

# NodeJS入门和企业级项目实战

第6天课堂笔记

班级：全栈前端训练营0219班

讲师：邵山欢

日期：2017年6月12日

邵山欢老师

微博：@邵山欢

QQ : 179427026

E-mail : ssh@iqianduan.cn

爱前端官网 ：[www.iqianduan.cn](http://www.iqianduan.cn)

# 目录

[NodeJS入门和企业级项目实战 1](#_Toc5548)

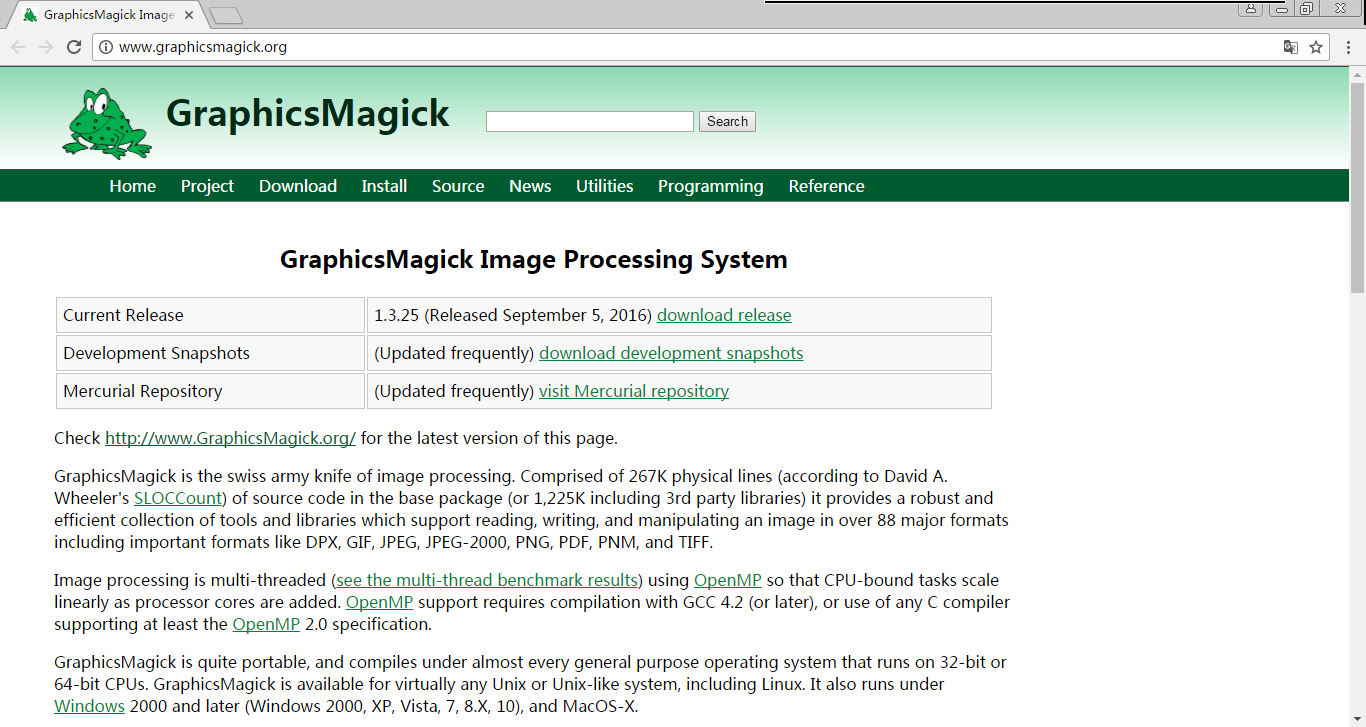
[目录 2](#_Toc10345)

[一、班级说说项目开发 - gm的使用 3](#_Toc15841)

[二、班级说说项目开发 - cut接口 6](#_Toc11550)

# 一、班级说说项目开发 - gm的使用

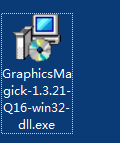
官网： http://www.graphicsmagick.org/

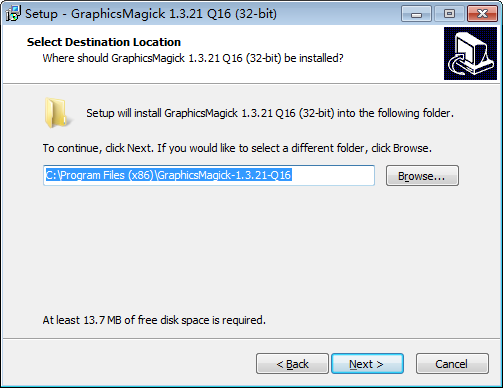




GraphicsMagick是德国一帮学生的发明，主要是图片算法。

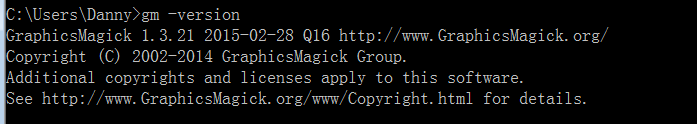
官网下载exe包，苹果用户自己找去吧。





我们的目的是要用命令行可以去处理图片。此时就要设置环境变量。**将安装路径设置为环境变量，这里不截图了**。

设置为环境变量之后，就能用gm -version验证是否安装成功：



接下来我们就能用gm命令（CMD中）对图片进行操作。

百度搜索“graphic magick 命令”即可。

gm原来叫做Imagemagick，API是相同的。

文档： http://www.cnblogs.com/ITtangtang/p/3951240.html

文档： http://blog.csdn.net/pzw\_0612/article/details/52299776

比如可以将图片改尺寸为200×200，注意是字母x，不是乘号：

gm convert 1.jpg -resize "200x200" 11.jpg

非等比缩图，按给定的参数缩图（缺点：长宽比会变化）

gm convert 1.jpg -resize "200x200!" 111.jpg

|  |  |  |
| --- | --- | --- |
| 1 | 11 | 111 |

gm convert 1.jpg -crop "144x91+112+204" 11111.jpg

gm在CMD中没用，必须要让nodejs操作gm。

安装依赖，就叫做“gm”

https://www.npmjs.com/package/gm

cnpm install --save gm

nodejs操作gm非常方便：

var fs = require('fs');

var gm = require('gm');

gm('./1.jpg')

.crop(200, 200, 150, 130)

.resize(50,50)

.noProfile()

.write('./11.jpg', function (err) {

if (!err) console.log('done');

});

# 二、班级说说项目开发 - cut接口

做一个接口/cut post请求，传入w、h、x、y、图片名字。

但是，由于我们的页面已经缩放了img，所以必须要求按比例乘以x、y、w、h才行：

$("#cutBtn").click(function(){

//得到原图和呈递的图片的比例

var rate = Number($(".bigimg").attr("data-default-w")) / picw;

//所有的参数乘以比例

//并没有提交图片路径，cut接口知道切谁么？知道！因为我们存session。

$.post("/cut",{

x : x \* rate,

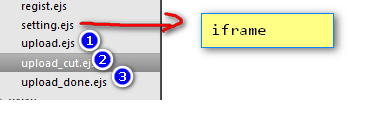
y : y \* rate,

w : w \* rate,

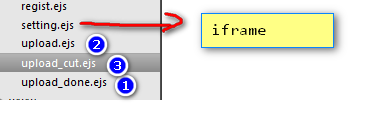
h : h \* rate

});

});



默认应该呈递当前头像：



两个问题：

1）用户的默认头像，应该持久在数据库中；

2）用户改了头像之后，应该将路径持久在数据库中。

为了保存用户头像地址，所以一定要开辟一个字段。但是要有默认字段，mongoose中设置默认字段的方法：

var userSchema = new mongoose.Schema({

"email" : String,

"pwd" : String,

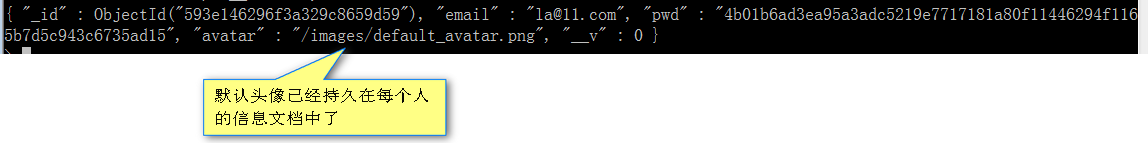
**"avatar" : {**

**type : String ,**

**default : "/images/default\_avatar.png"**

**}**

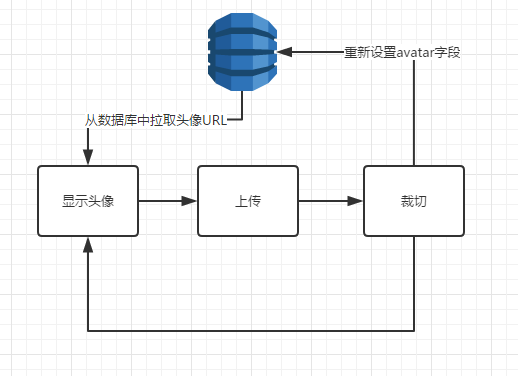
});



内嵌的入口改变一下：

<iframe src="/upload\_done" frameborder="0" width="100%" height="350px"></iframe>

cut的时候不仅仅要切图，更要持久化地址！



# 三、班级说说项目开发 - 发帖功能

## 3.1 前端界面

一切以最简化的方式进行。ajaxfileupload.js 是jq的异步文件上传插件。

页面上也要有input="file"的框，但是可以隐藏掉。

我们猫腻放一个input=file，然后display: none !important;

|  |  |
| --- | --- |
|  |  |

<li>

<b class="glyphicon glyphicon-plus"></b>

<input type="file" name="pic" class="dn"/>

</li>

模拟触发

<script type="text/javascript">

$(".picupbox b").click(function(){

$(this).siblings("input").click();

});

</script>

ajaxFileUpload插件的API：

http://blog.csdn.net/xujie3/article/details/51860139

<script src="/js/jquery.min.js"></script>

<script src="/js/bootstrap.min.js"></script>

<script src="/js/ajaxfileupload.js"></script>

<script type="text/javascript">

$(".picupbox b").click(function(){

$(this).siblings("input").click();

});

$(".picupbox input[type=file]").bind("change",function(){

//安装插件之后，就能合法使用ajaxFileUpload()函数

$.ajaxFileUpload({

type: "POST",

url: "/shuoshuopicupload",

data:{}, //要传到后台的参数，没有可以不写

secureuri : false, //是否启用安全提交，默认为false

fileElementId:'pic1', //文件选择框的id属性 （这个框框虽然不可见了，但是这里依然重要）

dataType: 'json', //服务器返回的格式

async : true, //是不是异步的

success: function(data){

alert(data.result);

},

error: function (data, status, e){

}

});

});

</script>

接口：

//图片上传

exports.shuoshuopicupload = (req,res) => {

//下面两句话来自formidable，来自：https://www.npmjs.com/package/formidable

var form = new formidable.IncomingForm();

//设置上传的文件地址

form.uploadDir = "./uploads";

//保留文件拓展名

form.keepExtensions = true;

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

res.send({"result" : 88888});

});

}

此时可以完成异步上传了。

问题：

* 多个上传图片，上传一张多一个li。
* 缩略图的显示
* 可以删除
* 可以排序
* 小菊花
* 图片数量提示

排序用jqueryui，自己看API即可。

[啊，吧，从，的]

startnumber : 2

endnumber : 0

var item = app.splice(2,1);

app.splice(0,0,item);

## 3.2 发帖接口

$("#tijiaobtn").click(function(){

$.ajax({

"type" : "post",

"url" : "/fabu",

"data" : {

"content" : $("textarea").val(),

"picarr" : picupbox.getPicurlarray()

},

"traditional" : true,

"success" : function(){

}

});

});

//发帖

exports.fabu = (req,res) => {

//下面两句话来自formidable，来自：https://www.npmjs.com/package/formidable

var form = new formidable.IncomingForm();

//处理

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

var content = fields.content;

var picarr = fields.picarr;

var tz = new Tiezi({

email : req.session.email,

content : content,

picarr : picarr,

date : new Date()

});

tz.save((err)=>{

if(err){

res.json({"result" : -1});

}else{

res.json({"result" : 1});

}

});

});

};

数据库Tiezi.js：

//引包

var mongoose = require('mongoose');

//创建一个schema

var tieziSchema = new mongoose.Schema({

"email" : String,

"content" : String,

"picarr" : [String],

"date" : Date

});

//创建一个模型

var Tiezi = mongoose.model("tiezi" , tieziSchema);

//暴露

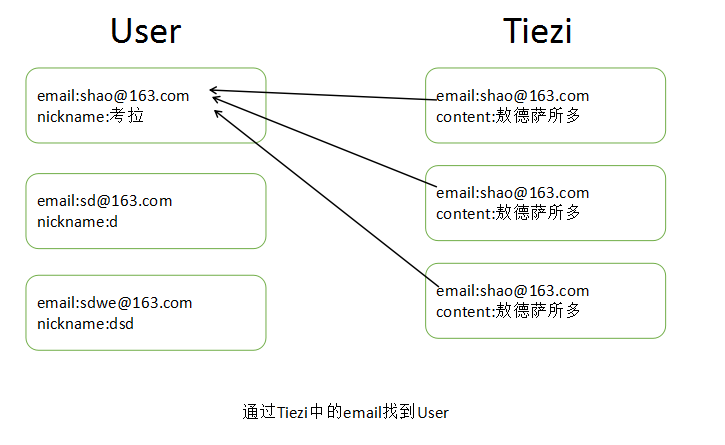
module.exports = Tiezi;

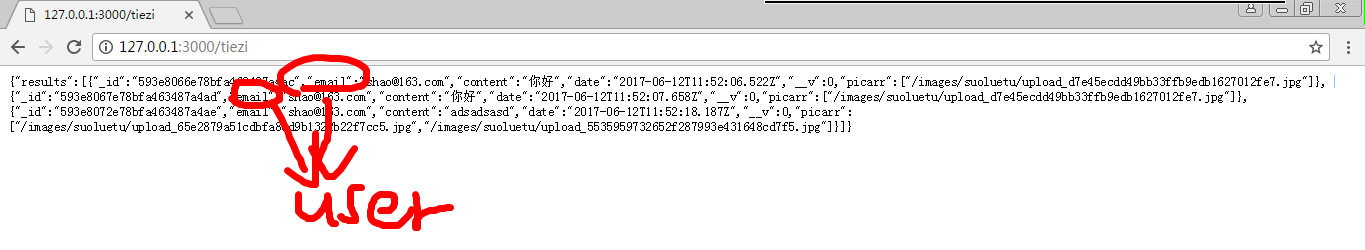
# 四、班级说说项目开发 - 显示帖子

## 4.1 接口

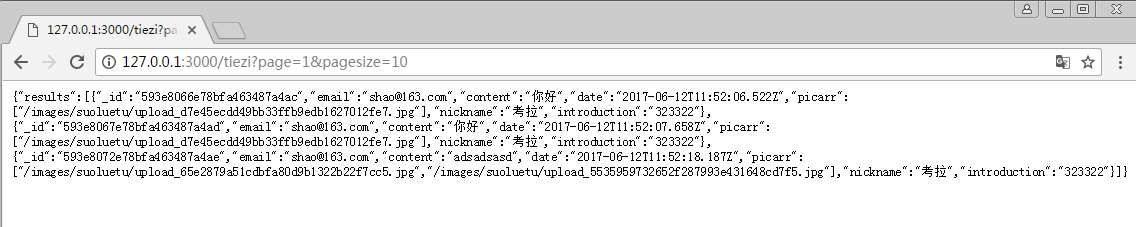
数据库中没有存发帖的人的昵称、一句话简介，只保存了email。

email叫做“外键”。





mongoose中的所有从数据库中新增的对象，都很难加schema之外的键！



得到帖子：

exports.getTiezi = (req,res) => {

var page = url.parse(req.url , true).query.page;

var pagesize = url.parse(req.url , true).query.pagesize;

var count = 0;

var results = [];

Tiezi.find({}).sort(

{"date" : -1}

).skip((page - 1) \* pagesize).limit(pagesize).exec((err,docs) => {

if(docs.length == 0){

res.json({"results" : []});

return;

}

//下面寻找发帖人的信息

docs.forEach((doc)=>{

User.find({"email" : doc.email} , (err , us)=>{

var u = us[0];

//来一个新对象，因为mongoose不让我们添加属性

var o = {

"\_id" : doc["\_id"],

"email" : doc["email"],

"content" : doc["content"],

"date" : doc["date"],

"picarr" : doc["picarr"],

"nickname" : u["nickname"],

"introduction" : u["introduction"]

};

results.push(o);

//计数器

count++;

if(count == docs.length){

res.json({"results" : results});

}

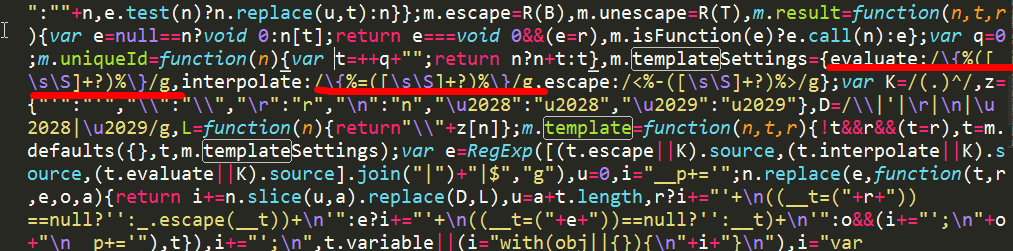
});

});

});

};

underscore要改变源码，更改模板标记。在源码中搜索“template”改变正则即可。



<script type="text/template">

<div class="media">

<div class="media-left">

<a href="#">

<img class="media-object fatierentouxiang" src="{%=avatar%}" >

</a>

</div>

<div class="media-body">

<h4 class="media-heading">{%=nickname%}说：</h4>

<p class="content">

{%=content%}

</p>

<p>

评论(444) 赞(444) {%=date%}

</p>

</div>

</div>

</script>