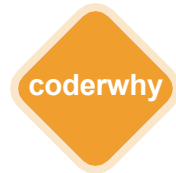


content



- 1 认识包装类型
- 2 数字类型Number
- 3 数学对象Math
- 4 字符串类型String
- 5 数组Array使用详解
- 6 Date类型的使用



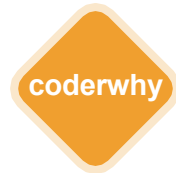
原始类型的包装类

- JavaScript的原始类型**并非对象类型**，所以从理论上来说，它们是**没有办法获取属性或者调用方法的**。
- 但是，在开发中会看到，我们会经常这样操作：

```
var message = "Hello World"
var words = message.split(" ")
var length = message.length

var num = 2.54432
num = num.toFixed(2)
```

- 那么，为什么会出现这样奇怪的现象呢？（悖论）
 - 原始类型是**简单的值**，默认**并不能调用属性和方法**；
 - 这是因为JavaScript为了可以**使其可以获取属性和调用方法**，**对其封装了对应的包装类型**；
- 常见的包装类型有：**String、Number、Boolean、Symbol、BigInt类型**



包装类型的使用过程

- 默认情况，当我们调用一个原始类型的属性或者方法时，会进行如下操作：
 - 根据原始值，创建一个原始类型对应的包装类型对象；
 - 调用对应的属性或者方法，返回一个新的值；
 - 创建的包装类对象被销毁；
 - 通常JavaScript引擎会进行很多的优化，它可以跳过创建包装类的过程在 直接完成属性的获取或者方法的调用。

- 我们也可以自己来创建一个包装类的对象：

- name1是字面量 (literal) 的创建方式，name2是new创建对象的方式；

```
var name1 = "why"
var name2 = new String("why")
console.log(typeof name1) // string
console.log(typeof name2) // object
console.log(name1 === name2) // false
```

- 注意事项：null、undefined没有任何的方法，也没有对应的“对象包装类”；



Number类的补充

- 前面我们已经学习了Number类型，它有一个对应的数字**包装类型Number**，我们来对它的方法做一些补充。
- **Number属性补充：**
 - Number.MAX_SAFE_INTEGER: JavaScript 中最大的安全整数 ($2^{53} - 1$);
 - Number.MIN_SAFE_INTEGER: JavaScript 中最小的安全整数 $-(2^{53} - 1)$
- **Number实例方法补充：**
 - **方法一：toString(base)**，将数字转成字符串，并且按照base进制进行转化
 - ✓ base 的范围可以从 2 到 36，默认情况下是 10;
 - ✓ 注意：如果是直接对一个数字操作，需要使用`..`运算符;
 - **方法二：toFixed(digits)**，格式化一个数字，保留digits位的小数;
 - ✓ digits的范围是0到20（包含）之间;
- **Number类方法补充：**
 - **方法一：Number.parseInt(string[, radix])**，将字符串解析成整数，也有对应的全局方法parseInt;
 - **方法二：Number.parseFloat(string)**，将字符串解析成浮点数，也有对应的全局方法parseFloat;
- **更多Number的知识，可以查看MDN文档：**
 - https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/Number



Math对象

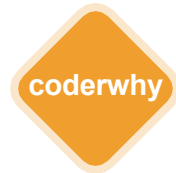
- 在除了Number类可以对数字进行处理之外，JavaScript还提供了一个**Math对象**。
 - Math是一个**内置对象**（不是一个构造函数），它拥有一些**数学常数属性和数学函数方法**；
- **Math常见的属性：**
 - Math.PI：圆周率，约等于 3.14159；

- **Math常见的方法：**

- **Math.floor**：向下舍入取整
- **Math.ceil**：向上舍入取整
- **Math.round**：四舍五入取整
- **Math.random**：生成0~1的随机数（包含0，不包含1）
- **Math.pow(x, y)**：返回x的y次幂

数字	Math.floor	Math.ceil	Math.round
3.1	3	4	3
3.6	3	4	4

- **Math中还有很多其他数学相关的方法，可以查看MDN文档：**
 - https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/Math



String类的补充（一） - 基本使用

- 在开发中，我们经常需要对字符串进行各种各样的操作，String类提供给了我们对应的属性和方法。
- String常见的属性：
 - `length`：获取字符串的长度；
- String也有很多常见的方法和操作，我们来进行学习。

- 操作一：访问字符串的字符
 - 使用方法一：通过字符串的索引 `str[0]`
 - 使用方法二：通过`str.charAt(pos)`方法
 - 它们的区别是索引的方式没有找到会返回`undefined`，而`charAt`没有找到会返回空字符串；
- 练习：字符串的遍历
 - 方式一：普通for循环
 - 方式二：for..of遍历

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/678121025031006120>