组用于存放筛选之后的数据。
- ② 遍历原来的数组，把不是 0 的数据添加到新数组里面(此时要注意采用数组名+索引的格式接收数据)。
- ③ 新数组里面的个数，用 length 不断累加。



6. 数组案例



实现代码

```
var arr = [2, 0, 6, 1, 77, 0, 52, 0, 25, 7];  
var newArr = []; // 空数组的默认的长度为 0  
// 定义一个变量 i 用来计算新数组的索引号  
for (var i = 0; i < arr.length; i++) {  
    // 找出大于 10 的数  
    if (arr[i] != 0) {  
        // 给新数组  
        // 每次存入一个值, newArr长度都会 +1  
        newArr[newArr.length] = arr[i];  
    }  
}  
console.log(newArr);
```

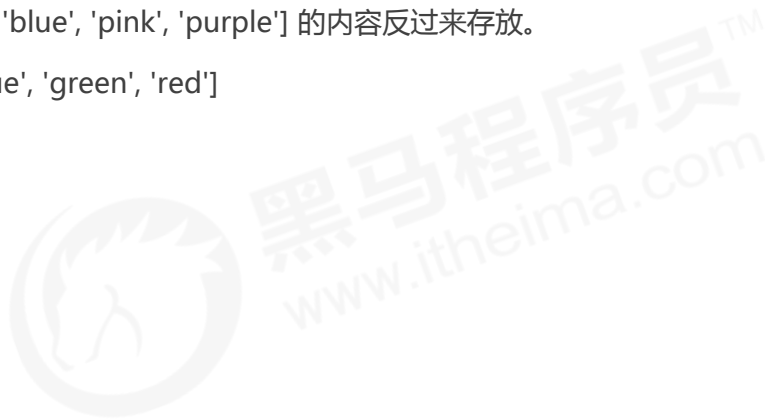
6. 数组案例



课堂案例 2：翻转数组

要求: 将数组 ['red', 'green', 'blue', 'pink', 'purple'] 的内容反过来存放。

输出： ['purple', 'pink', 'blue', 'green', 'red']



6. 数组案例



案例分析

核心： 把 arr 的 最后一个元素 取出来 给 新数组 作为第一个

```
['red', 'green', 'blue', 'pink', 'purple'];
```

```
['purple', 'pink', 'blue', 'green', 'red'];
```

把第 4 个 给 新数组 第 0 个

把第 3 个 给 新数组 第 1 个

把第 2 个 给 新数组 第 2 个

把第 1 个 给 新数组 第 3 个

把第 0 个 给 新数组 第 4 个

43210 是 长度(5) - 1

01234 正好就是

newArr.length

6. 数组案例



实现代码

```
var arr = ['red', 'green', 'blue', 'pink', 'purple'];  
var newArr = [];  
for (var i = 0; i < arr.length; i++) {  
    // newArr 是接收方, arr 是输送方  
    newArr[i] = arr[arr.length - i - 1];  
}  
console.log(newArr);
```

6. 数组案例



课堂案例 3：数组排序（冒泡排序）

我们先复习下如何把2个变量交换数据

```
// 利用第三个变量  
var num1 = 10;  
var num2 = 20;  
var temp = num1;  
num1 = num2;  
num2 = temp;  
console.log(num1, num2);
```

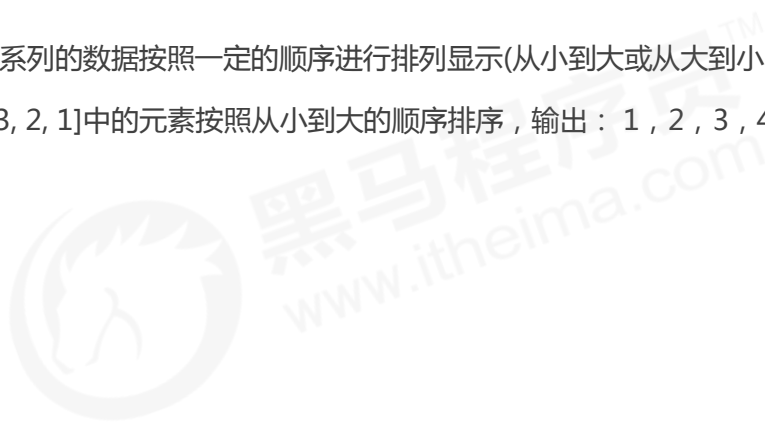

6. 数组案例



课堂案例 3：数组排序（冒泡排序）

冒泡排序：是一种算法，把一系列的数据按照一定的顺序进行排列显示(从小到大或从大到小)。

例如，我们可以将数组 [5, 4, 3, 2, 1] 中的元素按照从小到大的顺序排序，输出：1, 2, 3, 4, 5



6. 数组案例



案例分析

5 4 3 2 1

第1趟:

4 3 2 1 5

第2趟:

3 2 1 4 5

第3趟:

2 1 3 4 5

第4趟:

1 2 3 4 5

1. 一共需要的趟数 我们用外层for 循环

5个数据我们一共需要走4趟

长度就是 数组长度 减去 1 $arr.length - 1$

2. 每一趟交换次数 我们 用里层 for循环

第一趟 交换 4次

第二趟 交换 3次

第三趟 交换 2次

第四趟 交换 1次

长度就是 数组长度 减去 次数

但是我们次数 是从 0次开始的 所以 最终 $arr.length - i - 1$

3. 交换2个变量就好了

6. 数组案例



实现代码

```
var arr = [5, 4, 3, 2, 1];  
for (var i = 0; i < arr.length - 1; i++) {  
    for (var j = 0; j < arr.length - i - 1; j++) {  
        if (arr[j] > arr[j + 1]) {  
            var temp = arr[j];  
            arr[j] = arr[j + 1];  
            arr[j + 1] = temp;  
        }  
    }  
}  
console.log(arr);
```



黑马程序员

www.itheima.com

传智播客旗下高端IT教育品牌