forEach到底可以改变原数组吗?

> 链接：https://juejin.cn/post/6844903912093253645#comment

> 这几天在平时练习的时候，发现一个很匪夷所思的问题！就是我的印象中，`forEach`是可以改变原数组的呀！！！????，but！为什么现在这么简单的字符串数字组成的数组，咋就永远改不了原数组那？？？咋就这么费劲的吗？？？
>
> 后来，在各种尝试无果后，硬是逼着我，无奈的打开了项目，我就想看看平时我都用的这么理所当然可以改变的`forEach`到底咋就可以改变了！
>
> 后来，仔细一看，我一般在项目中数组操作的都是**对象数组**，而不是**数字字符串数组**，接着上网一查才知道根本原因：原来是因为是**引用类型与基本数据类型的区别呀！**
>
> 参考链接：blog.csdn.net/Janus_lian/...

1、基本数据类型 -> 死都改不动原数组！

```
const array = [1, 2, 3, 4];
array.forEach(ele => {
ele = ele * 3
})
console.log(array); // [1,2,3,4]
```

2、引用类型 -> 类似对象数组可以爽快改变偶~

```
const objArr = [{
    name: 'wxw',
    age: 22
}, {
    name: 'wxw2',
    age: 33
}]
objArr.forEach(ele => {
    if (ele.name === 'wxw2') {
        ele.age = 88
    }
})
console.log(objArr); // [{name: "wxw", age: 22},{name: "wxw2", age: 88}]
```

3、那引用类型 -> 改变整个单次循环的item那？ -> NO！不行

```
const changeItemArr = [{
    name: 'wxw',
    age: 22
}, {
    name: 'wxw2',
    age: 33
```

```
}]
changeItemArr.forEach(ele => {
    if (ele.name === 'wxw2') {
        ele = {
            name: 'change',
            age: 77
        }
    }
})
console.log(changeItemArr); // [{name: "wxw", age: 22},{name: "wxw2", age: 33}]
```

4、终极无敌屡试不爽的方法！

```
// 基本类型可以欧~
const numArr = [33,4,55];
numArr.forEach((ele, index, arr) => {
    if (ele === 33) {
        arr[index] = 999
    }
})
console.log(numArr);  // [999, 4, 55]

// 引用类型也可以欧~
const allChangeArr = [{
    name: 'wxw',
    age: 22
}, {
    name: 'wxw2',
    age: 33
}]
allChangeArr.forEach((ele, index, arr) => {
    if (ele.name === 'wxw2') {
        arr[index] = {
            name: 'change',
            age: 77
        }
    }
})
console.log(allChangeArr); // // [{name: "wxw", age: 22},{name: "change", age: 77}]
```

总结一下

- 基本类型我们当次循环拿到的**ele**，只是forEach给我们在另一个地方复制创建新元素，是和原数组这个元素没有半毛钱联系的！所以，我们使命给循环拿到的**ele**赋值都是无用功！
- 专业的概念说就是：JavaScript是有**基本数据类型**与**引用数据类型**之分的。对于基本数据类型：`number`，`string`，`Boolean`，`null`，`undefined` 它们在**栈内存中直接存储变量与值**。而Object对象的真正的数据是保存在**堆内存**，**栈内只保存了对象的变量以及对应的堆的地址**，所以操作Object其实就是直接操作了原数组对象本身。
- forEach 的基本原理也是for循环，使用arr[index]的形式赋值改变，无论什么就都可以改变了。