复习

全局对象

nodejs: global

js: window

process.arch/platform/env/version/pid/kill()

console.log/info/warn/error/time/timeEnd

Buffer.alloc(5,'abcde')

全局函数

setTimeout(回调函数,间隔时间)/clearTimeout

setInterval/clearInterval

模块

核心模块，自定义模块，第三方模块

(function(){

代码

//导出

module.exports.a=1;

})

引入

require()

1.模块系统

|  |
| --- |
| require: 是一个函数，用于引入模块，返回一个对象  module: 指代当前的模块  module.exports: 当前模块的导出对象  exports: 指向module.exports  \_\_filename: 当前模块的绝对路径和文件名称  \_\_dirname: 当前模块的绝对路径 |

2.模块分类

|  |  |  |
| --- | --- | --- |
|  | 以路径开头 | 不以路径开头 |
| 文件模块 | require('./circle.js')  常用于引入用户自定义的模块 | require('querystring')  常用于引入官方提供的核心模块 |
| 目录模块 | require('./02\_2')  到02\_2目录下寻找package.json文件中main属性对应文件，如果找不到引入index.js | require('04\_2')  到当前目录下的node\_modules中寻找目录04\_2，如果找不到会到上一级目录寻找，直到顶级目录；常用于引入第三方模块 |

练习:创建模块03\_1.js，引入当前目录下的03\_2目录，在03\_2目录含有文件fun.js，导出一个函数add(计算两个数字相加)；在03\_1.js中调用该函数。

练习:在05目录下创建模块05\_1.js，引入不带路径的05\_2目录，含有文件foo.js，打印'hello'；

3.包和npm

CommonJS是一种规范，JS向服务器端发展就是遵循了这种规范，才出现模块概念。

包(package): 就是一个目录模块

npm: 用于管理包的工具

**切换目录**

cd 完整路径 回车

在指定目录下空白区域，按住shift，单击鼠标右键，选择'在此处打开powershell窗口'

**使用npm安装**

npm install 包名称

自动下载到当前目录下的node\_modules中

npm init -y 初始化一个package.json文件，项目说明文件，会记录安装的所有的包

npm install 自动安装package.json中记录包

**npm官网**

www.npmjs.com

3.查询字符串模块

查询字符串:浏览器向web服务器发请求，传递数据的一种方式。位于URL中?后边的部分

https:// jd.com/Search?keyword=电脑**&**enc=utf-8

parse() 将查询字符串解析为对象

stringify() 将对象格式化为查询字符串

练习: 获取查询字符串中商品的价格和名称

http://www.codeboy.com/shopping?price=4999&pname=dell

4.url模块

用于对url处理和解析

|  |
| --- |
| http://www.codeboy.com:80/detail.html?lid=5&pname=dell |

parse() 将url解析为对象

format() 将对象格式化为url

练习: 获取以下url中查询字符串的数据

https://www.tmooc.cn:443/course.html?cid=18&cname=javascript

①将url解析为对象，获取到查询字符串

②把获取的查询字符串解析为对象

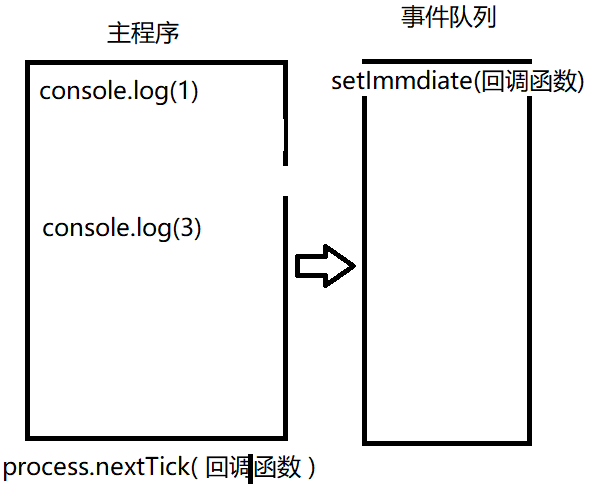
5.立即执行定时器

|  |
| --- |
| 开启  var timer=setImmediate( 回调函数 )  清除  clearImmediate(timer) |

在事件队列的开头执行

|  |
| --- |
| process.nextTick( 回调函数 ) |

在主程序的末尾执行



6.fs模板

文件系统模块

用于文件的操作；文件包含目录形式和文件形式

(1)查看文件的状态

fs.stat( path, callback ) / fs.statSync( path )

path 文件的路径

callback 回调函数

err 可能产生的错误

stats **文件的状态**

isFile() 是否为文件

isDirectory() 是否为目录

7.同步和异步

同步: 会阻止代码往后执行，只有前边的代码执行完才能执行后边的，通过**返回值**获取结果

异步: 不会阻止代码往后执行，后边的代码可以先执行，不需要等待前边代码，通过**回调函数**获取结果。

课后任务

(1)复习今天内容，整理思维导图

(2)练习

创建main.js模块，引入目录模块home,在home中含有文件fun.js，导出一个函数，传递文件的路径，返回文件的状态，在main.js中调用该函数。

get('02\_2')

(3)预习http协议(请求，响应)，http模块