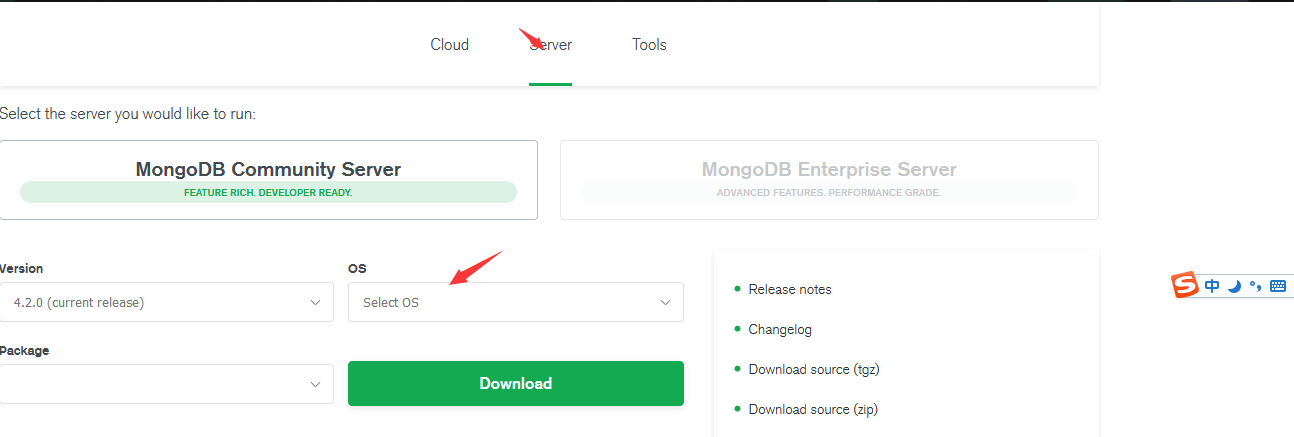
# 讲义07

## 安装mongodb

### 1.下载地址

https://www.mongodb.com/download-center/community



测试：

1. 测试是否成功

打开cmd，输入mongo查询版本，是否安装成功

1. 链接数据库

开启mongodb数据库服务 如： mongod --dbpath 数据库的安装目录

新开cmd，使用数据库 如： mongo 127.0.0.1:27017

1. 创建数据库

use DATABASE\_NAME

1. 查询数据看

show dbs

## 安装express

### 创建项目目录

如 c: myprotect

### 构建package.json文件

npm init --yes

### 安装express框架

npm install express --save

Npm install supervisor --save -g 用来运行文件的模块（选按）

### express快速入门

1. 简单使用

//1.引入

var express = require('express');

var app = new express();

//2.配置路由

app.get('/', function(req, res) {

res.send('Hello World!');

});

//3.监听端口

app.listen(3000, '127.0.0.1');

## 安装ejs

### 安装ejs模板引擎

npm install ejs --save 新版本

Npm install [ejs@2.7.1](mailto:ejs@2.7.1) --save 老版本

### express中使用ejs

var express = require("express");

var app = express();

app.set("view engine", "ejs");

app.get("/", function(req, res) {

res.render("news", {

"news": ["我是小新闻啊", "我也是啊", "哈哈哈哈"]

});

});

App.listen(3000)

### ejs模板引擎简单操作

指定模板位置 ，默认模板位置在 views

|  |
| --- |
| app.set('views', \_\_dirname + '/info); |

Ejs 引入模板

|  |
| --- |
| <%- include header.ejs %> 2.7.1  <%- include(‘header’) %> 3.0.1 |

Ejs 绑定数据

|  |
| --- |
| <%=h%> |

Ejs 模板判断语句

|  |
| --- |
| <% if(true){ %>  <div>true</div>  <%} else{ %>  <div>false</div>  <%} %> |

Ejs 模板中循环数据

|  |
| --- |
| <%for(var i=0;i<list.length;i++) { %>  <li><%=list[i] %></li>  <%}%> |

### 4.Ejs 后缀修改为 Html

这是一个小技巧，看着.ejs 的后缀总觉得不爽，使用如下方法，可以将模板文件的后缀换成我们习惯的.html。

* 在 app.js 的头上定义 ejs:,代码如下:

|  |
| --- |
| var ejs = require('ejs'); |

* 注册 html 模板引擎代码如下：

|  |
| --- |
| app.engine('html',ejs.\_\_express); |

* 将模板引擎换成 html 代码如下:

|  |
| --- |
| app.set('view engine', 'html'); |

* 修改模板文件的后缀为.html。

## Nodejs安装body-parser中间件

* GET 请求的参数在 URL 中，在原生 Node 中，需要使用 url 模块来识别参数字符串。在Express 中，不需要使用 url 模块了。可以直接使用 req.query 对象。
* POST 请求在 express 中不能直接获得，可以使用 body-parser 模块。使用后，将可以用req.body 得到参数。但是如果表单中含有文件上传，那么还是需要使用 formidable 模块。

### 1.安装

|  |
| --- |
| npm install body-parser --save |

### 2.使用 req.body 获取 post 过来的参数

|  |
| --- |
| var express = require('express')  var bodyParser = require('body-parser')  var app = express()  // parse application/x-www-form-urlencoded  app.use(bodyParser.urlencoded({ extended: false }))  // parse application/json  app.use(bodyParser.json())  app.use(function (req, res) {  res.setHeader('Content-Type', 'text/plain')  res.write('you posted:\n')  res.end(JSON.stringify(req.body, null, 2))  }) |

## 五、Nodejs 操作 MongoDb 数据库

### 1.项目目录中安装mongodb模块

npm install mongodb@2.2.33 --save-dev

### nodejs链接mongodb数据库

var express = require("express");

//数据库引用

var MongoClient = require('mongodb').MongoClient;

var app = express();

//数据库连接的地址，最后的斜杠表示数据库名字

var shujukuURL = 'mongodb://localhost:27017/news';

app.get("/",function(req,res){

//连接数据库，这是一个异步的操作

MongoClient.connect(shujukuURL, function(err, db) {

res.writeHead(200,{"Content-Type":"text/html;charset=UTF8"});

if(err){

res.send("数据库连接失败");

return;

}

res.write("恭喜，数据库已经成功连接 \n");

});

});

App.listen(8020);

### Nodejs 查找 MongoDb 数据库内容

var express = require("express");

//数据库引用

var MongoClient = require('mongodb').MongoClient;

var app = express();

//数据库连接的地址，最后的斜杠表示数据库名字

var shujukuURL = 'mongodb://localhost:27017/news';

app.get("/",function(req,res){

//连接数据库，这是一个异步的操作

MongoClient.connect(shujukuURL, function(err, db) {

res.writeHead(200,{"Content-Type":"text/html;charset=UTF8"});

if(err){

res.send("数据库连接失败");

return;

}

res.write("恭喜，数据库已经成功连接 \n");

Db.collection(‘user’).find({}).toArray(function(err,res){

Console.log(res)

Db.close()

})

});

});

## 六、商品管理系统

1.创建项目目录（之前已经完成）

1. 登陆页、后台商品管理、类别管理、购物车管理（购物车查询【修改】，用户管理忽略，直接设置管理员）
2. 文件上传（商品的增删改查）
3. **//配置public目录为我们的静态资源目录**

app.use(express.static('public'));

1. **//获取post传参操作**

[www.npmjs.com](http://www.npmjs.com) 官方网站

npm install body-parser --save

var bodyParser = require('body-parser')

// parse application/x-www-form-urlencoded

app.use(bodyParser.urlencoded({ extended: false }))

// parse application/json

app.use(bodyParser.json())

req.body 获取

1. **//图片上传操作**

Multiparty 模块

//设置应用中间件，/upload指向了upload目录

app.use('/upload',express.static('upload'));

npm install multiparty --save

var multiparty = require('multiparty');

//获取提交数据 fields post数据 files图片信息

var form = new multiparty.Form();

Form.uploadDir = “upload”

form.parse(req, function(err, fields, files) {

});

1. **处理商品id问题**

var ObjectID = require('mongodb').ObjectID;

1. **处理登陆状态问题**

npm install express-session --save

var session = require('express-session')

固定格式：

app.use(session({

  secret: 'keyboard cat',

  resave: false,

  saveUninitialized: true，

cookie: {

maxAge:1000\*60\*30

},

rolling:true

}))

Req.session.userinfo = “admin”

1. **微信api请求**

*//链接数据库，获取类别信息*mongoClient.connect(DBurl,**function**(err,db){

db.**collection**(**'type'**).find({}).toArray(**function**(errs,result){  
 res.writeHead(200,{**'Content-Type'**:**"application/json"**})  
 res.write(***JSON***.stringify(result))  
 res.end();  
 })  
})