安装目录：

1. 固态硬盘C 优先固态硬盘

优点：启动速度快

缺点：系统损坏、资料丢失 借助PE系统（大白菜、优启）提取C盘资料

1. 机械硬盘 分2-4个盘区 一般安装到除了C以外的盘区

优点：保存资料 缺点：繁琐 修改安装目录

搜狗浏览器：D:/programs/sougou

重装系统🡪所有软件重装🡺安装D 使用数据就会还原

下载软件：

官方网站🡺百度软件下载🡺软件下载站（太平洋下载） 普通/本地 下载

安装软件：

1. exe

修改安装目录🡺去掉捆绑软件🡺去掉开机启动、关联软件

1. zip、rar

360压缩 （不要用winrar）

绿色版软件 🡺 移动到常用安装目录

过程：

选择安装目录🡺解包🡺修改注册表/环境变量/开机启动项

注册表：运行🡺 输入regedit 🡺 注册表谨慎修改

环境变量：windows系统快速找到某一个软件的全局变量

计算机🡺属性🡺高级系统设置🡺环境变量

卸载软件：

删除安装目录下文件 🡺 修改注册表/开机启动/环境变量 🡺 成功

软件管家：

重点卸载：

lenovo/ASUS软件 （影音/驱动）

oneDrive

影音播放器、浏览器（保留2个谷歌浏览器、国产浏览器）

迈克菲杀毒、金山毒霸

流氓软件

IIS

SQLserver

mySQL

扩展名/隐藏文件

win7：组织—文件夹和搜索选项—查看—隐藏已知文件类型的扩展名/显示隐藏文件、文件夹

win10：查看--隐藏已知文件类型的扩展名/显示隐藏文件、文件夹

PicPick

PS CS6/CC

HbuiderX

Sublime Text 3

ALT+DELETE 填充前景色

取消选区 Ctrl+D

CTRL+R 显示标尺 => 右键标尺 => 像素

编辑 => 首选项 => 单位与标尺 => 像素

存储为web格式：CTRL+ALT+SHIFT+S

SHIFT多选切片

1. JPG格式

品质范围：60-80

体积：MAX=300KB，特殊情况除外

1. png格式

网页设计规范

max：300kb

.png24 应用场景：图片尺寸小，logo/图标/必须透明的情况

.jpg 应用场景：尺寸较大，适用大部分情况

.gif应用场景：加载动画，loading……

字体

1. 标题/说明 微软雅黑/宋体
2. 修饰、美化 做成图片/改成微软雅黑

字体大小

PC端最小字体：12px

手机端最小字体：没有

灰色、白色、红色、黑色

安全距离：960px-1200px

显示/隐藏参考线：Ctrl+H

手机屏幕：320-414

ipad：640-960

PC端屏幕：960-1920

Retina屏幕 2500-3600

压缩图片

导入图片🡺Ctrl+Alt+Shift+S 存储为web格式🡺格式（jpg/png24）🡺品质（65-75）🡺存储

参考线：

显示/隐藏参考线：Ctrl+H

1. 标尺 Ctrl+R
2. 删除参考线：移动工具拖拽参考线到标尺上
3. 添加参考线：左键拖拽标尺，自动生成参考线

jpg切片

切片：拖拽

切片/切片选择 工具切换：按住Ctrl

多选切片：按住shift

复制切片：按住alt

删除切片：切片选择工具🡺选中切片🡺delete

切片选择工具：

设置切片：双击切片🡺选项：切片名称、宽、高

存储为web格式

1. 注意：每一个切片单独一种格式设置，需要挨个设置格式或者shift多选切片🡺设置格式
2. 重命名切片，不允许中文或者特殊符号
3. 多选切片🡺存储🡺选中的切片

png透明切片 图片体积容易很大

1. 新建 透明图层文件
2. 移动工具选择需要切片的图层，将其移动至新建的图层文件中，会自动复制图层
3. 切片工具，切片
4. 存储为web格式🡺逐个选择切片🡺格式.png24🡺存储🡺选中的切片

如果是多个图层

1. 多选（Ctrl/Shift）相关的图层n个
2. 移动工具+Alt，复制图层至新建透明文件里面
3. Ctrl+Shift+F 格式化代码
4. Ctrl+Alt+S 全部保存代码

foxmail/outlook

调取资源文件：Alt+/

强行换行：Ctrl+Enter

删除一行：Ctrl+D

Git使用

应用场景：管控代码，版本控制软件

衍生物：Github（世界程序员交友网站） 码云、Coding

1. Git 下载 官网
2. 小乌龟TortoiseGit 安装包/汉化包

使用方法：

1. 在任意磁盘创建一个文件夹（中/英），右键🡺创建版本库
2. 设置右键菜单：在文件夹里，右键🡺设置🡺常规设置🡺右键菜单
3. 设置远端并拉取（获取最新代码）：右键🡺拉取🡺管理远端🡺获取git仓库地址🡺url🡺确定
4. 提交代码：

第一种情况：第一次提交

* 1. 拉取 右键🡺拉取🡺确定
  2. 提交 右键🡺提交🡺填写日志🡺全选文件🡺提交并推送🡺确定
  3. 推送（合并到第二步）

第二种情况：第二次以后的提交

1. 拉取 右键🡺拉取🡺确定
2. 推送 右键🡺推送🡺确定
3. 账号：silky 密码：152922jxm
4. 修改git客户端--账户/密码

控制面板🡺用户账户🡺凭据管理器🡺普通凭据🡺查找到git.dev.tecent.com🡺编辑🡺重新输入账号密码

1. 删除/修改

正常删除/修改文件🡺提交🡺推送

在没有border的情况下，优先使用padding，兼容性好，没有副作用

如果改变a链接的颜色，必须使用选择器定位到a标签，设置颜色，父级设置颜色无效

div包裹img，意外出现4px空白，解决方案：img的display:block

图片和文字 垂直居中对齐：

给每一个内联元素添加CSS属性——vertical-align:middle

1. 单行文本截断

width 宽度

overflow：hidden 隐藏多余文字

white-space：nowrap 强制一行显示

text-overflow：ellipse 出现省略号（…）

1. 多行文本截断

max-height

line-height

overflow：hidden

1. img标签 应用场景：灵活多变，经常修改图片
2. background背景图

应用场景：固定，不经常修改 图标-雪碧图、Banner图片、背景图

图片左右居中

1. padding
2. text-align:center 内联元素有效

列表与浮动结合使用

1. 列表li的class类必须一样
2. 清除浮动 <div class=”clear”></div>
3. 左右两边元素贴边处理：
   1. 人为的加宽父级元素的宽度
   2. margin-left：-20px；特殊情况

* display:none 元素消失、不占据位置
* visibility:hidden 内容消失、占据位置 效果相当于opacity：0
* opacity：透明度 0~1 缺点：子元素也会透明

return语句

1. 终止后面的语句执行。
2. 函数返回数值

读 🡺 改 🡺 写 🡺 分享

手机端页面：

1. 设置viewport

<meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1,maximum-scale=1,user-scalable=no" />

1. 处理文字，字号

px 绝对单位

% 相对单位 相对于父级元素

em 相对单位 相对于父级元素

rem相对单位 相对HTML标签 flexible.js 750px设计图 rem值==实际px值/75px

vw /vh 相对单位 相对于视图 750px设计图 vw值 == 实际px \* 100vw / 750px

1. 边框border的处理 伪元素::after transform:scaleY(0.5)
2. 处理图片，自动缩放

img标签的图片 添加宽 高 vw单位

雪碧图 使用px单位 所有相关数值（背景图尺寸、图标宽高、X/Y坐标）除以2

1. 超大图、小尺寸图片 适配

移动端适配方案

1. PC端一套 + 手机端一套 应用场景：复杂界面
2. 三端（PC端 + 手机端 + iPad端）共用一套：响应式布局 应用场景：简单布局 对设计要求较高

响应式常规方法

1. 媒体查询+CSS属性 应用场景：简单布局
2. 切换代码片段 应用场景：复杂而且HTML结构都要修改

WebGL

canvas svg API

3D游戏：layaBox（微信小游戏/app/pc）、cocos2d、白鹭引擎（Egret）

图表类：hichart、echart、D3.js、Go.js

Three.js 3D展示

在指定目录打开cmd窗口：

在目录上，按住Ctrl+Shift，右键🡺在此处打开命令窗口（Power Shell）

Express 最流行框架

Koa 先进技术

Egg 基于Koa 阿里巴巴

npm start 在package.json文件script定义运行语句

npm run \*\*\* npm run build/ npm run test

JSON 数据

API测试工具：POSTMAN

关系型数据库：MySQL、SQL Server、Oracle

非关系型数据库：MongoDB

两个AJAX发送数据，如何保证顺序执行：

1. 改成同步传输async:false
2. 使用回调函数嵌套执行

富文本编辑器：

wangEditor

ueditor

Vue组件：

data：组件内部使用的数据

props：组件外部传入的数据

Emmet插件 tab 展开代码

ctrl+shift+S 保存所有的文件

ctrl+enter 强制下一行

ctrl+shift+F 格式化代码 （手动写空格、折行、tab）

[www.baidu.com](http://www.baidu.com) 12.220.56.23:3000

Ctrl+[ 选择区块

注释快捷键 ctrl+/ 取消 设置

相对地址:本地文件

img/bg.jpg 当前目录

../img/bg.jpg 返回上一级目录

绝对地址：<http://www.baidu.com/news.html>

ctrl+S 保存

ctrl+shift+S 全部保存

alt+/ 选择路径

ALT+/ 重新选择路径

邮件链接：需要邮件客户端（foxmail/）配合

tr==table row 行

th==table header 表头

td==table data 单元格

thead

tbody

action=”接口地址url” method=”提交数据的方式GET/POST”

type=button

type=submit

type=reset

demo测试示例

git提交步骤：

1. 拉取🡺提交🡺写好日志🡺提交并推送
2. 拉取🡺提交🡺文件列表里面空，说明已经提交成功🡺右键推送
3. 显示隐藏文件夹
4. 找到.git文件夹，复制里面的config文件至别的位置
5. 回到上一层，删除所有文件/夹，包括.git
6. 重新建立版本库，将config放回原先的位置，重新拉取

外部样式表 项目常用

内部样式表 一般用来做demo

内联样式 一般js操作产生的，提高优先级

后代选择器跳级

id 一般用在js编程

div.cube 选择器指 div上面有一个class=”cube”

padding没有副作用，优先使用

margin 有副作用，margin合并

注意 文字溢出的情况

浮动和清除浮动成对出现

清除浮动一般放在最靠近浮动元素的尾部，清除浮动只对同级元素有效

检查是否正确清除浮动，看父级元素的高度，高度==0，浮动没清除

有效宽度：1000-1200 保证网页在任何PC上都能看得见

定位：left/right/top/bottom

4个方位，上下取一个，左右取一个

定位使用时，通常需要设置width、height

一般情况下，4方位不要与margin同时定义，同一个标签上浮动与定位不要一起使用

定位一般都需要宽高，除非内部有东西撑开他

绝对定位：

1、遮罩

1、先选参照物，参照物加relative

2、定位4方位

z-index可以应用到任何有定位的元素上面，relative/fixed/absolute

图片和文字垂直对齐

1. 相对定位
2. 内联元素的垂直对齐 vertical-align:middle

a链接里面图片img占据的空间多了2px高度，

解决方法：

1、给a的字体大小font-size:0

2、直接改变a的display:inline-block，定死高度

伪类：不是正式的class，浏览器支持，没法自定义

a链接的4个状态的顺序必须依次写

伪元素：不是正式的元素，扮演了一个元素的角色，浏览器自动生成

必须有content样式属性

雪碧图在创建的的时候，图片尺寸尽量有冗余，雪碧制作完成之后，尽量不要修改尺寸大小

假如雪碧图已经满了，可以创建新的雪碧图，把剩余的图标放置到新雪碧图里

雪碧图制作：

1. 移动工具，拖放图标至png里面，摆放图标
2. 为了后期修改，保存雪碧图成PSD,
3. 导出雪碧图至设计图目录，png格式

雪碧图的使用

1、打开PSD文件，

提高样式优先级

1. 添加父级class +10
2. 添加内联样式
3. 添加！important

雪碧图（IE8/IE9以上）

字体图标（IE10以上）

手机版

ECMAScript==基础语法

DOM==HTML/CSS操作

BOM==浏览器操作

script写到任何地方，一般写到文件的尾部，方便SEO

alert 调试，弹出字符串

键值对

变量命名的方法：

1. 驼峰式写法

var str=“你好”；

var firstName=”john”

1. 匈牙利写法

var sName=”hello”;

var nTotal=32;

字符串连接，使用+号；

NaN==Not a Number

数组==特殊化的对象

全局变量：

局部变量：function内部定义的变量

数组和for循环最配

1. 确定是什么对象，确定是什么数据类型
2. 查找对应的对象属性/方法
3. 根据需求使用对应的方法

split（“,”） 以规律性的间隔字符，切割字符串，返回一个字符串数组 字符串🡺数组

join（”-”） 以规律性的间隔字符，组合字符串，返回一个字符串 数组🡺字符串

计算从0~2021之间的闰年，将结果做成一个数组，return出来

1. 能被4整除但不能被100整除
2. 被400整除
3. for循环，push方法

* 若 a 小于 b，在排序后的数组中 a 应该出现在 b 之前，则返回一个小于 0 的值。
* 若 a 等于 b，则返回 0。
* 若 a 大于 b，则返回一个大于 0 的值。

get方法 获取

set方法 设置

DOM学习方法/编程思路

1. 先查找具体元素(element对象)
2. 再使用具体的element的属性/方法
3. （可选）绑定具体的事件

复数==很多个

1. 理清思路 （经验）🡺大量的练习，看别人源代码
2. 拆解小目标（小功能）==4
3. 结合知识点完成小目标
4. 组合==衔接

return 语句之后的代码都不会执行

创建元素3步

1. 创建标签，creatElement();
2. 修饰标签
3. 将标签插入 appendChild()/ insertBefore()

推荐网站：

w3cschool

菜鸟教程

js书籍

javascript高级程序设计 （PDF）

1. 鼠标事件 结合事件对象event
2. 键盘事件
3. 窗口改变大小

addEventlistener 谷歌火狐都可以使用

attachEvent IE专用

检测用户使用的浏览器版本，提示用户升级/做兼容性处理

项目 🡺功能+用户体验

定时器

滚动、动画、过程事件

重复添加定时器

1. 获取当前时间 按照小时：分钟：秒 打印到页面上
2. 上面的过程写一个function,使用定时器不断运行

cookie本质上，在多个页面之间共享数据，浏览器提供支持。

session会话

JSON.parse(str) 将字符串转化成对象

JSON.stringfy(obj) 将对象转化成字符串

初学者：

工具类（面向对象）

高级：封装插件🡺框架

E=MC2

连缀：连续使用多个方法，只要操作主体不变

简写：$(selector).mousedown()==底层实现：$(selector).on(“mousedown”);

on方法可以绑定多个“click”事件，也可以绑定多个不同的事件（mousemove+click+keydown）;

off移除事件也需要指明是哪个函数；

内置三组动画只是将animate方法里面的参数预定好之后，简写之后的名称

fadeIn()/fadeOut() 关键性属性：opacity （透明度）缺点：将子孙元素也透明

slideUp()/slideDown() 关键性属性：height

匹配数组的下标index有两种：

1. 选择器——$(“input:eq(1)”) 查找第二的input
2. 方法——$(“input”).eq(1) 查找第二个

jQuery数组的循环

1. $(selector).each()：jQ对象可以使用
2. $.each()： js数组和jQ对象都可以使用

/^([a-zA-Z0-9\_-])[+@([a-zA-Z0-9\_-])+(.[a-zA-Z0-9\_-])+/](mailto:+@([a-zA-Z0-9_-])+(.[a-zA-Z0-9_-])+/)

GET方法：IE浏览器对URL长度限制2K，从服务器获取数据的时候一般是GET

POST方法：服务器限制不一，一般是2M。发送数据给服务器的时候一般是POST

js对象 包含element的属性和方法

js对象🡺jQ对象：$(js对象) 例如：$(this)

jQ对象：在js对象的基础上做了一次封装，添加了很多jQuery的属性和方法

jQ对象🡺js对象：添加索引下标：$(“button”)[1]

jQuery的data()方法只能存储在本页面，没法网页之间共享数据，在页面上的一个公共变量，逃脱闭包

2D/3D效果，一般推荐使用3D变换，原因最大化提高动画的执行效率，让浏览器调用GPU运算

字体图标

1. 阿里图标库：现成、上传生成
2. Font Awesome
3. Glyphicon Halflings

全局样式

字体大小

字体颜色

背景色

浮动、清除浮动

文字对齐

响应式图片

<http://127.0.0.1:3000/mobile/index.html>

http://**本机IP:3000/mobile/index.html**

**4个方位上下取一个，左右取一个**

**4个方位不要与margin同时使用，特殊情况**

**绝对定位：**

1. **先选参照物，添加relative**
2. **定位，width/height**

**数字对象**

**数组对象**

**字符串对象**

**Date对象**

**Math对象**

**regExp对象**

**学习方法：**

1. **确定好你要处理的变量的数据类型 typeof**
2. **根据数据类型查找对应的属性、方法**
3. **查看语法结构，编写代码**

**split(“”) 字符串🡺数组**

**join(“”) 数组🡺字符串**

**build 🡪 CLI配置文件**

**config 🡪 用户自定义配置文件**

**src 🡪 所有组件在此编写**

**assets 🡪 项目必须的资源文件，静态文件:图片（图标、img）/css(插件里面需要的)/字体图标/js（插件需要的）**

**components 🡪 组件**

**router 🡪 配置路由**

**App.vue 🡪 放置公共js、公共CSS、公共HTML、路由**

**main.js 🡪 webpack入口文件、配置vue组件**

**static 🡪 放置静态文件，基本上是测试图片**

**index.html 唯一的一个HTML文件，起始页，容器**

**@在vue-cli的路径里面表示：文件夹src(相对路径)**

**vue-router**

1. **路由定规则**
2. **路由传参**
3. **嵌套路由**

**vue UI框架**

**element 后台管理系统框架**

**iview 后台管理系统框架**

**Mint UI 移动端框架**

**1.新数据库CRM**

**2.**

**复制粘贴旧项目**

**express指令生成新项目，依赖包需自己安装**

1. **修改数据库连接，改成新的数据库名CRM**

**assets 图片路径：以你所在模块为起点**

**static 图片路径：必须以“/static”为起点开始**

**如何引入公共CSS文件 在App.vue使用@import语法**

**项目开发最后一步，打包：npm run build 在dist文件夹下所有的文件，就是放置到服务器端的文件。**

**例如express+node开发的后台，放置到public文件夹下即可。**

**npm cnpm 不能同时用于一个项目安装**

**Angular基础语法**

1. **数据绑定**

**{{data.name}}**

1. **for if**

**\*ngFor \*ngIf**

1. **属性绑定**

**[src] [href]**

1. **事件绑定**

**（click）(submit)**

1. **双向数据绑定**

**[(value)]**

**Service使用**

1. **创建service**

**injectable**

**export class serviceName{}**

1. **在module层级providers这一项申明**
2. **在component层级，class类constructor里面依赖注入**

Angular做网络代理

1. 项目根目录，新建proxy.conf.json，

{

"/api":

{

"target": "http://localhost:3000",

"secure": false

}

}

1. 修改启动指令npm start

"start": "ng serve --open --proxy-config proxy.conf.json"

mui框架 html+css+H5++js

H5+ 封装安卓 IOS两个系统的方法

ionic= Cordova+Angular4

mui= html+css+H5++js

APPcan APPcloud

浮动

1、浮动、不浮动 判定标准：符不符合DOM流 浮动滥用

2、浮动+清除浮动

3、清除浮动的方法：

1. <div class=”clear”></div>
2. 伪元素清除：clearfix::after{}

定位

1. 定位滥用
2. 动画结构、遮罩

JS对象使用

1. 判断数据类型 6种对象
2. 根据数据类型，查找对应的属性和方法
3. 根据需要的方法(如split)，查找对应的语法参数

数组🡺字符串 join()

字符串🡺数组 split()

正则表达式 （看懂，修改）

js事件委托

event.target

白鹭

layabox

three.js 3D展示、小游戏

手机版页面

1. 设置meta viewport
2. 元素宽度不要使用px,尽量使用百分数，高度不受限制
3. 普通图片尺寸 max-width:100% width:100% ，依赖外层元素宽度显示
4. 图标（图片）= 实际尺寸/2
5. 修改盒模型 box-sizing:border-box

width=border+padding+content

响应式页面 （PC+iPad+手机）

PC端 + 手机端

1. 并不是所有页面都适合做响应式，需要UI设计 + 前端配合
2. 某些功能并不能通过修改CSS就能达到响应式，可以2块相同内容隐藏/显示切换来达到目的
3. 大部分通过媒体查询，修改CSS达到目的
4. 字体大小的调整，我们通过REM单位调整,先设置HTML基准字号

字体大小单位

px

相对单位：%（父级）、em (父级) rem（root em）（HTML的字号）

Glyphicons

font Awesome ==fa

阿里图标

库 ：函数jQuery/zepto

UI框架：界面组件 bootstrap、layer UI、amaze UI

bootstrap v2 严重bug 兼容性好

bootstrap v3 合理 生态良好 float

bootstrap v4 优先手机版、辅助类丰富、flex-box

reset.css 重置浏览器样式在bootstrap里面不再使用，内置

1. 全局样式

辅助类样式（\*\*\*\*\*）

字体颜色（灰、红）+**白色**

背景（无）：白+ logo主色

浮动（pull-left、pull-right）

清除浮动（clearfix）

~~字号（h1-h6）~~

加粗（strong）

文本对齐（text-left、text-center、text-right）

列表（无样式列表/内联列表）

栅格系统（\*\*\*\*）

xs – 手机

sm – 平板

lg – 大屏幕

1. 媒体查询
2. 百分比宽
3. box-sizing:border-box 左浮动

使用方法：

1、外壳container

2、row

3、col-xs-(X)

响应式工具（\*\*）

某些特定屏幕下隐藏

hidden-xs

……（\*）

一定不要修改本来的样式名，附加自定义样式，提高优先级来修改

1. 组件

字体图标、下拉菜单（3部分）、输入框组、

导航条：

navbar-header：菜单、logo

navbar-collapse: navbar-nav（普通导航链接）navbar-left/navbar-right

缩略图、列表、面板组合

1. js组件

启动方式：

data启动：data自定义属性 data-\*\*

js启动：使用js代码来启动

以上两种启动方式不能同时使用

模态框、下拉菜单、滚动侦测、标签页、collapse、轮播图（自己换成swiper）

amaze UI框架

bootstrap基础🡺amaze UI

体积小、功能少 重、功能全

1. 辅助类（\*\*\*\*\*）

am-container

浮动/清除浮动

垂直居中（内联元素）

margin/padding 层叠

字体颜色（红色/灰色）

字号

文字对齐

单行文字截断

栅格系统（\*\*\*\*）

| **Class** | **区间** |
| --- | --- |
| am-u-sm-\* | 0 - 640px |
| am-u-md-\* | 641px - 1024px |
| am-u-lg-\* | 1025px + |

<div class="am-g doc-am-g">

<div class="am-u-sm-6 am-u-md-4 am-u-lg-3">sm-6 md-4 lg-3</div>

<div class="am-u-sm-6 am-u-md-8 am-u-lg-9">sm-6 md-8 lg-9</div>

</div>

居中列

栅格，单元格不要定义额外的多余样式，在里面div单独写

等分栅格：

1、必须使用ul/ol

2、等分定义class附着到ul/ol，里面不需要定义

响应式工具（\*\*）

某些特定屏幕下隐藏

图片

1. am-img-response 修正图片溢出
2. 圆形图片 （img正方形）
3. 样式组件（只看样子，不需要记代码）

按钮、表单、表单

小徽章、图标、输入框组、列表（\*）、分页、面板、缩略图、CSS动画

1. js插件（\*\*\*\*\*）记忆功能

折叠面板、下拉菜单、模态框（\*\*\*）、轮播图、侧边栏、滚动侦测、平滑滚动（回到顶部）、固定元素、选项卡、日期组件（\*\*\*\*\*）、表单验证（\*\*\*\*\*）

日期组件：提取日期（value）、赋值日期、禁用日期

表单验证：

启动表单验证:（data-am-validator/js启动）

常用属性:（required、pattern（email、url、number）、minlength/maxlength）

验证方法：.validator('isFormValid')

自定义验证（扩展正则库）：

1. 定义正则表达式 必须DOM ready之前
2. js-pattern-XXXX 附加class

node.exe

nodejs

webkit==Blink

IE == Trident

火狐 == [Gecko](https://baike.baidu.com/item/Gecko)

vue Angular react

Express

Koa

EGG 阿里系基于Koa

MongoDB Mongoose MongoDB

MySQL MySQL

wangEditor 88

Ueditor

插件失效 ajax回调函数里面

原文输出 {{@xxx}}

Angular1（Angularjs）脏检查🡺真DOM

Angular2+ 重构 2 4 5 (ng-core+ng-router+http+测试)

ES6🡺Typescript(微软ES6扩展：类型检查)🡺Angular

1. 提高优先级，添加父级class
2. 如果不能提高，继续添加祖先元素
3. 如果还不能提高，采用ID选择器
4. 最后采用！important,慎用

库：函数的集合

框架：UI组件+功能函数 bootstrap