

# 模块作用域

一个文件就是一个模块

它都有自己的作用域

使用var定义的一个变量，它并不是全局的而是属于当前模块下的

```
var a = 100;
```

定义全局变量——`global.a = 200;`

```
console.log(a)——100
```

```
console.log(global.a)——200
```

# 模块加载系统

```
require("路径")
```

# 模块加载机制

## 1、绝对路径

```
__filename //获取当前文件路径
```

```
require('D:/nodejs/module/2.js')
```

## 2、相对路径

```
require('./2.js'); //一定要加./
```

`require('2.js');`如果不加它将加载Node的核心模块，或者`node_modules`目录下的模块

## 3、文件名

```
require('./2');
```

- 1、首先按文件名查找，
- 2、如果没有找到会按文件名.js 查找
- 3、如果还没有找到会按文件名.json查找
- 4、如果还没有找到会按文件名.node查找

文件名——>.js——>.json——>.node——>抛出错误

# 外部访问模块中定义的变量

在一个模块中var一个变量是属于它内部的变量，不能够外部访问

想要访问要通过以下方法：

## 1、把变量作为全局对象global的一个属性——不推荐

## 2、使用模块对象module

在`module`对象下有一个子对象`exports`,

我们可以通过它去访问其它模块下的局部变量

```
require('./4.js') //这个方法的返回值，其实就是被加载模块中的module.exports  
module.exports.a = a;  
var m5 = require('./4.js')
```

### 3、直接使用模块下的`exports`——**推荐**

```
exports.a = a;  
var m5 = require('./4.js')
```

**不要直接给`exports`和`module.exports`赋值，这样做会切断引用关系**