目录

[第一天](#_Toc23681_WPSOffice_Level1) [1](#_Toc23681_WPSOffice_Level1)

[第二天](#_Toc24306_WPSOffice_Level1) [2](#_Toc24306_WPSOffice_Level1)

[第三天](#_Toc10885_WPSOffice_Level1) [3](#_Toc10885_WPSOffice_Level1)

[第四天](#_Toc22627_WPSOffice_Level1) [4](#_Toc22627_WPSOffice_Level1)

[第五天](#_Toc19256_WPSOffice_Level1) [5](#_Toc19256_WPSOffice_Level1)

[第八天 6](#_Toc9552_WPSOffice_Level1)

[第九天 7](#_Toc6844_WPSOffice_Level1)

[第十天 8](#_Toc10154_WPSOffice_Level1)

[第十一天 - 项目实战 - 1 （录屏） 8](#_Toc16378_WPSOffice_Level1)

[第十二天 - 项目实战 - 2 （录屏） 9](#_Toc26028_WPSOffice_Level1)

[第十三天 - 项目实战 - 3 （录屏） 10](#_Toc22812_WPSOffice_Level1)

[第十四天 - 项目实战 - 4（录屏） 11](#_Toc20470_WPSOffice_Level1)

[第十五天 - 项目实战 - 5（录屏） 10](#_Toc8252_WPSOffice_Level1)

[第十六天 - 项目实战 - 6（录屏） 11](#_Toc12122_WPSOffice_Level1)

[第十七天 - 项目实战 - 7（录屏） 11](#_Toc26333_WPSOffice_Level1)

[第十八天 - 项目实战 - 8（录屏）](#_Toc20893_WPSOffice_Level1) [12](#_Toc20893_WPSOffice_Level1)

[第一周作业](#_Toc7411_WPSOffice_Level1) [12](#_Toc7411_WPSOffice_Level1)

[第二周作业](#_Toc24333_WPSOffice_Level1) [12](#_Toc24333_WPSOffice_Level1)

[第三周作业 13](#_Toc13020_WPSOffice_Level1)

第一天

1、检测系统中node版本号的指令是？

node -v

2、如何退出node执行环境（REPL环境）？

ctrl+c c

3、node如何创建服务器，写出代码？

var http=require(‘http’);

var server=http.createServer(function(req,res,next){

//业务逻辑 })

server.listen(3000,function(err){if(!err){console.log(‘服务器运行在3000端口上’)}})

1. http.createServer(function(req,res){ // 获取请求 })在如上函数中如何获取前端请求地址？

req.url

5、req,res分别代表什么？

req:全称request请求对象

res:全称 response响应对象

1. 如何向客户端浏览器响应‘hello world’？

res.write(‘hello world’);

res.end()

7.浏览器请求 ‘/favicon.ico’表示什么？如何做出响应？

表示浏览器（客户端）请求页面图标

if(req.url==’/favicon.ico’){

fs.readFile(‘图标地址’,function(err,data){

res.write(data);

res.end()

})

}

第二天

1. 如何加载fs模块？

var fs=require(“fs”);

1. fs模块如何异步读取一个文件(文件路径为：’./file.json’)？

fs.readFile(‘./file.json’,’utf-8’,function(err,data){

If(!err){

// data即为读取内容

}

})

1. fs如何同步读取文件（文件的路径为：‘./file.json’）？

fs.readFileSync(‘’./file.json”,”utf-8”);

1. 同步和异步的区别？各举一个Js中同步和异步的案例？

异步：在一件事情没有做完，可以接着做另外一件事 非阻塞

同步：一件事情没有做完，就不可以做下一件事 阻塞。

同步案例：for循环语句，alert() 等

异步案例：定时器，ajax异步请求

5、fs如何异步写入文件，写入内容为‘abc’,写入文件路径为’./file.txt’？

fs.writeFile(‘./file.txt’,’abc’,function(err){ })

1. 写出注册的form表单，提交地址为’/api/login’,提交方式为’get’,用户名字段为user,密码字段为pass？

<form action=”/api/login” method=’get’>

用户名：<input type=’text’ name=”user”/>

密码：<input type=’password’ user=”pass”>

<button type=’submit’ >注册</button>

</form>

7、querystring模块如何解析如下字符串,（‘user=tom&pass=123456’）？

querystring.parse(‘user=tom?pass=123456’)

8、url模块如何解析请求地址req.url, 得到如下参数{pathanem:’/’,query:{user:xxx,pass:xxxx}} ？

url.parse(req.url,true)

第三天

1、node服务器如何设置跨域请求和中文乱码？

res.writeHead(200, {

'Content-Type': 'application/json;charset=utf-8',

'Access-Control-Allow-Origin': '\*'

})

1. npm是什么？

npm是包管理工具

1. npm如何下载 jquery？

npm install jquery@版本号 或者 npm install jquery

4、npm工具下载的包放在哪个文件夹里？

node\_modules

5、npm如何创建package.json？

npm init

6、cnpm 和npm的区别？我们下载插件一般用哪个？

npm 下载资源服务器在国外，cnpm下载资源服务器为国内淘宝镜像 。 一般下载用cnpm

1. commenjs规范，如何导出一个模块？

module.exports={}

1. commenjs规范,如何导入一个模块？

var module=require(‘模块名或路径’)

1. 自己仿写url模块（能把’/api?user=tom&pass=12345’解析成对象{pathname:’/api’,query:{user:’tom’,pass:12345}}），遵循commenjs规范？

module.exports={

parse:function(str,foo){

var arr=str.split('?');

var pathname=arr[0]

if(foo){

var arr1=arr[1].split('&');

var query={};

for(var i=0;i<arr1.length;i++){

var arr2=arr1[i].split('=');

query[arr2[0]]=arr2[1];

}

}else{

var query=arr[1];

}

return {pathanme:pathname,query:query}

}

}

1. npm包的发布流程？
2. npm login 登录

username：用户名

password:密码

email:邮箱

2.npm publish 发布

第四天

1. node搭建服务器，完成移动端页面（录屏）？



1. node服务器如何接受post请求？

var str=””

req.on(‘data’,function(thunk){

str+=thunk;

})

req.on(‘end’,function(){

})

3.get与post区别？

|  |  |  |
| --- | --- | --- |
| 历史 | 参数保留在浏览器历史中。 | 参数不会保存在浏览器历史中。 |
| 对数据长度的限制 | 是的。当发送数据时，GET 方法向 URL 添加数据；URL 的长度是受限制的（URL 的最大长度是 2048 个字符）。 | 无限制。 |
| 安全性 | 与 POST 相比，GET 的安全性较差，因为所发送的数据是 URL 的一部分。  在发送密码或其他敏感信息时绝不要使用 GET ！ | POST 比 GET 更安全，因为参数不会被保存在浏览器历史或 web 服务器日志中。 |
| 可见性 | 数据在 URL 中对所有人都是可见的。 | 数据不会显示在 URL 中。 |

第五天

1. express搭建服务器完成案例（注意：视图文件的配置，视图文件引擎的配置，静态资源文件的配置）（录屏）？

一．首先创建app.js文件如下：

//加载express模块

var express=require("express");

// 创建app服务器 等同于 http.createServer()

var app=express();

//加载路由模块

var IndexRouter=require('./routes/index.js');

var RouterHome=require('./routes/home.js')

//导入path模块（处理路径的模块）

var path=require("path");

//加载ejs模板

var ejs=require("ejs");

//配置 视图文件位置

app.set("views",path.join(\_\_dirname,"views"));

//配置 视图文件引擎

app.set('view engine','ejs')

//设置静态资源文件位置

app.use('/public',express.static(path.join(\_\_dirname,'public')))

//指定ip或域名 （本地ip:127.0.0.1等同于本地域名localhost）

var ip='127.0.0.1';

//指定端口

var port=3000;

//app服务器监听端口

app.listen(port,ip,function(){ //app服务器监听端口

console.log("服务器运行在http://"+ip+":"+port)

});

//配置 一级路由规则

app.use('/',IndexRouter)

app.use('/a',RouterA)

//404处理

function fn(req,res){ // app添加 get请求根路径的路由

// res.send("hello world");// 向前台输出

console.log(req.url)

res.write("404");

res.end()

}

1. 安装依赖 express ,ejs
2. 创建views（视图文件），public（静态资源文件）,routes（路由文件）文件夹
3. routes下创建路由文件index.js,a.js

//加载express模块

var express = require('express');

//创建router对象

var router = express.Router();

//配置 二级路由

router.get('/', function(req, res, next) {

res.render('index')

});

//导出路由模块

module.exports = router;

1. ejs文件中如何声明变量a?

<% a=100 %>

1. ejs文件中如何使用输入变量a?

<%= a %>

4.ejs文件中如何引用 外部ejs文件（比如：./head.ejs）？

<% include ./head.ejs %>

1. 如果ejs文件中有数据 list=[{name:’apple’,price:5,address:’烟台’}，{name:”香蕉”,price:8,address:’海南’}，..... ],如何用表格将他们展示出来？

<table border="1">

<thead>

<tr>

<th>水果名</th>

<th>价格</th>

<th>产地</th>

</tr>

</thead>

<tbody>

<% for(var a in list){ %>

<tr>

<td>

<%= list[a].name %>

</td>

<td>

<%= list[a].price %>

</td>

<td>

<%= list[a].address %>

</td>

</tr>

<% } %>

</tbody>

</table>

第六天

要点：1.path.parse() 解析地址

1. path.normalize()格式化路径 注意'..' 和 '.'
2. path.join() 连接路径
3. path.resolve() 转换为绝对路径

1.\_ \_dirname表示什么？

当前文件所在目录

1. \_ \_filename表示什么？

当前文件路径

1. var str='c:\\wamp\\www\\a.html';

var obj=path.parse(str);

请写出obj.base,obj.ext,obj.dir,obj.name的值？

obj.base==‘a.html’

obj.ext==’.html’

obj.dir==’c:\\wamp\\www\\’

obj.name==’a’

1. console.log(path.normalize('/test/test1//abc/tab/..'));输出结果？

‘/test/test1/abc/’

1. console.log(path.join('/test', 'test1', 'home/api', 'tab', '..'));输出结果？

‘/test/test1/home/api/’

第七天

1. 全局安装express-generater 指令是？

cnpm install express-generator -g

或者

npm install express-generator -g

1. express创建项目test指令？

express -e test

1. express创建的项目中文件夹和文件有哪些？各自含义？

app.js 是主文件

package.json 是配置文件

bin启动文件

public静态资源文件

routes路由

views是视图文件

node\_modules 项目依赖

第八天

1、命令行启动mongodb数据库服务器在27017端口上的指令是（数据库位置c:/data/db）？

mongod --dbpath c:/data/db --port 27017

1. 命令行连接数据库数据库服务器的指令是（本地服务器，端口27017）？

mongo --port 27017

3、命令行查看数据库的指令？

show dbs

4、命令行查看当前数据库集合的指令？

show collections

5、命令行向user集合添加文档{user:tom，pass:123456}的指令？

db.user.insert({user:’tom’,pass:123456});

1. 命令行查询user集合中所有文档？

db.user.find()

7、命令行查询user集合中age大于18且小于20的文档？

db.user.find({age:{$gt:18,$lt:20}})

1. 命令行更新 user集合中name为‘李四’的文档，sex改为女，其他字段不变？

db.user.update({name:’李四’},{$set:{sex:’女’}})

1. 命令行删除user集合中 age大于20且性别sex不为女的所有文档

db.user.remove({$and:[{age:{$gt:20}},{sex:{$ne:’女’}}]})

或

db.user.remove({age:{$gt:20},sex:{$ne:’女’}})

第九天

1. 案例:

exprss搭建服务器，完成以下要求

1. 通过访问http://localhot:3000 加载首页
2. 页面中轮播图部分的图片数据，通过ajax请求http://localhost:3000/lunbo获取，node服务器完成此接口。
3. 页面中主要内容部分数据，通过ajax请求http://localhost:3000/main获取，node服务器完成此接口
4. 自己创建json文件，定义轮播图数据以及主要内容数据
5. 轮播效果（可用swiper）滑动效果（可用iscroll）



第十天

要点：使用bootstrap+express+mongoose+mongodb完成留言板

1、请简述mongoose是什么及作用？

Mongoose是封装了MongoDB操作的一个对象模型库

作用是连接后台服务器与数据库服务器，操作数据库（增删改查）

1. 安装mongoose指令?

cnpm install mongoose --save

1. mongoose连接数据库的代码？

mongoose.connet(‘mongodb://127.0.0.1:27017/dbname’,function(err){

If(!err){

console.log(‘数据库连接成功’)

}

})

5、请简述mongoose中schema、model、entity概念？

Schema ： 一种以文件形式存储的数据库模型骨架，不具备数据库的操作能力

Model ： 由 Schema 发布生成的集合模型，具有抽象属性和行为的数据库操作

Entity ： 由 Model 创建的实体，他的操作也会影响数据库。

1. 创建一个Schema，数据结构为user字符串类型，txt字符串类型，另添加时间戳？

var mongoose=require('mongoose');

var Schema=mongooe.Schema;

var myschema=new Schema({user:String,txt:String},{timestamps:true});

module.exports=mySchema;

1. 创建model模型，集合名为User（model中所用到的schema路径为 ../schema/user.js）？

var mongoose=require('mongoose');

var mySchema=require('../schema/user');

var User=mongoose.model('User',mySchema);

module.exports=User;

第十一天

1. 若通过mongoose创建的集合模型为User，User集合添加一条文档{user:’tom’,txt:’hello world’}，添加成功后重定向‘/’路径？,

User.create({user:’tom’,txt:’hello world’},function(err,data){

If(!err){

res.redirect(‘/’)

}

})

2、若通过mongoose创建的集合模型为User，请使用实体（entity.save()）的方式向User集合中添加一条文档{user:’tom’,txt:’hello world’}，添加成功后重定向‘/’路径？

var entity=new User({user:’tom’,txt:’hello world’});

entity.save(function(err,data){

If(!err){

res.redirect(‘/’)

})

})

1. 若通过mongoose创建的集合模型为User，User查询所有文档，查询成功后渲染index文件，（res.render(‘index’,{list:datas}) ）？

User.find({},function(err,datas){

If(!err){

res.render(‘index’,{list:datas})

}

})

4、若通过mongoose创建的集合模型为User，User集合删除{\_id:12345}的集合，删除成功后重定向‘/’路径？

User.remove({\_id:12345},function(err,data){

If(!err){

res.redirect(‘/’);

}

})

第十二天

1. 完成案例如下（录屏）：



1. 若通过mongoose创建的集合模型为Student，Student集合删除{\_id:12345}的集合，删除成功后向前端响应json数据为{code:1,msg:”删除成功”}？

Student.remove({\_id:12345},function(err,data){

If(!err){  
res.json({code:1,msg:’删除成功’})

}

})

1. 若通过mongoose创建的集合模型为Student，Student集合修改{\_id:12345}的集合，把sex修改为’女’，修改成功后向前端响应json数据为{code:1,msg:”修改成功”}？

Student.update({\_id:12345},{$set:{sex:’女’}},function(err,data){

If(!err){

res.json({code:1,msg:’修改成功’})\_

}

})

第十三天 - 项目实战 - 1 （录屏）

1、完成注册逻辑



第十四天 - 项目实战 - 2（录屏）

1. 

第十五天 - 项目实战 - 3（录屏）

1、注册登录添加cookie,设置七天免登录

第十六天 - 项目实战 - 4（录屏）

1.超级管理员页面开发

第十七天 - 项目实战 - 5（录屏）

1、普通用户页面开发

第十八天 - 项目实战 - 8（录屏）

1、汇总

第一周作业

1. Node提供的系统模块url，用那个方法可以将路径转化为对象方便使用？

url.parse()

1. url模块的使用

将请求地址解析成对象。

3、请使用http搭建一个最简单的服务器，直接写代码？

var http=require(‘http’);

var server=http.createServer(function(req,res,next){

//业务逻辑 })

server.listen(3000,function(err){if(!err){console.log(‘服务器运行在3000端口上’)}})

第二周作业

1、nodejs中的全局对象是谁？有全局对象window么？

global nodejs中没有window对象

3、如何给input绑定键盘事件？

Input.onkeydow(function(){})

6、未知宽高垂直水平居中？

给父元素设置绝对定位，子元素相对定位，子元素left：50%；top:50%; margen-left:-本身宽度一半；margen-top:-本身高度一半

7、请简述同步异步的区别？

异步：在一件事情没有做完，可以接着做另外一件事 非阻塞

同步：一件事情没有做完，就不可以做下一件事 阻塞。

同步案例：for循环语句，alert() 等

异步案例：定时器，ajax异步请求

8、请简述commonjs规范中模块导入和导出语法？

模块导出： module.exports={}

模块导入: require(‘路径’)

9、使用模块的好处？

封装模块，可以提高开发效率，提高代码复用率。

1. 解析该url 由哪些部分组成并说明

第三周作业

1、如果要全局安装express-generator使用的命令是？

2、如何使用express创建一个项目？

3、如何启动express项目？

4、为什么npm start可以启动express项目？

5、express项目中public目录的作用是？

6、Routes目录的作用是？

7、views文件夹的作用是？

8、app.js文件的作用是？

9、bin目录的作用是？