

# util实用工具 & os模块

## 1. util实用工具

- util 是一个Node.js 核心模块，提供常用函数的集合，用于弥补核心 JavaScript 的功能 过于精简的不足。
- 使用方法如下：

```
const util = require('util');
```

- [util实用工具在线文档](#)

### ① util.format(format[, ...args])

- format 一个类似 printf 的格式字符串。

```
const util = require("util");

console.log(util.format("我的名字叫%s，今年%d岁。", "张三", 20));
// 我的名字叫张三，今年20岁。
```

- util.format() 是一种用作调试工具的同步方法。某些输入值可能会产生严重的性能开销，从而阻止事件循环。请谨慎使用此功能，切勿在热代码路径中使用。

### ② util.promisify(original):

- 传入一个遵循常见的错误优先的回调风格的函数（即以 (err, value) => ... 回调作为最后一个参数），并返回一个返回 promise 的版本。
- 案例：

```
//1、没有使用promisify获取当前文件大小
const fs = require('fs');

let size = 0; //初始一个文件大小变量

fs.stat("./ddemo.js", (err, stats) => {
    size = stats.size;
    //console.log('当前文件大小:', size); //此处可以获取到
});
console.log('当前文件大小:', size); //此处获取不到

//2. 使用promisify获取当前文件大小
const util = require('util');
const fs = require('fs');
```

```
const stat = util.promisify(fs.stat);
stat('./nd01.js').then((stats) => {
  console.log('当前文件大小:', stats.size);
}).catch((error) => {
  //处理错误。
});
```

### ③ util.types类型检测

- util.types提供各种内置对象的类型检查 (推荐写法)
- 以前是没有types, 直接使用目前官方已经开始弃用。

```
const util = require('util');

util.isRegExp(/some regexp/)
// true
util.isRegExp(new RegExp('another regexp'))
// true
util.isRegExp({})
// false
```

### util.isRegExp(object)

- 如果给定的参数 "object" 是一个正则表达式返回true, 否则返回false。

```
const util = require('util');

//1. 正则类型验证
util.isRegExp(/some regexp/)
// true
util.isRegExp(new RegExp('another regexp'))
// true
util.isRegExp({})
// false

//2. 日期类型验证
console.log(util.types.isDate(new Date()));
// true
console.log(util.types.isDate(Date()));
// false (without 'new' returns a String)
console.log(util.types.isDate({}));
// false
```

## 2. OS模块

- Node.js中的os模块提供了一些基本的系统操作函数。我们可以通过以下方式引入该模块，并使用：

```
const os = require("os");

console.log('CPU的字节序：' + os.endianness());

console.log('操作系统名：' + os.type());

console.log('操作系统的发行版本：' + os.release());

console.log('编译时的操作系统名：' + os.platform());

console.log('系统内存总量：' + os.totalmem() + " bytes.");

console.log('操作系统空闲内存量：' + os.freemem() + " bytes.");
```

- 输出结果：

```
CPU的字节序：LE
操作系统名：Darwin
操作系统的发行版本：18.7.0
编译时的操作系统名：darwin
系统内存总量：17179869184 bytes.
操作系统空闲内存量：64544768 bytes.
```