# Tiny Code 2-Coming from Wave

# 小代码 2-浪沫迩来

又名 《火星人学习报》第2期 代号:浪沫迩来 2019年6月 ~ 2019年9月

http://2293.ml/tinycode/ https://github.com/2293/tinycode/

[TOC]

## 人物风云

世界上最受欢迎的经典大牌程序员

* Bill Gates（1955.10.28, 微软联合创始人）
* James Gosling（Java Creator）
* Richard Stallman（GNU项目创建者）
* Bjarne Stroustrup（C ++ Creator）
* Tim Beners-Lee（1955.06.08-, HTML和WWW发明人）
* Ken Thompson（UNIX共同创作者）
* Linus Torvalds（Linux内核创建者）
* Dennis Ritchie（C编程语言创建者）
* Jack Dorsey（Twitter创作者）
* Ruchi Sanghvi（FB的第一位女工程师）
* Drew Houston（Dropbox Creator）
* Mark Elliot Zuckerberg(1984.05.14-, FB创始人)
* Larry Wall(1954年9月27日-, Perl语言)
* Yukihiro Matsumoto（Ruby翻译）
* John Resig（Jquery，Javascript库）
* Phil Katz（ZIP格式）
* Igor Pavlov (ipavlov, 7-zip creator)
* Rasmus Lerdorf（PHP Creator）
* Niklaus Wirth（Pascal Creator）
* John McCarthy (1927.09.04 - 2011.10.24, 发明Lisp语言)
* Brendan Eich(JavaScript creator, https://brendaneich.com )

## 神奇命令行&冷酷网址

1. 自由钢琴 http://www.autopiano.cn
2. https://music.mli.im/music.web
3. PowerPoint幻灯片制作工具：https://www.powtoon.com/home/
4. GIF动画在线编辑器：https://ezgif.com/
5. 在线图表制作协作平台：https://creately.com/
6. 免费超大文件中转站：https://wetransfer.com/
7. 人工智能自动绘画工具：https://www.autodraw.com/
8. 免费网站幻灯片制作工具：http://wowslider.com/
9. 纯情部落 http://2293.ml

### Windows 10的资源管理器地址栏中输入cmd, 回车则打开命令行窗口并处于当前路径

### C:32>help

获取某个命令的帮助  
C:> help doskey  
C:> doskey /?  
  
列出原生的DOS命令  
C:\Windows\system32>help  
有关某个命令的详细信息，请键入 HELP 命令名  
ASSOC 显示或修改文件扩展名关联。  
ATTRIB 显示或更改文件属性。  
BREAK 设置或清除扩展式 CTRL+C 检查。  
BCDEDIT 设置启动数据库中的属性以控制启动加载。  
CACLS 显示或修改文件的访问控制列表(ACL)。  
CALL 从另一个批处理程序调用这一个。  
...

### 获取安装的驱动程序信息

driverquery -v

### 查看BIOS版本

C:\Users\ljl>wmic bios get smbiosbiosversion  
SMBIOSBIOSVersion  
6QET70WW (1.40 )

### ip地址管理和查询

ipconfig /release  
ipconfig /renew  
ipconfig /flushdns  
ipconfig /all ::可查到mac地址，即以太网适配器的物理地址

### 查看网络活动连接，将为您提供当前打开的端口和相关IP地址的列表

netstat -an netstat -ano|findstr “80” #端口占用查询

### 超级ping，返回一个跃点跟踪列表

C:\>pathping qq.com  
  
通过最多 30 个跃点跟踪  
到 qq.com [111.161.64.40] 的路由:  
 0 xiqi [192.168.0.102]  
 1 192.168.0.1  
 2 192.168.1.1  
 3 10.11.192.1  
 4 183.214.234.9  
 5 \* \* \*  
正在计算统计信息，已耗时 100 秒...

### Windows命令shutdown的详细用法

C:\>shutdown /?  
用法: shutdown [/i | /l | /s | /sg | /r | /g | /a | /p | /h | /e | /o] [/hybrid] [/soft] [/fw] [/f]  
 [/m \\computer][/t xxx][/d [p|u:]xx:yy [/c "comment"]]  
  
 没有参数 显示帮助。这与键入 /? 是一样的。  
 /? 显示帮助。这与不键入任何选项是一样的。  
 /i 显示图形用户界面(GUI)。  
 这必须是第一个选项。  
 /l 注销。这不能与 /m 或 /d 选项一起使用。  
 /s 关闭计算机。  
 /sg 关闭计算机。在下一次启动时，  
 重启任何注册的应用程序。  
 /r 完全关闭并重启计算机。  
 /g 完全关闭并重新启动计算机。在重新启动系统后，  
 重启任何注册的应用程序。  
 /a 中止系统关闭。  
 这只能在超时期间使用。  
 与 /fw 结合使用，以清除任何未完成的至固件的引导。  
 /p 关闭本地计算机，没有超时或警告。  
 可以与 /d 和 /f 选项一起使用。  
 /h 休眠本地计算机。  
 可以与 /f 选项一起使用。  
 /hybrid 执行计算机关闭并进行准备以快速启动。  
 必须与 /s 选项一起使用。  
 /fw 与关闭选项结合使用，使下次启动转到  
 固件用户界面。  
 /e 记录计算机意外关闭的原因。  
 /o 转到高级启动选项菜单并重新启动计算机。  
 必须与 /r 选项一起使用。  
 /m \\computer 指定目标计算机。  
 /t xxx 将关闭前的超时时间设置为 xxx 秒。  
 有效范围是 0-315360000 (10 年)，默认值为 30。  
 如果超时时间大于 0，则默示为  
 /f 参数。  
 /c "comment" 有关重新启动或关闭的原因的注释。  
 最多允许 512 个字符。  
 /f 强制关闭正在运行的应用程序而不事先警告用户。  
 如果为 /t 参数指定大于 0 的值，  
 则默示为 /f 参数。  
 /d [p|u:]xx:yy 提供重新启动或关闭的原因。  
 p 指示重启或关闭是计划内的。  
 u 指示原因是用户定义的。  
 如果未指定 p 也未指定 u，则重新启动或关闭  
 是计划外的。  
 xx 是主要原因编号(小于 256 的正整数)。  
 yy 是次要原因编号(小于 65536 的正整数)。

shutdown /h :: 休眠（hibernate）本地计算机  
shutdown /r /o ::重启你的电脑进入高级启动选项菜单，你可以在这里访问安全模式和Windows恢复实用程序   
shutdown /s /t 30 ::30秒后关闭计算机

### sfc, System File Checker

sfc, System File Checker是一种自动扫描和修复工具，专注于Windows系统文件

### tasklist 和 taskkill

tasklist /svc ::显示与每个任务相关的服务   
tasklist /v ::详细任务列表   
tasklist /m ::定位与活动任务相关的DLL文件  
TASKKILL /F /IM QQ\* /T ::强制结束QQ开头的进程及其子进程  
taskkill /F /pid 3328 /pid 3360 /T

### 发送消息给Windows在线用户，局域网内适用

msg /server:192.168.1.6 \* “你好,聊会儿吧 C:>msg /server:192.168.1.5 \* 好呀”

### 显示完整的无线设备和网络信息

netsh wlan show all

netsh(Network Shell) 是一个windows系统本身提供的功能强大的网络配置命令行工具

### Windows用户管理命令，黑客必知必会

net user cody 123456 /add  
net user cody localgroup Administrators  
net user cody /ACTIVE:YES  
net user cody 654321 :: 把cody的密码改为654321  
net user John /delete

## HTML&Javascript&CSS

### 用fetch在Chrome DevTools 或者 Firefox Devtools的控制台中做Ajax请求

//To GET a text file:  
fetch('http://a-boy.tk')  
 .then(res => res.text())  
 .then(console.log)  
  
// Example POST method implementation:  
postData('http://example.com/answer', {answer: 42})  
 .then(data => console.log(JSON.stringify(data))) // JSON-string from `response.json()` call  
 .catch(error => console.error(error));  
  
function postData(url = '', data = {}) {  
 // Default options are marked with \*  
 return fetch(url, {  
 method: 'POST', // \*GET, POST, PUT, DELETE, etc.  
 mode: 'cors', // no-cors, cors, \*same-origin  
 cache: 'no-cache', // \*default, no-cache, reload, force-cache, only-if-cached  
 credentials: 'same-origin', // include, \*same-origin, omit  
 headers: {  
 'Content-Type': 'application/json',  
 // 'Content-Type': 'application/x-www-form-urlencoded',  
 },  
 redirect: 'follow', // manual, \*follow, error  
 referrer: 'no-referrer', // no-referrer, \*client  
 body: JSON.stringify(data), // body data type must match "Content-Type" header  
 })  
 .then(response => response.json()); // parses JSON response into native Javascript objects   
}  
  
//Chrome Devtools 现在支持一种新的async/await 语法  
const response = await fetch('http://2293.ml/tinycode/TinyCode1.md')  
console.log(await response.text())  
//记得你的请求应该遵从同源策略(same-origin policy)

### 快速掌握nw.js(node-webkit)开发桌面应用

https://github.com/nwjs/nw.js  
Official site: https://nwjs.io

#### 快速入门

新建文件 index.html:

<!DOCTYPE html>  
<html lang="zh">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>livemark</title>  
</head>  
<body>  
 We are using node.js <script>document.write(process.version)</script><br>  
 <input type="text" name="edit1" id="edit1" value="2293.ml">   
</body>  
</html>

新建文件 package.json:

{  
 "name": "livemark",  
 "version": "0.0.1",  
 "main": "index.html"  
}

假设这两个文件放在F: 目录下，然后运行:

$ cd F:\cloud\nwjs\   
$ D:\app\nwjs-sdk-v0.38.3-win-x64\nw.exe livemark

提示: Windows中，你可以把包含package.json的文件夹拖到nw.exe上来运行 还可以把应用打包后再运行

"C:\Program Files\7-Zip\7z.exe" a livemark.zip .\livemark\\*  
D:\app\nwjs-sdk-v0.38.3-win-x64\nw.exe livemark.zip  
ren livemark.zip livemark.nw  
D:\app\nwjs-sdk-v0.38.3-win-x64\nw.exe livemark.nw

#### 发布应用

注意到 D:-v0.38.3-win-x64.json, 它是不带参数运行nw.exe时打开的缺省应用，因此可以将自己的应用合并到nwjs目录下重命名后再次发布。  
还可以将两个文件合并以后再发布 > copy /b nw.exe+app.nw app.exe Linux中合并的命令行为: > catwhich nwapp.nw > app && chmod +x app

具体细节可参考官方文档 https://github.com/nwjs/nw.js/wiki/How-to-package-and-distribute-your-apps

### 快速学习vuejs

**基于examples构建是开发项目的最快方式！**

下面分享一下我学习vuejs的快乐历程：

① 阅读 https://vuejs.org/ ,遇到vue-devtools和ngrok.com, ngrok让我深深着迷，浮想联翩……  
git clone https://github.com/vuejs/vue F:\cloud\opensource\vue by Github Desktop 2.1.0  
因为好奇，实际上我还一口气clone了{vuejs/vue-cli vuejs/vue-devtools vuejs/vuepress cdnjs/cdnjs mjackson/unpkg}

② open F:\cloud\opensource\vue with Visual Studio Code 1.37.0  
F:>py -m http.server 80 http://127.0.0.1/examples/markdown/ examples/svg/ benchmarks/svg/

… … 浏览这些范例, VScode中对照翻阅源码, 很快就领会了Vue.js的创意与重点

③ 接下来再次探索性的研读https://vuejs.org/v2/guide/ , 一边就在vue项目的examples目录中编写了我的第一个vuejs应用examples/hi, 它是这次快乐历程的记录，也是我的笔记。

file: examples/hi/index.html

<!DOCTYPE html>  
<html lang="zh">  
<head>  
 <meta charset="utf-8">  
 <title>hi Vue.js</title>  
 <link rel="stylesheet" href="style.css">  
 <script src="https://unpkg.com/vue"></script>  
</head>  
<body>  
 <div id="app">  
 <p>{{ message }}<br>{{somename}}</p>  
 <p>{{ currentTime }}</p>  
 <input type="text" v-model="somename">  
 <span v-bind:title="somename">这个span的title属性动态展示app1.somename的值</span>  
 <span v-if="seen">Now you see me</span>  
  
 <ol>  
 <li v-for="todo in todos">  
 {{ todo.text }}  
 </li>  
 </ol>  
  
 <button v-on:click="somename= genRandomWord(5)">随机取名</button>  
   
 </div>  
  
 <script>  
 var vm = new Vue({  
 el: '#app',  
 data: {  
 somename:'qwert',  
 message: 'Hello Vue.js! 这是一个多美丽又遗憾的世界 ',   
 seen: true,  
 //控制台中输入 vm.seen=false 使它不显示  
 todos: [  
 { text: 'Learn JavaScript' },  
 { text: 'Learn Vue' },  
 { text: 'Build something awesome' }  
 ]  
 //vm.todos.push({ text: '找到你' })  
 },  
 computed: { currentTime: () => new Date().toLocaleString()},  
 methods: {  
 genRandomWord: (len) => 'abcdefghijklmnopqrstuvwxyz'.split('').sort(() => 0.5 - Math.random()).join('').substr(0, len)  
 }  
 })  
 </script>  
</body>  
</html>

file: examples/hi/component.html

<script src="https://unpkg.com/vue"></script>  
<div id="app-7">  
 <ol>  
 <component1 v-for="x in publishList" v-bind:obj="x" v-bind:key="x.id"></component1>  
 </ol>  
</div>  
<script>  
 Vue.component('component1', {  
 props: ['obj'],  
 template: '<li>{{ obj.organization }} - {{ obj.url }}</li>'  
 })  
   
 /\* vm short for ViewModel \*/  
 var vm = new Vue({  
 el: '#app-7',  
 data: {  
 publishList: [  
 { id: 0, organization: '华章图书', url: 'www.hzbook.com' },  
 { id: 1, organization: '博文视点', url: 'http://www.broadview.com.cn' },  
 { id: 2, organization: "O'Reilly 北京", url: 'http://www.oreilly.com.cn' }  
 ]  
 }  
 })  
 </script>

总结一下， - v-bind ,简写样例<a :href="url"> ... </a> - v-model - v-for - v-if - v-else-if - v-else - v-show - v-on:click ，简写样例<a @click="doSomething"> ... </a> - <a v-on:[eventName]=“doSomething”> … 动态参数 dynamic arguments - {{somename | capitalize }} 这里capitalize是一个filter,首字母大写

//Vuejs模板语法(Vuejs Template Syntax)  
{{ number + 1 }}  
{{ ok ? 'YES' : 'NO' }}  
{{ message.split('').reverse().join('') }}  
<div v-bind:id="'list-' + id"></div>

④ 使用vue-cli > npm uninstall vue-cli -g #如果你安装了以前的版本(1.x, 2.x)  
npm install -g @vue/cli  
vue –version  
3.10.0 vue ui

⑤⑥⑦⑧⑨⑩

### 5分钟掌握Angularjs

<script src="//code.angularjs.org/snapshot/angular.min.js"></script>  
<div ng-app="">  
 <p>名字 : <input type="text" ng-model="name"></p>  
 <h1>Hello {{name}}</h1>  
 <p>{{5+5}}</p>  
</div>

ng-repeat

<div ng-app="" ng-init="primes=[2,3,5,7,11,13,17,19]">   
<ul>  
 <li ng-repeat="x in primes">  
 {{ x }}  
 </li>  
</ul>

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Example - example-guide-concepts-1-production</title>  
 <script src="//code.angularjs.org/snapshot/angular.min.js"></script>  
</head>  
<body >  
 <div ng-app ng-init="qty=1;cost=2">  
 <b>Invoice:</b>  
 <div>  
 Quantity: <input type="number" min="0" ng-model="qty">  
 </div>  
 <div>  
 Costs: <input type="number" min="0" ng-model="cost">  
 </div>  
 <div>  
 <b>Total:</b> {{qty \* cost | currency}}  
 </div>  
</div>  
</body>  
</html>

模块module 控制器controller

<div ng-app="myApp" ng-controller="myCtrl">  
名: <input type="text" ng-model="firstName"><br>  
姓: <input type="text" ng-model="lastName"><br>  
<br>  
姓名: {{firstName + " " + lastName}}   
</div>  
   
<script>  
var app = angular.module('myApp', []);  
app.controller('myCtrl', function($scope) {  
 $scope.firstName= "John";  
 $scope.lastName= "Doe";  
});  
</script>

自定义指令

<div foo-directive></div>  
  
<script>  
var app = angular.module("myApp", []);  
app.directive("fooDirective", function() {  
 return {  
 restrict : "A",  
 template : "<h1>自定义指令!</h1>"  
 };  
});  
</script>

angularjs cheat sheet

https://www.runoob.com/angularjs/angularjs-tutorial.html

### Javascript Tricks and Bookmarklets

// 一些最基本的Javascript技巧  
 !!foo //转为bool值  
 ~~2.235 // 2，取整，~ 是按位取非运算符  
 ~~-2.235 // -2  
 Math.floor(-2.235) // -3  
 ~~ "-3.05" // -3 ， 这里~~代替了parseInt的作用  
 ~~ "3.2E10" // 1935228928  
 +"-12.56" // -12.56 ， parseInt  
 +new Date // 时间微秒值再取整，例如1563763275275  
 2.33 | 0 // 2  
 2.33 >> 0 // 2  
 +!![] // 1 , +!![]===+true===1  
 ++[[]][+[]]+[+[]] == 10  
 v=[] && 10 // 10  
 v=[] || 10 // []  
 [66] + 10; //'6610'  
 10 + [66]; //"1066"  
 [{}] + 10; //"[object Object]10"  
 (10)["toString"]() === "10"  
 Array(3).fill('a') // ["a", "a", "a"]  
  
//Chrome中奇怪的js运算  
parseInt(0.0000008) // 8  
  
// 创建过去七天的数组  
[...Array(7).keys()].map(days => new Date(Date.now() - 86400000 \* days));  
  
// 生成长度为11的随机字母数字字符串  
Math.random().toString(36).substring(2);  
// 4yzqwrvw9iu  
  
//获取URL的查询参数  
//?foo=bar&baz=bing => {foo: bar, baz: bing}  
q={};location.search.replace(/([^?&=]+)=([^&]+)/g,(\_,k,v)=>q[k]=v);q;  
  
//网页上显示本地时间  
<span id="timebar"></span>  
<script>setInterval(()=>document.getElementById("timebar").innerHTML=new Date().toLocaleString()/\*.slice(10,19)\*/) </script>  
  
//混淆数组shuffle  
arr=[...Array(7).keys()]  
arr.slice().sort(() => Math.random() - 0.5)  
  
// 生成随机十六进制颜色代码 如：'#c618b2'  
'#' + Math.floor(Math.random() \* 0xffffff).toString(16).padStart(6, '0');  
  
// 数组去重  
[...new Set(arr)]  
  
//数字从右边起，隔三位加上一个逗号  
'1234567890'.replace(/\B(?=(\d{3})+(?!\d))/g, ',')  
  
//freeze冻结对象，使它不可修改  
var obj = {foo: 'bar'};  
Object.freeze(obj);

// 用字符串返回一个键盘图形  
(\_=>[..."`1234567890-=~~QWERTYUIOP[]\\~ASDFGHJKL;'~~ZXCVBNM,./~"].map(x=>(o+=`/${b='\_'.repeat(w=x<y?2:' 667699'[x=["BS","TAB","CAPS","ENTER"][p++]||'SHIFT',p])}\\|`,m+=y+(x+' ').slice(0,w)+y+y,n+=y+b+y+y,l+=' \_\_'+b)[73]&&(k.push(l,m,n,o),l='',m=n=o=y),m=n=o=y='|',p=l=k=[])&&k.join`  
  
`)()

## HTML速查表

<!DOCTYPE html>  
<html lang="zh">  
<head>  
<meta charset="utf-8">  
<title>网页标题</title>  
</head>  
<body>  
HTML 基本文档。内容......  
</body>  
</html>  
  
基本标签（Basic Tags）  
<h1>最大的标题</h1>  
<h2> . . . </h2>  
<h3> . . . </h3>  
<h4> . . . </h4>  
<h5> . . . </h5>  
<h6>最小的标题</h6>  
   
<p>这是一个段落。</p>  
<br> （换行）  
<hr> （水平线）  
<!-- 这是注释 -->  
<!--[if lt IE 9]>  
 <script src="http://unpkg.com/html5shiv.min.js"></script>  
<![endif]-->  
  
文本格式化（Formatting）  
<b>粗体文本</b>  
<code>计算机代码</code>  
<em>强调文本</em>  
<i>斜体文本</i>  
<kbd>键盘输入</kbd>   
<pre>预格式化文本</pre>  
<small>更小的文本</small>  
<strong>重要的文本</strong>  
   
<abbr> UFO（缩写）</abbr>  
<address> （联系信息）</address>  
<bdo dir="rtl"> （文字方向）</bdo>  
<blockquote> （从另一个源引用的部分）</blockquote>  
<cite> （引用）</cite>  
<del>删除的文本</del>  
<ins>插入的文本</ins>  
<sub>下标<sub>  
<sup>上标</sup>  
链接（Links）  
普通的链接：<a href="http://2293.ml">链接文本</a>  
图像链接： <a href="http://www.example.com/" target="\_blank"><img src="URL" alt="替换文本"></a>  
target: \_self \_blank \_parent \_top rel="noreferrer" rel="noopener" <br>  
<a href="tel:+123456789">Phone</a>  
<a href="mailto:webmaster@example.com">发送e-mail</a>  
  
书签：  
<a name="tips">提示部分</a>  
<a href="#tips">跳到提示部分</a>  
图片（Images）  
<img src="URL" alt="替换文本" height="42" width="42">  
  
样式/区块（Styles/Sections）  
<style type="text/css">  
h1 {color:red;}  
p {border:1px dashed blue;}  
</style>  
<div>文档中的块级元素。  
 HTTP request methods: HEAD GET POST PUT DELETE CONNECT OPTIONS TRACE PATCH  
 <span dir="rtl">אה, אני אוהב להיות ליד חוף הים</span>  
</div>  
<span>文档中的内联元素</span>  
  
无序列表  
<ul>  
 <li>项目</li>  
 <li>项目</li>  
</ul>  
有序列表  
<ol>  
 <li>第一项</li>  
 <li>第二项</li>  
</ol>  
定义列表  
<dl>  
 <dt>dinosaur</dt>  
 <dd>n.恐龙；守旧落伍的人；过时落后的东西<br>  
[网络]恐龙世纪；古生恐龙；小恐龙</dd>  
 <dt>pug</dt>  
 <dd>n.哈巴狗 [网络]巴哥犬；八哥犬；帕格</dd>  
</dl>  
  
表格（Tables）  
<table border="1">  
 <caption>标题</caption>  
 <thead>  
 <tr>  
 <th>列1</th>  
 <th colspan="2">合并的两列</th>  
 </tr>  
 </thead>  
 <tr>  
 <td>1，1</td>  
 <td>1，2</td>  
 <td>1，3</td>  
 </tr>  
</table>  
框架（Iframe）  
<iframe src="data:text/plain;base64,SGVsbG8sIFdvcmxkIQ%3D%3D"></iframe>  
<iframe width="560" height="315" src="https://www.youtube.com/embed/videoseries?list=PLx0sYbCqOb8TBPRdmBHs5Iftvv9TPboYG" frameborder="0" allow="autoplay; encrypted-media" allowfullscreen></iframe>  
表单（Forms）  
<form action="demo\_form.php" method="post">  
<fieldset><legend>栏目1</legend>  
 <input type="text" name="email" size="40" maxlength="50">  
 <input type="password" name="pwd">  
</fieldset>  
  
<input type="range" name="" min="" max="">  
<input type="week" name="">  
<input type="color" name="">  
<input type="checkbox" checked="checked">  
<input type="radio" checked="checked">  
<input type="submit" value="Send">  
<input type="reset">  
<input type="hidden">  
<select name="favFruit" multiple size="4">  
<option>苹果</option>  
<option selected="selected">香蕉</option>  
<option>樱桃</option>  
</select>  
<textarea name="comment" rows="60" cols="20">  
!"#$%&'()\*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^\_`abcdefghijklmnopqrstuvwxyz{|}~  
¡¢£¤¥¦§¨©ª«¬­®¯°±²³´µ¶·¸¹º»¼½¾¿ÀÁÂÃÄÅÆÇÈÉÊËÌÍÎÏÐÑÒÓÔÕÖ×ØÙÚÛÜÝÞßàáâãäåæçèéêëìíîïðñòóôõö÷øùúûüýþÿĀāĂăĄą  
</textarea>  
   
</form>  
实体（Entities）  
&lt; &gt; &amp; &quot; &copy; &trade; &nbsp; &#169;   
 < > & " © ™ space ©   
  
HTML5提供了新的元素来创建更好的页面结构：  
标签 描述  
<article> 定义页面独立的内容区域。  
<aside> 侧边栏内容。  
<bdi> 一段特殊文本方向的文本。  
<command> 定义命令按钮，比如单选按钮、复选框或按钮  
<details> 用于描述文档或文档某个部分的细节  
<dialog> 定义对话框，比如提示框  
<summary> 标签包含 details 元素的标题  
<figure> 规定独立的流内容（图像、图表、照片、代码等等）。  
<figcaption> 定义 <figure> 元素的标题  
<footer> 定义 section 或 document 的页脚。  
<header> 定义了文档的头部区域  
<mark> 带有记号的文本。  
<meter> 定义度量衡。仅用于已知最大和最小值的度量。  
<nav> 导航链接的部分。  
<progress> 任何类型的任务的进度。  
<ruby> 定义 ruby 注释（中文注音或字符）。  
<rt> 定义字符（中文注音或字符）的解释或发音。  
<rp> 在 ruby 注释中使用，定义不支持 ruby 元素的浏览器所显示的内容。  
<section> 节，区段。  
<time> 定义日期或时间。  
<wbr> 规定在文本中的何处适合添加换行符。  
  
<video width="320" height="240" controls>  
 <source src="movie.mp4" type="video/mp4">  
 <source src="movie.ogg" type="video/ogg">  
 如果你的浏览器不支持 video 标签此行将显示...  
</video>  
  
<audio controls autoplay>  
 <source src="horse.ogg" type="audio/ogg">  
 <source src="horse.mp3" type="audio/mpeg">  
 <source src="water.wav" type="audio/wav">  
如果你的浏览器不支持 audio 元素此行将显示...  
</audio>  
  
<canvas id="chessboard" width="150" height="150"></canvas>

## Python 奇巧异技

### 无限字符动画

python -c "while 1:import random;print(random.choice('^\_^'), end='')"

### 一行代码看漫画

import antigravity

### 打印心形图案

print('\n'.join([' '.join(['%s\*%s=%-2s' % (y, x, x\*y) for y in range(1, x+1)]) for x in range(1, 10)]))  
  
print('\n'.join([''.join([('IloveU'[(x-y)%len('IloveU')]if((x\*0.05)\*\*2+(y\*0.1)\*\*2-1)\*\*3-(x\*0.05)\*\*2\*(y\*0.1)\*\*3<=0 else' ')for x in range(-30,30)])for y in range(15,-15,-1)]))

## 数学探索与发现

### 将自然数表示成三个立方数之和

33 = 8866128975287528^3 - 8778405442862239^3 -2736111468807040^3  
42 =(-80538738812075974)^3 + 80435758145817515^3 + 12602123297335631^3  
 1 = (9t^3 + 1)^3 + (9t^4)^3 + (-9t^4 - 3t)^3   
 2 = (6t^3 + 1)^3 + (-6t^3 + 1)^3 + (-6t^2)^3  
(a+b)^3 - a^3 - b^3 = 3ab(a+b)  
a^3+b^3+c^3-3abc=(a+b+c)(a^2+b^2+c^2-ab-bc-ca)  
F[3\*n] = F[n+1]^3+F[n]^3-F[n-1]^3 = 2\*F[n]^3+3\*F[n+1]\*F[n]\*F[n-1] = 5\*F[n]^3+3\*(-1)^n\*F[n] , here Fibonacci Sequence F[n]=F[n-1]+F[n-2], F[0]=0,F[1]=1  
...

0和形如9k±4的数不可写成三个立方数之和，其余的都可以。  
Remark: for n≤1000, the problem is still open only for 114, 165, 390, 579, 627, 633, 732, 795, 906, 921, and 975

### 证明哥德巴赫猜想

2019-09-02,中国科大毕业的程序员 Cody Luo(cody@ ustc.edu)发表了Goldbach’s Conjecture的一个证明，使用Sagemath作为计算文档工具，以“分配各不相同质因数”的方法，发现并证明了两个简洁的不等式：

Goldbach Conjecture Inequality 1: gold(n) < prime\_pi(n)+sigma(n,0)  
gold(n): the min non-negative integer makes that both n-g and n+g are primes  
prime\_pi(n): the count of primes in 1..n  
sigma(n,0): the count of n.divisors()  
  
gold(n) < prime\_pi(n), while n>344   
gold(n) < prime\_pi(n)\*4395/3449751 ≈ prime\_pi(n)\*0.0013, while n>57989356  
  
Goldbach Conjecture Inequality 2: gold(n) < prime\_pi(prime\_pi(n)+n)