



小生方勤

2019年11月05日 阅读 2146

已关注

【前端词典】arr.filter(parseInt) 考查点为何

这一类的面试题目时不时出现，那么它的考点在哪里呢？

其实这类题目的很清楚，即考察：

- 数组方法的基础知识
- 数值转换方法的基础知识

也就是考察 **基础知识** 的扎实程度，换句话说这是一道基础题。

map()、filter()、reduce() 等数组方法

说明一下：**map()**、**filter()** 是 ES5 为数组定义的迭代方法，**reduce()** 是 ES5 为数组定义的归并方法。这些方法并不是 ES6 新增的语法（之前在一场面试中，面试官很肯定的说这是 ES6 新增的方法）。

- 你可以在《JavaScript 高级程序设计-第三版》96-97 页中看到明确的说明。

语法：

```
array.filter(callback[,thisObject]);  
// array.map(callback[,thisObject]);  
// array.reduce(callback[,initialValue])
```

复制代码

都不会改变原始数组。

参数：

```
callback: 要对每个数组元素执行的回调函数。  
thisObject: 在执行回调函数时定义的 this 对象(没有传递或者为 null, 将会使用全局对象)。  
// initialValue: 作为第一次调用 callback 函数时的第一个参数的值。
```

复制代码

map()

对原数组中的每个元素按顺序执行一次指定的函数（即 callback），callback 每次执行后的返回值（包括 undefined）组合起来形成一个新数组。

回调函数可以有三个参数：当前元素，当前元素的索引 **[可选]** 和当前的数组对象 **[可选]**。

filter()

对原数组中的每个元素执行一次指定的函数（即 callback），并且创建一个新的数组，该数组元素是所有回调函数执行时返回值为 true 的原数组元素。

回调函数可以有三个参数：当前元素，当前元素的索引 **[可选]** 和当前的数组对象 **[可选]**。

reduce()

对原数组中的每个元素执行一次指定的函数（即 callback），返回最后一次回调返回值。

回调函数可以有四个参数：累计器，当前元素，当前元素的索引 **[可选]** 和当前的数组对象 **[可选]**。

提醒：reduce() 区别于 map()/filter() 的地方就在回调函数的第一个参数 - **accumulator: 累计器**。

parseInt()、parseFloat() 两个数值转换方法

parseInt 语法

```
parseInt(string, radix)
```

复制代码

将一个字符串 string 转换为 radix（默认为 10）进制的整数，radix 为介于 2-36 之间的数。

注意：string 将看作是一个数的 n 进制表示，返回的值为十进制。

返回值：

- 如果被解析参数的第一个字符无法被转化成数值类型，则返回 **NaN**
- 如果 radix 小于 2 或者大于 36，则返回 **NaN**

在 radix 为 undefined，或者 radix 为 0 或者没有指定的情况下，JavaScript 作如下处理：

parseInt 语法

```
parseInt(string, radix)
```

复制代码

关于作者



小生方勤

打杂 @ tencent



获得点赞 13,150



文章被阅读 415,601

你可能感兴趣的小册



基于 Swoole 的联机对战游戏实践

241人已购买

试读>



深入理解NLP的中文分词：从原理到实践

206人已购买

试读>



下载掘金客户端

一个帮助开发者成长的社区



相关文章

【前端词典】提高幸福感的 9 个 CSS 技巧

👤 1706 🗨 136

【前端词典】5 种滚动吸顶实现方式的比较[性能升级版]

👤 1047 🗨 74

【前端词典】分享 8 个有趣且实用的 API

👤 1084 🗨 79

【前端词典】进阶必备的网络基础（上）

👤 800 🗨 43

【前端词典】如何向老板解释反向代理

👤 527 🗨 154

目录

• **map()、filter()、reduce() 等...**

- map()
- filter()
- reduce()

• **parseInt()、parseFloat() 两个...**

- parseInt 语法
- parseFloat 语法

• **解释 arr.filter(parseInt)**

• **总结**

将一个字符串 string 转换为 radix (值默认为 10) 进制的整数, radix 为介于 2-36 之间的数。

注意 : string 将看作是一个数的 n 进制表示, 返回的值为十进制。

返回值 :

- 如果被解析参数的第一个字符无法被转化成数值类型, 则返回 **NaN**
- 如果 radix 小于 2 或者大于 36, 则返回 **NaN**

在 radix 为 undefined, 或者 radix 为 0 或者没有指定的情况下, JavaScript 作如下处理 :

- 如果字符串 string 以 "0x" 或者 "0X" 开头, 则 radix 是 16 (16进制).
- 如果字符串 string 以 "0" 开头, radix 是 8 (八进制) 或者 10 (十进制), 具体值由实现环境决定. ECMAScript 5 规定使用 10, 但是并不是所有的浏览器都遵循这个规定. 因此, 永远都要明确给出 radix 参数的值
- 如果字符串 string 以其它任何值开头, 则 radix 是 10 (十进制)。

parseFloat 语法

```
parseFloat(string)
```

只有一个参数, 用作将给定值解析成浮点数 (如果给定值不能被转换成数值, 则会返回

2、经历如下过程 :

```
- parseInt(2,0) // 结果为 2, 转成 Boolean 为 true
- parseInt(8,1) // 结果为 NaN, 转成 Boolean 为 false
- parseInt(11,2) // 结果为 3, 转成 Boolean 为 true
- parseInt(18,3) // 结果为 3, 转成 Boolean 为 true
```

所以最后返回 `[2, 11, 10]`。

总结

基础很重要, 这是送分题。

关注下面的标签, 发现更多相似文章

JavaScript 前端



小生方勤

发布了 32 篇专栏 · 获得点赞 13,150 · 获得阅读 415,601

已关注

[安装掘金浏览器插件](#)
打开新标签页发现好内容, 掘金、GitHub、Dribbble、ProductHunt 等站点内容轻松获取。快来安装掘金浏览器插件获取高质量内容吧!



小生方勤

发布了 32 篇专栏 · 获得点赞 13,150 · 获得阅读 415,601

已关注

[安装掘金浏览器插件](#)
打开新标签页发现好内容, 掘金、GitHub、Dribbble、ProductHunt 等站点内容轻松获取。快来安装掘金浏览器插件获取高质量内容吧!

评论



Reference_Error

输入评论...



Reference_Error

FE @ 首席摸鱼官

1月前

👍 0 回复



我对花儿笑

前端工程师 @ 美团点评

更深入的用法, 试试[1, 2, '11e', '1e1', 'e11'].map(parseInt)

2月前

👍 0 回复



Tylerrrkd

FE @ 卓越教育

送分题。developer.mozilla.org/zh-CN/docs/Web... => 使用技巧案例

2月前

👍 0 回复



leoljustsave

前端实习生

打 call, 但是 ECMAScript 5 那个应该是勘误吧 ... MDN 都是 ECMAScript 2015 ...

2月前

👍 0 回复



小生方勤

(作者) 打杂 @ tencent

回复 leoljustsave: 网上的一些说法是错误的, 你可一看商程或者看 ecma-international.org/ecma-262/5.1, ployfill 是兼容更低的版本

2月前

👍 0 回复



leoljustsave

前端实习生

回复 小生方勤: 链接里面是说有通用, 但是具体实现还要看情况, 所以 MDN 里面对应的是 Draft, 然后在 ES6 里面作为标准. 主要是有点纠结下次面试官期待地问我用过哪些 ES6 特性的时候我该不该说这个

2月前

👍 0 回复



自由灵魂

总结得很清晰

2月前

👍 1 0 回复

相关推荐

专栏 · 恋猫de小郭 · 2小时前 · 前端 / Flutter
移动端开发真的已经式微了吗？

👍 25 🗨 32



专栏 · isboyjc · 18小时前 · JavaScript / 前端
「硬核JS」一次搞懂JS运行机制

👍 16 🗨 1



专栏 · 喝粥api和 · 1天前 · Vue.js / 前端
Vue优雅设计一个组件

👍 3 🗨 4

专栏 · JessicaC · 14小时前 · 掘金翻译计划 / 前端
[译] 5 个简单步骤为您的网站创建 Sitemap

👍 2 🗨 1

专栏 · zhangbao90s · 1小时前 · JavaScript / 前端
JS：生成器函数和迭代器之前的关系

👍 2 🗨 1

专栏 · 北冥南葵 · 13小时前 · JavaScript
使用Unicode为私密文本添加数字水印

👍 2 🗨

专栏 · 五月君 · 35分钟前 · JavaScript
JavaScript 浮点数之谜：0.1 + 0.2 为什么不等于 0.3？

👍 1 🗨



存 · 专栏 · 杨小顺 · 17小时前 · 代码规范 / 前端
你所需要知道的代码整洁之道

👍 2 🗨



专栏 · 前端瓶子君 · 2天前 · 前端
是时候抛弃Postman了，试试 VS Code 自带神器插件🔥🔥🔥

👍 426 🗨 55

专栏 · 给你一世倾城 · 11小时前 · WebSocket / 前端
IM系统的前世今生——2小时快速搭建高性能、可拓展的IM系统

👍 53 🗨 5

专栏 · aoping · 2小时前 · JavaScript
[译]JavaScript可视化：作用域链

👍 4 🗨

