今天学习的内容

1：复习昨天重点知识

nodejs模块分类

1)官方提供模块:

querystring,url,buffer,fs,http

2)第三方模块--npm

mysql

3)自定义模块

文件模块 user.js require("./user");

目录模块 node\_modules/db/package.json require("db");

http模块

const http = require("http");

var server = http.createServer();

server.listen(80);

server.on("request",(req,res)=>{})

mysql模块的使用

npm i mysql ./node\_modules/mysql

#加载模块

const mysql = require("mysql");

#创建连接 发送sql

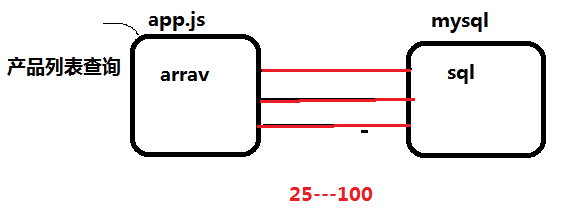
var conn = mysql.createConnection({});

conn.query("INSERT INTO t\_user values(?,?)",['tom',1],function(err,result){});

#关闭连接

conn.end();

2:为了提升数据库操作性能，nodejs提供完善功能连接池.



1:创建连接池

var pool = mysql.createPool({...})

2:从连接池中获取连接

pool.getConnection((err,conn)=>{

3: 释放连接，回到连接池

conn.release();

});

10:25---10:35

练习:动态 web 服务器, 接收客户端请求，若客户请求地址是/register,则向客户端输出 public/register.html中的内容.

<form action="register.do">

用户名<input type="text" name="uname" />

密码<input type="password" name="upwd" />

提交<input type="submit"/>

</form>

若客户端请求URL 是register.do,则解析URL中查询字符串，把客户端提交uname和upwd追加写入jd\_user表中，并向客户端输入"注册成功".

3:Express框架

使用官方提供http模块可以创建一个web服务器应用,但是此模块非常底层，要处理各种情形,比较繁琐。推荐使http 模块进一步封装简化模块--express.

第三方模块，是一个基于nodejs的http模块而编写高层模块,简化web服务端应的开发,express是一个请求处理工具，用于接客户端请求消息，返回响应消息.

可以到 npmjs.org上下载该模块

该模块还有自己专用官网: http://expressjs.com

中文镜像网站: http://www.expressjs.com.cn

使用方法

1)下载 express模块

npm i express

2)在自己的模块中引入express模块

const express = require("express");

3)使用 http 模块创建一个 web服务器对象，让

express()担当请求处理函数.

var app = express();

var server = http.createServer(app);

server.listen(8080);