

## 第四章 数值扩展

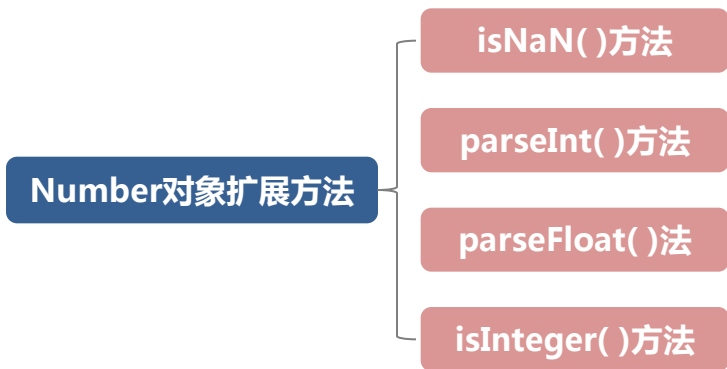
一样的在线教育，不一样的教学品质



# 目录 Contents

- ◆ Number对象扩展方法
- ◆ Math对象的扩展

# Number对象的扩展



# 1.isNaN( )方法

isNaN( )方法：检查一个值是否为NaN；

```
console.log(Number.isNaN(12));
```

```
console.log(Number.isNaN(NaN));
```

## ■ 2.parseInt( )方法和parseFloat( )方法

**parseInt( )方法**：解析一个字符串，并返回一个**整数**；

**parseFloat( )方法**：解析一个字符串，并返回一个**浮点数**；

**ES6写法**：

```
console.log(Number.parseInt('12.89'))  
  
console.log(Number.parseFloat('12.23'))
```

**目的**：逐步减少全局的方法，使语言逐步模块化。

## 3.isInteger( )方法

**isInteger( )方法**：用来判断一个数值是否为**整数**。

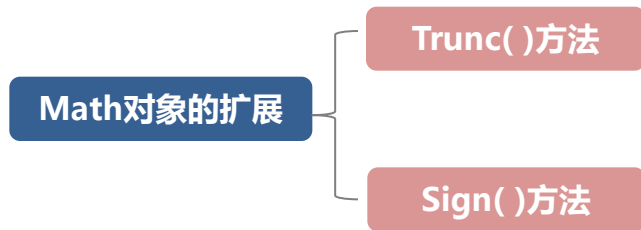
```
console.log(Number.isInteger(68));  
  
console.log(Number.isInteger(68.0));  
  
console.log(Number.isInteger(68.2));
```

**说明**：ES6中的isInteger方法认为，68.0和68是同一个值。



# 目录 Contents

- ◆ Number对象扩展方法
- ◆ Math对象的扩展





# ■ 1.trunc( )方法

**作用：**去除一个数的小数部分，返回整数部分。

```
console.log(Math.trunc(5.1));  
  
console.log(Math.trunc(5.9));  
  
console.log(Math.trunc(-5.1));  
  
console.log(Math.trunc(-5.9));
```

**注意：**对于非数值，trunc( )方法内部会使用Number对象中的方法将其先转换为数值。

对于空值和无法截取整数的值，返回的都是NaN。

## ■ 2.sign( )方法

作用：判断一个数到底是正数，负数还是0。

该方法返回值有5种情况：

- ◆ 参数为正数，返回+1
- ◆ 参数为负数，返回-1
- ◆ 参数为0，返回0
- ◆ 参数为-0，返回-0
- ◆ 其他值，返回NaN



一样的在线教育，不一样的教学品质