

如何通过浏览器 JavaScript API 访问服务器数据库

前言

把传统产业的 C/S 和互联网的 B/S，以及基于微信公众号/小程序的 W/S 结合起来实现同源服务器的多客户端的云服务产品设计一直是我这几年研究的方向。我们姑且把这种设计模式称之为 CBS（客户端-浏览器-服务器）开发，WCBS（微信-客户端-浏览器-服务器）开发模式，通过这些开发模式来实现各行业的互联网+。

在本次分享中，我将详细介绍如何使用 webmongo 提供的客户端 JavaScript API 直接在 HTML 页面中访问服务器数据库。通过这种开发模式，你不需要单独编写服务端的数据库操作接口，就可以快速完成中小型动态网站的建设。

webmongo项目简介:

通过客户端访问服务器端 MongoDB 的 JavaScript API。本项目是[dbcloud]的一个分支（<https://github.com/csmake/dbcloud>）。

Accessing server-side mongodb through client javascript API. This project is a branch of dbcloud.

你几乎可以完成绝大部分 MongoDB 调用通过浏览器的 JavaScript API。

You can do almost invoke on mongodb through the javascript API in browser.

客户端的 JavaScript API 支持 IE6.0 +、Chrome、Firefox 和微信（译者注：小程序和公众号）。

The client javascript api support IE6.0+ Chrome FireFox and Wechat.

应用场景:

快速原型设计，无需编写服务器端代码。

Rapid prototyping does not need to write server-side code.

学院派教学和研究，所见即所得。

Academic teaching and research, WYSIWYG.

企业内部应用。

Enterprise internal application.

对数据库表结构安全不关注的其他应用。

Other applications that do not focus on the security of the database table structure.

内容提纲

1. 如何快速搭建能进行数据库操作的网站？
2. webmongo API 如何在 HTML 中进行调用？
3. 如何快速构建能同时被浏览器和微信小程序访问的 JavaScript API。
4. HTML 页面数据库访问性能如何保证客户端的体验不受影响？
5. 申请 csmake 开发者账号，将 Java 库转换为支持浏览器调用的 JavaScript API。

如何快速搭建能进行数据库操作的网站？

我做的第一个Web应用是2004年在清华上学的时候，所谓的电子商务课程大作业。3个人一组，一个负责Server，一个负责Client，一个搞页面美工。小组内经常需要沟通协作，这花费了绝大部分时间，有时候我干脆把其他2个同学的工作也做了，这样可以出一个demo版本，方便大家达成一致意见。

最后项目非常成功，被作为典型还在大课上跟大家汇报演示。做完以后觉得实在是累，前端要html+css+javascript, 后端要java,sqlserver, weblogic这么多东西要弄，新概念很多，十分头疼，而且那个时候对J2EE，动态网页这些刚接触，基本上是稀里糊涂等到毕业的时候才弄明白是怎么回事。所以，我一直在想做网络应用能不能像写单机软件那样，环节少点，沟通少点，效率高点，绝不能跟以前一样那么累了。

2000年左右的时候，能搞几个静态页面，搞搞企业产品宣传就是很NB的了。如果在用javascript做几个特效，flash搞几个动画，那就更高级了。但是随着互联网的推进，交互式应用日渐增多。2007年和一个同学给中关村一个公司做一个电脑配件的供销存管理，那个时候接触到PHP，和JSP类似，可以在静态网页里面嵌入进行动态数据展示。那个项目的核心就是跟数据库打交道。但是那个时候每次的数据库操作都要把页面提交回服务器端。我和搞Server和MySQL数据库端的同学有时候也争论的面红耳赤。原因就是数据库的字段经常修改，逻辑经常修改，他一修改我的页面就得重新搞。而且关系型数据修改起来从数据库定义和数据都要进行兼容性处理。如果不兼容就要修改前端代码。

因此，我们有了这样的想法：

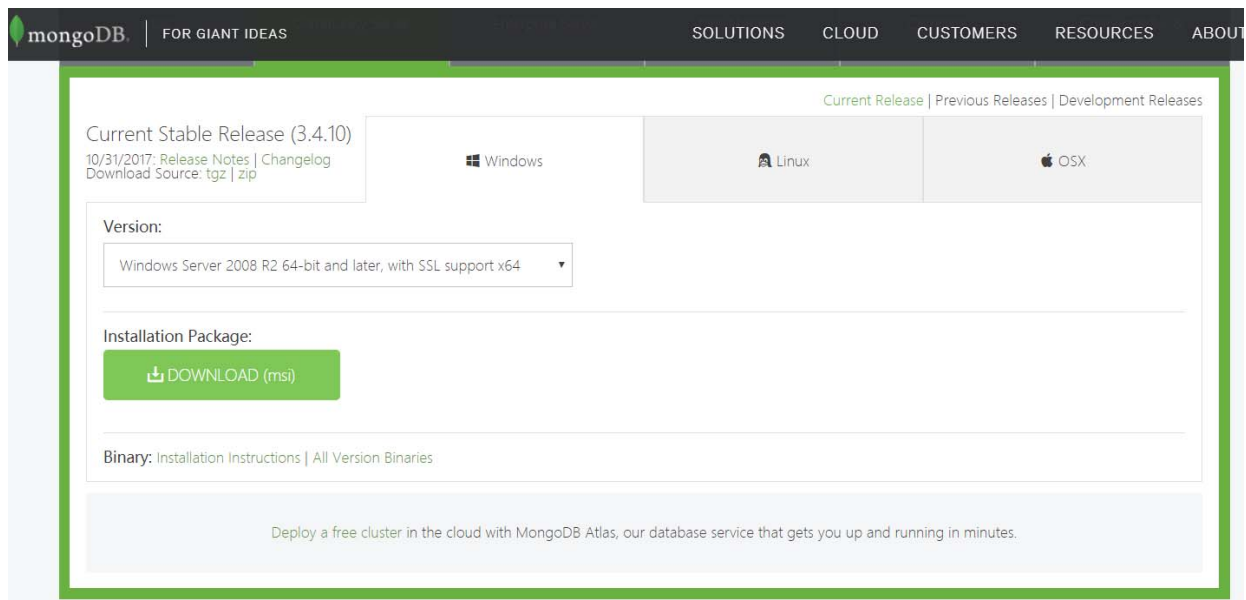
能不能服务器端不用写，如果有逻辑直接在客户端就搞定，这样可以减少服务端人员的投入。而且项目初期，逻辑和数据库变化是常有的事情。开发人员整天抱怨客户需求老变，经历这么多年的项目之后，我发现客户变需求是天经地义的，开始的时候没有一个人能搞的足够清楚，换谁来也是这个结果。

但是数据库一直都是只能在服务器端才能操作，浏览器端如何操作呢？我最近在CSDN上也询问大家，都说没有见过在客户端操作数据库的。

这里我们就来说说如何快速建设支持数据库访问的网站：

1) 下载安装mongodb：

<https://www.mongodb.com/download-center#community>



2) 下载安装JDK 1.8:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Java SE Development Kit 8u151		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.9 MB	jdk-8u151-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.85 MB	jdk-8u151-linux-arm64-vfp-hflt.tar.gz
Linux x86	168.95 MB	jdk-8u151-linux-i586.rpm
Linux x86	183.73 MB	jdk-8u151-linux-i586.tar.gz
Linux x64	166.1 MB	jdk-8u151-linux-x64.rpm
Linux x64	180.95 MB	jdk-8u151-linux-x64.tar.gz
macOS	247.06 MB	jdk-8u151-macosx-x64.dmg
Solaris SPARC 64-bit	140.06 MB	jdk-8u151-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.32 MB	jdk-8u151-solaris-sparcv9.tar.gz
Solaris x64	140.65 MB	jdk-8u151-solaris-x64.tar.Z
Solaris x64	97 MB	jdk-8u151-solaris-x64.tar.gz
Windows x86	198.04 MB	jdk-8u151-windows-i586.exe
Windows x64	205.95 MB	jdk-8u151-windows-x64.exe

3) 下载tomcat8.0:

<http://tomcat.apache.org/>

The Apache Tomcat Project is proud to announce the release of version 8.0.47 of Apache Tomcat. Apache Tomcat 8.0.47 includes fixes for issues identified in 8.0.46 as well as other enhancements and changes. The notable changes compared to 8.0.46 include:

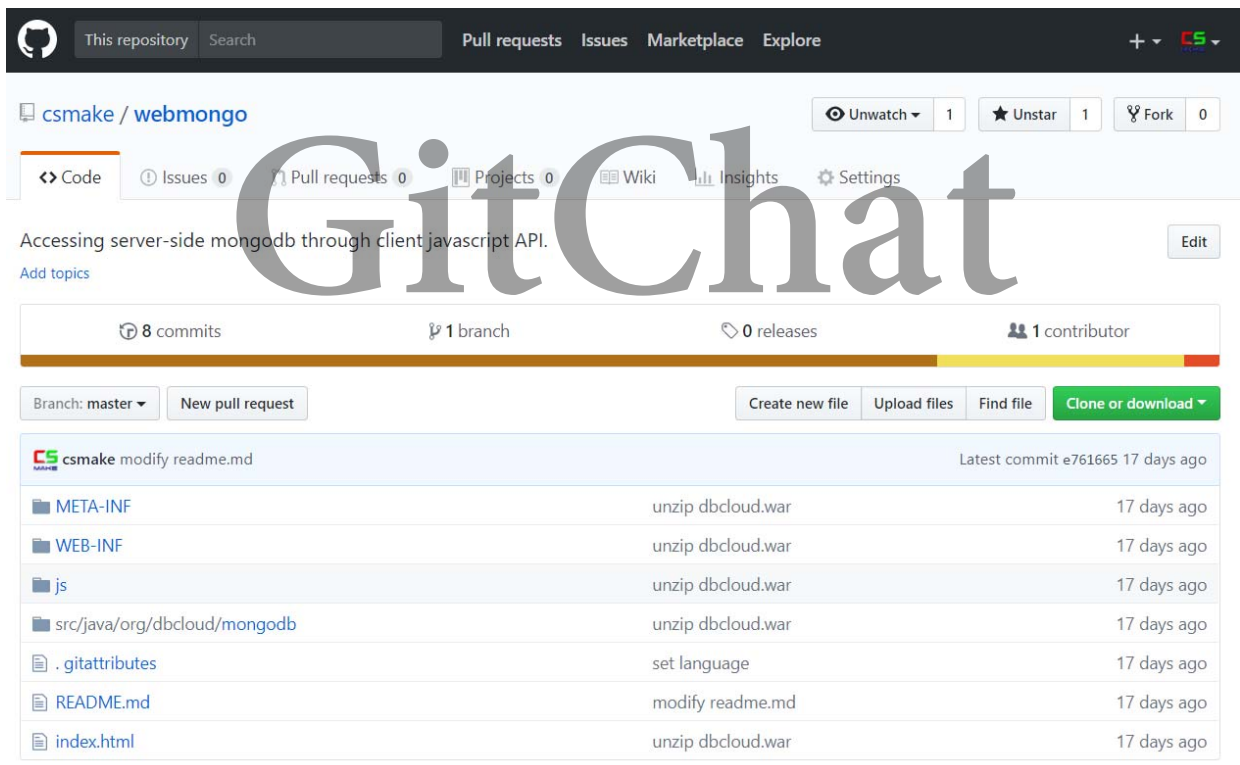
- A fix for CVE-2017-12617.
- Add ExtractingRoot, a new WebResourceRoot implementation that extracts JARs to the work directory for improved performance when deploying packed WAR files.
- Update the packaged version of the Tomcat Native Library to 1.2.14 to pick up the latest Windows binaries built with APR 1.6.2 and OpenSSL 1.0.2l.

Full details of these changes, and all the other changes, are available in the [Tomcat 8 changelog](#).

Note: End of life date for Apache Tomcat 8.0.x is announced. [Read more...](#)

[Download](#)

4) 下载webmongo发行包：到赛思美科官方主页下载（<http://www.csmake.com/>）或者从GitHub上下载（<https://github.com/csmake/webmongo>）。webmongo部署到tomcat webapps 目录



CS make / webmongo

Unwatch 1 Unstar 1 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Accessing server-side mongodb through client javascript API. [Add topics](#) [Edit](#)

8 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

csmake modify readme.md		Latest commit e761665 17 days ago
META-INF	unzip dbcloud.war	17 days ago
WEB-INF	unzip dbcloud.war	17 days ago
js	unzip dbcloud.war	17 days ago
src/java/org/dbcloud/mongodb	unzip dbcloud.war	17 days ago
.gitattributes	set language	17 days ago
README.md	modify readme.md	17 days ago
index.html	unzip dbcloud.war	17 days ago
















我的电脑 > HOME (H:) > tomcat > Apache Tomcat 8.0.27 > webapps >




















名称	修改日期	类型	大小
dbcloud	2017/11/9 8:31	文件夹	
docs	2017/3/14 8:55	文件夹	
examples	2017/3/14 8:55	文件夹	
host-manager	2017/3/14 8:55	文件夹	
manager	2017/3/14 8:55	文件夹	
ROOT	2017/3/14 8:55	文件夹	
webmongo	2017/11/25 15:34	文件夹	
webmongo.war	2017/11/9 10:40	WAR 文件	2,101 KB

5) 启动mongodb和tomcat

↑

C:\Program Files\MongoDB\Server\3.2\bin

名称	修改日期	类型	大小
 bsondump.exe	2015/12/4 20:37	应用程序	4,658 KB
 mongo.exe	2015/12/4 20:42	应用程序	9,699 KB
 mongod.exe	2015/12/4 20:46	应用程序	19,112 KB
 mongod.pdb	2015/12/4 20:46	PDB 文件	156,932 KB
 mongod.exe	2015/12/4 20:37	应用程序	26,302 KB
 mongoexport.exe	2015/12/4 20:37	应用程序	6,341 KB
 mongofiles.exe	2015/12/4 20:37	应用程序	6,189 KB
 mongoimport.exe	2015/12/4 20:37	应用程序	6,460 KB
 mongooplog.exe	2015/12/4 20:37	应用程序	5,915 KB
 mongoperf.exe	2015/12/4 20:46	应用程序	16,506 KB
 mongorestore.exe	2015/12/4 20:37	应用程序	44,884 KB
 mongos.exe	2015/12/4 20:45	应用程序	8,072 KB
 mongos.pdb	2015/12/4 20:45	PDB 文件	85,396 KB
 mongostat.exe	2015/12/4 20:37	应用程序	6,138 KB
 mongotop.exe	2015/12/4 20:37	应用程序	6,001 KB

我的电脑 > HOME (H:) > tomcat > Apache Tomcat 8.0.27 > bin					搜索"bin"
名称	修改日期	类型	大小		
 catalina.bat	2015/9/28 15:19	Windows 批处理...	14 KB		
 catalina.sh	2015/9/28 15:19	SH 文件	21 KB		
 catalina-tasks.xml	2015/9/28 15:19	XML 文档	2 KB		
 commons-daemon.jar	2016/9/27 19:47	Executable Jar File	24 KB		
 commons-daemon-native.tar.gz	2015/9/28 15:19	GZ 文件	201 KB		
 configtest.bat	2015/9/28 15:19	Windows 批处理...	2 KB		
 configtest.sh	2015/9/28 15:19	SH 文件	2 KB		
 daemon.sh	2015/9/28 15:19	SH 文件	8 KB		
 digest.bat	2015/9/28 15:19	Windows 批处理...	3 KB		
 digest.sh	2015/9/28 15:19	SH 文件	2 KB		
 service.bat	2015/9/28 15:19	Windows 批处理...	7 KB		
 setclasspath.bat	2015/9/28 15:19	Windows 批处理...	4 KB		
 setclasspath.sh	2015/9/28 15:19	SH 文件	4 KB		
 shutdown.bat	2015/9/28 15:19	Windows 批处理...	2 KB		
 shutdown.sh	2015/9/28 15:19	SH 文件	2 KB		
 startup.bat	2015/9/28 15:19	Windows 批处理...	2 KB		
 startup.sh	2015/9/28 15:19	SH 文件	2 KB		
 tomcat8.exe	2015/9/28 15:19	应用程序	85 KB		
 tomcat8w.exe	2015/9/28 15:19	应用程序	108 KB		

类型: Windows 批处理文件
大小: 1.97 KB
修改日期: 2015/9/28 15:19

6) 打开浏览器（推荐google chrome），在地址栏录入 <http://localhost:8080/webmongo>




 Git Repository: git


 The Groovy program


 机械制图


 第三方库


 已导入


 网站模板


 招标投标


 讲座, 博客


 公司

```

clear table

insertOne

insertOne and set options

insertMany

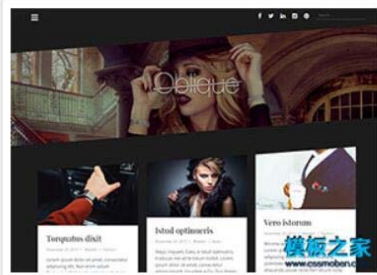
namespace:{"collectionName":"user","databaseName":"test","fullName":"test.user"}

find all

[{"_id":"5a191d9477da182100d33ca6","name":"dbcloud"}, {"_id":"5a191d9477da182100d33ca7","name":"validation"}, {"_id":"5a191d9477da182100d33ca8","name":"dbcloud0","index":0}, {"_id":"5a191d9477da182100d33ca9","name":"dbcloud1","index":1}, {"_id":"5a191d9477da182100d33caa","name":"dbcloud2","index":2}, {"_id":"5a191d9477da182100d33cab","name":"dbcloud3","index":3}, {"_id":"5a191d9477da182100d33cac","name":"dbcloud4","index":4}, {"_id":"5a191d9477da182100d33cad","name":"dbcloud5","index":5}, {"_id":"5a191d9477da182100d33cae","name":"dbcloud6","index":6}, {"_id":"5a191d9477da182100d33caf","name":"dbcloud7","index":7}, {"_id":"5a191d9477da182100d33cb0","name":"dbcloud8","index":8}, {"_id":"5a191d9477da182100d33cb1","name":"dbcloud9","index":9}, {"_id":"5a191d9477da182100d33cb2","name":"dbcloud10","index":10},
  
```

7) 下载网站模板：在百度里面输入“网页模板”，我常用模板之家。将下载的网页模板放入webmongo的web文件夹。将webmongo相关的js包含到相关页面中（详见后面相关章节）

WordPress主题模板



时装模特摄影wordpress博客主题模板



蓝色响应式wordpress个人博客主题模板SKY



企业主题+自媒体主题，仿沙优心理网主题



大气宽屏互联网企业wordpress模板



漂亮wordpress科技新闻博客主题模板下载



fashion时尚搭配主题wp博客模板

webmongo API 如何在 HTML 中进行调用？

快速开始一个项目，基本篇：

- 第一步，启动mongod。
- 第二步，把webmongo.war在部署到Apache tomcat8.0，如果是其他压缩包解压后放到webapps目录中。
- 第三步，打开浏览器（Chrome，Firefox和IE）把HTTP：//本地：8080 / webmongo /，地址可能会有所不同取决于你的设置

开始编写你自己的Web应用程序，进阶篇：

- 第一步：复制mongocollection.java和mongocollectionservlet.java从webmongo / src / java / ...，进入你的项目的src / java / ...
- 第二步：复制webmongo/js目录下所有*.js文件到你项目的js目录。说明：json2.js是为了调试时查看JSON数据格式，非必需。由于webmongo使用了JQuery的Ajax进行数据访问，这个必须。亲测使用jQuery-1.12.4.min.js可以在IE6上正常通过webmongo访问数据库。其他版本未测试。如果不考虑低端兼容，直接使用高版本的JQuery也是没有问题的。
- 第三步：编辑web.xml如下：

```
<servlet>
    <servlet-name>MongoCollectionServlet</servlet-name>
    <servlet-
```

```

class>org.dbcloud.mongodb.MongoCollectionServlet</servlet-class>
    <init-param>
        <param-name>dbhost</param-name>
        <param-value>127.0.0.1</param-value>
    </init-param>
    <init-param>
        <param-name>dbport</param-name>
        <param-value>27017</param-value>
    </init-param>
    <init-param>
        <param-name>db</param-name>
        <param-value>test</param-value>
    </init-param>
    <init-param>
        <param-name>connection</param-name>
        <param-value>test</param-value>
    </init-param>
    <init-param>
        <param-name>user</param-name>
        <param-value></param-value>
    </init-param>
    <init-param>
        <param-name>password</param-name>
        <param-value></param-value>
    </init-param>
</servlet>
<servlet-mapping>
    <servlet-name>MongoCollectionServlet</servlet-name>
    <url-pattern>/org.dbcloud.mongodb.MongoCollection</url-
pattern>
</servlet-mapping>

```

- 第四步：编写你的HTML文件

请查阅webmongo项目[web/index.html](#)示例页面。

webmongo应用示例：

```

<!DOCTYPE html>
<html>
    <head>
    </head>
    <body>
        <div id='log'></div>
    </body>
    <script src="js/json2.js" note='for IE6.0 ,old browser'></script>
    <script src="js/jquery-1.12.4.min.js" note='for IE6.0 ,new version
can use JQuery 2.x'></script>
    <script src="js/org.dbcloud.mongodb.MongoCollection.js"></script>
    <script>
        try {

```



```

function log(msg) {
    $("#log").append("<p>" + msg + "</p>");
}
var table = new org.dbcloud.mongodb.MongoCollection();
log("clear table");
table.deleteMany({});
log("insertOne and set options");
table.insertOne({name: 'validation'}, {validation: true});
log("count:" + table.count({index: {'$exists': true}}));
var data = table.find({'index': {'$gt': 5}}, {projection:
{'_id': 0}});
log("find index > 5 and exclude _id column");
log(JSON.stringify(data));
log("replaceOne");
table.replaceOne({name: 'dbcloud'}, {name: 'replace',
status: 0});
log("updateMany with options ");
table.updateMany({'index': {'$gt': 10}}, {'$set': {name:
'updateMany', status: 100, index: 100}}, {upsert: true, validation:
false});
log("find all");
var data = table.find();
log(JSON.stringify(data));
log('findOneAndUpdate');
log(JSON.stringify(table.findOneAndUpdate({'index': {'$gt':
10}}, {'$set': {name: 'findOneAndUpdate'}, '$inc': {index: -1}})));
table.close();
} catch (e) {
    alert(e.message);
}
</script>
</html>

```

如何快速构建能同时被浏览器和微信小程序访问的JavaScript API

webmongo是csmake工具链的一个示例工程。CSMAKE是一系列工具链。可以将单机版的java/.net类库编译成支持服务器和客户端的代码。服务端的代码可以运行在Tomcat，GlussFish服务器中，同时提供java，C#版本供其他客户端或服务器调用，提供javascript API供浏览器和微信小程序调用。因此程序员可以仅仅关注业务，而无需在开发接口上分心。其他接口的调用与java类库的调用方式完全一致。API级别的调用，各个接口开发人员可以直接参考原java类库的文档。

成为csmake[注册](#)程序员，可以获得免费csmake express版本的使用授权，该版本可以输出javascript客户端代码，直接进行快速网站的开发。

HTML 页面数据库访问性能如何保证客户端的体验不受影响？

webmongo前述的示例都是安装阻塞模式，即顺序执行的方式进行演示的，主要是为了方便描述，实际使用过程中，特别是web应用中网络延迟大，阻塞模式会造成界面卡死，用户体验不好，因此在实际使用中，除非是局域网或者是java/C#等客户端为了简化程序设计而采用阻塞模式外，javascript和微信小程序上的应用都是异步执行的。关于微信小程序的应用可以参考校友云名片小程序设计。



csmake异步调用说明，任何一个由csmake生成的javascript api函数，当第一个参数作为回调函数采用如下callback的格式（包含done,fail,always中任何一个函数）时，自动升级为异步调用，函数的原来的参数依次后移。

```
var callback={
done:function(retValue){},
fail:function(e){alert(e.message);},
always:function(){}
}
**假设有如下java类**
package com.csmake;
public class API{
    public API(String version){
        //指定版本...
    }
    public int add(int a, int b){
        return a+b;
    }
}
**html页面中阻塞式顺序调用**
var api=new com.csmake.API("v1.0");
var retValue = api.add(2,3);
alert("2+3="+retValue);
**html页面中非阻塞异步调用**
new com.csmake.API({
done:function(api){
```

```
api.add({done:function(retValue ){ alert("2+3="+retValue);}},2,3);//  
成员函数的异步调用，也可以用阻塞模式调用如： alert("2+3="+api.add(2,3));  
},  
fail:function(e){  
    alert("API初始化失败，原因是："+e.message);  
}  
},"v1.0");//构造函数也支持异步
```

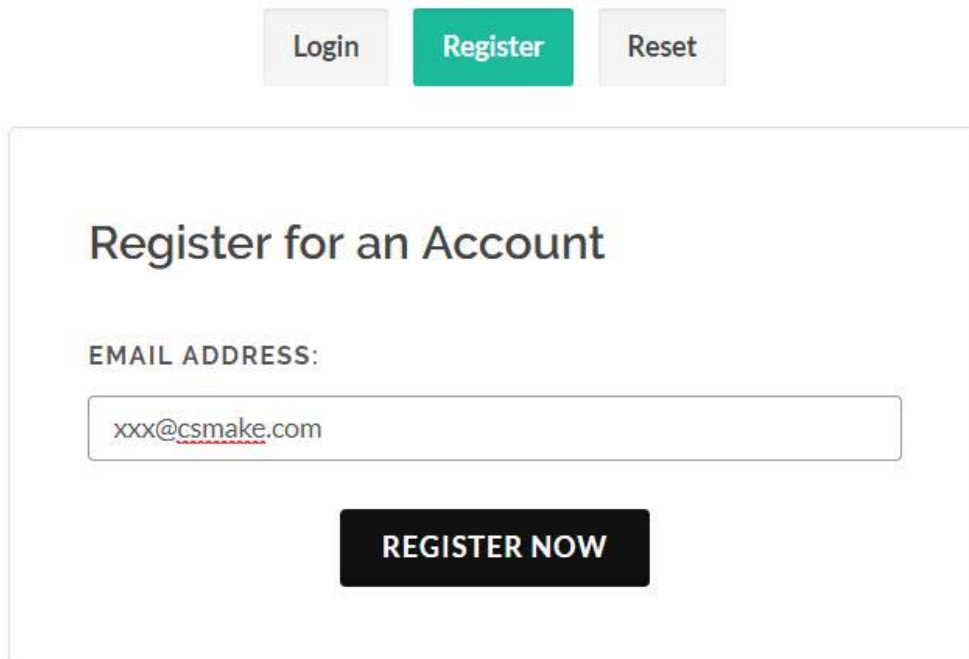
webmongo异步查询示例

```
var table = new org.dbcloud.mongodb.MongoCollection();//我们选择采用阻塞模  
式调用类的构造函数。  
table.find({  
done:function(res){  
alert(JSON.stringify(res));  
}  
});//异步调用了find无参函数
```

申请 csmake 开发者账号

注册时选择成为csmake注册程序员，可以获得免费csmake express版本的使用授权，该版本可以输出javascript客户端代码，直接进行快速网站的开发。

1) 先用邮件注册普通账号



2) 24小时内进入注册邮箱打开激活链接完成信息录入，同时选择“APPLY FOR REGISTERED PROGRAMMER”成为注册程序员。

☒ APPLY FOR REGISTERED PROGRAMMER

VERIFICATION CODE:



3) 再次进入注册邮箱，如果审核通过，会收到用户名(username)和密码(password)作为使用csmake express的凭证。csmake express 部署在云端，使用时必须联网。

4) 现在csmake express版本。



5) 解压 csmake express 查看 readme.txt, 找到 csmake.cfg, 修改其中的 username 和 password。