复习

ES6

块级作用域 let { }

不允许多次使用let声明

参数增强 function fn(a,b,c=0){ }

箭头函数(回调函数) sort( (a,b)=>{ return a-b; })

模板字符串 ` ${js表达式} `

nodejs

全局对象 global/process/console/Buffer

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

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

Buffer.alloc(6,'abcdef') toString/String

模块

功能体 文件(模块)封装

(function(exports,require,module,\_\_filename,\_\_dirname){

require: 引入模块

module: 当前的模块

module.exports.a=1 :要导出的内容

})

1.模块系统

|  |
| --- |
| require: 引入模块，返回一个对象  module: 指代当前的模块对象  module.exports: 当前模块的导出对象  exports: 指代module.exports  \_\_filename: 当前模块的绝对目录和模块名称  \_\_dirname: 当前模块的绝对目录 |

2. 模块引入

|  |  |  |
| --- | --- | --- |
|  | 以路径开头(用于自定义模块) | 不以路径开头 |
| 文件模块 | require('./circle.js')  常用于引入自定义的模块，后缀名.js可以省略 | require('url')  常用于引入官方提供的核心模块 |
| 目录模块 | require('./02\_2')  到02\_2下寻找package.json文件中main属性对应的文件名称，如果找不到引入index.js | require('04\_2')  会到当前目录下的node\_modules中寻找04\_2目录，如果找不到，往上一级查找指定顶级目录；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目录模块，在03\_2目录有一个文件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：创建目录05，在05下创建05\_1.js引入不以路径开头目录模块05\_2，含有文件foo.js，随便打印一句话；

3.包和npm

CommonJS是一种规范，JS向服务器端发展就遵循了这个规范。

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

在指定的目录空白区域，按住shift键，单击鼠标邮件，选择'在此处打开powershell窗口'

**使用npm安装(下载)**

npm install 包名称 回车

自动安装到所在目录下的node\_modules中

npm init -y 初始化一个package.json文件,用于记录安装的包相关信息

npm install 回车

自动安装package.json中记录的包名称以及版本号

**npm官网**

https://www.npmjs.com

4.querystring模块

查询字符串：浏览器向服务器发请求，传递数据的一种方式

jd.com/Search?keyword=手机**&**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编号和商品名称

lid=5&pname=apple

5.url模块

http://www.tmooc.cn:80/detail.html?lid=5

parse() 将URL解析为对象

protocol 协议

hostname 主机名 域名/ip地址

port 端口号

pathname 请求的文件路径名称

query 查询字符串

format() 将对象格式化为URL

注意：query中对应的是对象

6.全局函数

parseInt/parseFloat/isNaN/isFinite/eval...

(1)一次性定时器

|  |
| --- |
| 开启  var timer=setTimeout( 回调函数, 间隔时间 );  当间隔时间到了，执行一次回调函数; 单位是毫秒  清除  clearTimeout(timer) |

(2)周期性定时器

|  |
| --- |
| 开启  var timer=setInterval( 回调函数, 间隔时间 )  每隔一段时间，执行一次回调函数  清除  clearInterval(timer) |

(3)立即执行

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

在事件队列的开头执行

(4)立即执行

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

在主程序的后边执行

7.fs模块(文件操作)

文件包含文件和目录，统称为文件。

fs.stat( path, callback )

path 文件的路径

callback 回调函数，用于获取结果

err 可能产生的错误信息

stats 具体的文件状态

课后任务

(1)复习今天内容，整理思维导图

(2)获取URL中查询字符串的数据

https://www.tmooc.cn:80/course.html?cid=17&cname=web

(3)预习同步和异步，http协议 http模块