今天学习的内容

1:复习昨天知识重点

2:作业

3:今天的目标

3.1:第三方的模块(第二节课开网|重点)

安装5个第三方模块{e:/}

npm i express

npm i body-parser

npm i mysql

npm i cookie-parser

npm i express-session

npm i cors

3.2:nodejs 官方模块http

练习：静态web服务器:创建web服务器，接收客户端请求，

一律向客户端输出一句 <h3>hello nodejs</h3>

练习:动态web服务器:创建web服务器，接收客户端请求,

若客户端请求 /index.html则向客户输出 index.html内容

若客户端请求 /其它 则向客户输出

<h1>404 您查找的数据不存在</h1>

|  |
| --- |
| Error: listen EADDRINUSE :::8080  错误:原因 端口冲突(二个程序在工作时使用同一个端口)  解决:停一个程序或者修改程序端口 |

3.3:nodejs 第三方模块 mysql(此模块专用于访问与操作mysql数据库服务器)

为了精简node.js解释器、官方没有提供任何数据库相关模块,必须使用npm工具下载第三方模块，在www.npmjs.org 上搜索关键字 mysql ,可以得到很多相关模块。

使用工具下载安装

npm i mysql

mysql数据库:查询速度比较快更新慢下

数据库存储三种方案

(1)大量数据保存 mysql数据库中

(2)NOSQL解析方案:Redis(内存级别数据库)

(3)文件存储:试卷

3.4:nodejs 第三方模块 mysql-使用步骤

(1)创建到数据库的连接(保证mysql启动成功/第三方模块加载成功)

const mysql = require("mysql");

var conn = mysql.createConnection({...})

(2)创建一条SQL语句并且发送SQL

var sql = "...";

conn.query(sql,function(err,result){});

(3)关闭连接

conn.end();

3.5:nodejs 第三方模块 mysql-网络攻击手段

(1)SQL注入

利用:sql语句中弱点，添危害数据放在sql语句中.

var fname = "';drop table mysql.user ";

var fname = "';select into c:/aa.dat ";

var uname = "' OR 1=1 #";

var upwd = "";

解决方案一:占位符

var sql = "INSERT INTO user VALUES(null,?,?)";["tom","123"]

解决方案二:项目工程分为二个组成部分:

前端业务<----->核心功能(mysql);

(2)XSS 脚本攻击

var name = "<script>document.all.disabled=''</script>";

用户名 [jerry]

解决方案:

所有用户输入的数据过滤:drop table;

练习:仿写上述代码，使用node.js向 web1708acl02/t\_user

[uid/uname/upwd] 表添加一行新的记录.

练习:仿写上述代码，使用node.js向 web1708acl02/t\_user

uid=3 upwd=123456

练习:使用http和mysql模块

\*创建动态web服务器,接收如下请求URL

\*/register.html 服务器向客户端返回register.html内容

/register.do 接收客户端提交uname/upwd

GET req.query.uname

保存mysql t\_user

/login.html 服务器向客户端返回 login.html 内容

/login.do 接收客户端提交 unname/upwd

/userlist.html

/userlist.do 接收客户请求，发送用户列表

404.html

作业1:/userlist.do

作业2:授权