

style 修饰
width 宽度
height 高度
title 想说明的
text-align 水平对齐方式
center 居中 left 居左 right 居右
line-height 垂直对齐方式/ 行高 当行高等于高度时候垂直居中
background 背景
image 引入背景图片(url)
background-position: 背景定位(x 轴, y 轴;)
background-repeat 背景平铺
:no-repeat; 不平铺 repeat-x; 水平平铺 repeat-y; 垂直平铺
background:url() repeat left center #00;
顺序: 背景图片 重复 定位 颜色
top/ right/ bottom/ left/ center 方向英文: 上、右、下、左, 中
color 颜色

font 文字
font-family 字体
font-size 字号
font-weight: normal 正常 :bold 加粗
font-style: normal; 取消倾斜 :italic; 斜体
font:italic bold 24px/ 24px '楷体';
顺序: 是否斜体 是否粗体 字体大小/ 行高 字体
文本修饰属性:
text-decoration:
overline 顶划线
underline 下划线
line-through 中划线/ 删除线
none 无/ 取消下划线
display 转化显示模式 元素转化
block 块
inline 行内
inline-block 行内块

省略号
white-space: nowrap 强制不换行
overflow: hidden 溢出隐藏
text-overflow: ellipsis 省略号
font-family: '微软雅黑' 让省略号在下面
必须给宽度
伪类
a:link 设置未访问时候的颜色 点击后变成 a 的正常颜色
a:visited 设置访问过后颜色
a:hover 当鼠标放上去会变颜色
a:active 鼠标点击变颜色
顺序 lvha 驴哈

auto 自动居中
margin 外边距

margin- top 上外边距
margin- right 右外边距
margin- bottom 下外边距
margin- left 左外边距
padding 内边距
padding- top 上内边距
padding- right 右内边距
padding- bottom 下外边距
padding- left 左外边距
float 浮动 : left|right|none 左|右|不浮动
cursor: pointer 小手的代码。。
clear : left|right|both 清除浮动
overflow : hidden 溢出隐藏

index 首页
base 里面写清除默认样式的
text- indent : em 首行缩进 (LOGO 名字)
以后 logo 必须用 H1 引背景图片 里面写上字
position : relative|absolute|fixed 定位
relative 相对定位
absolute 绝对定位
fixed 固定定位
Left/ top/ right/ bottom 定位的偏移量
z- index 改变标签 z 轴堆叠顺序
opacity 透明代码 1 不透明 (0.9 0.8。。。) 0 透明
filter : alpha (opacity : 0- 100)

input {outline: none} 取消焦点

< form action= "链接" method= "">
post 相对安全 get 不安全
input 表单 单标签
type= "text" 账号/ 文本
type= "password" 密码
type= "submit" 提交
type= "button" 只是一个点击按钮
type= "radio" 单选框 label 双标签 选项内容
type= "checkbox" 复选框 label 双标签 选项内容
value= "提示 如 请输入密码"
placeholder= "也是提示"
select
option 选项栏
text area 下拉框
resize : none 禁止任意拖放
name= "user" 账号
name= "password" 密码

table{ border- collapse: collapse;} 让 border 合并
tr 加
colspan= "3" 横向合并

rowspan= "3" 纵向合并
align 左右对齐方式 middle (左) left center right
valign 上下对齐方式 top center bottom
cellspacing= "0" 清除外面的边距
cellpadding= "0" 清楚里面的边距

```
< table cellspacing= "0" cellpadding= "0">  
< thead> < tr> < th> 表头< / th> < / tr> < / thead>  
< tbody> < tr> < td> 表身< / td> < / tr> < / tbody>  
< tfoot> < tr> < td> 表底< / td> < / tr> < / tfoot>  
< / table>  
frameborder= "0" 取消边框  
scrolling= "no" 取消滚动条  
iframe src
```

```
< frameset>  
< frame />  
< / frameset>
```

letter- spacing : 具体数值 字母间距
word- spacing : 数值 单词间距
overflow : auto 溢出滚动
visibility : hidden 隐藏
charset 字符集
utf- 8 国际编码
gb2312/ gbk 中国编码
author 作者
copyright 版权
keywords 关键词
description 介绍

js
@media screen and (max- width: 900px){}; 屏幕自适应

媒体 屏幕 最大

push : 添加一个数组元素
document : 文档
pop : 删除最后一个数组元素
console : 控制台
shift : 删除第一个数组元素
string : 字符串
Concat 组合数组
undefined : 未定义
typeof : 关键字
join : 数组转化为指定分隔符
slice : 返回指定数组
length : 数组长度
Array : 数组
number : 数字
boolean : 布尔

alert : 提示框
parseInt : 转化为整型数值
continue : 继续
is NaN : 非数字
sort : 升序
break : 截断 跳出
reverse : 倒序
object : 对象类型
parseFloat : 转化为浮点值
index : 索引
floor : 向下取整
variable : 变量
ceil : 向上取整
round : 四舍五入
abs : 返回绝对值
eval : 解析字符串并运算
max : 比较最大值
return : 返回
min : 比较最小值
addEventListener : 事件名 函数
toFixed : 保留小数点
attachEvent : (on) 事件名
random : 随机数
element : 元素
eventName : 事件名
function : 函数
removeEventListener : 事件解除
detachEvent : 删除已注册事件
open () : 弹出一个新窗口
function : 函数 (方法)
showModalDialog : 弹出一个模式窗口
close () : 关闭当前窗口
Date : 日期对象
getFullYear () : 年份
getMonth () : 月份 (0 - 11)
getDate () : 几号
getDay () : 星期几 (周一至周六)
getHours () : 小时 (0 - 23)
getMinutes () : 分钟
getSeconds () : 秒数 (0 - 59)
getMilliseconds () : 毫秒数
getTime () : 从 1970 至今的毫秒数
Date.now () : 从 1970 至今的毫秒值
toLocaleDateString () : 输出本地化的日期
setTimeout : 延迟时间
toLocaleTimeString () : 输出本地化得时间
clearTimeout : 定时器对象
toLocaleString : 输出本地化的日期和时间
setInterval : 间隔时间

clearInterval：停止
window.status：浏览器状态栏信息
window.screen：屏幕信息
screen.width：屏幕宽度
screen.height：屏幕高度
screen.availwidth：去掉任务栏宽度
screen.avaiheight：去掉任务栏高度
history：浏览历史记录
location：地址栏对象
navigator：用于获取用户浏览器信息
appName：浏览器代码名的字符串表示
appName：返回浏览器的名称
appVersion：返回浏览器的平台和版本信息
platform：返回运行浏览器的操作系统平台
userAgent：返回由客户机发送服务器的
user-agent：头部的值
Exception：异常 意外情况 (try- - - catch)
获取元素对象方式:
getElementById：
getElementByName：
getElementByTagName：
getElementByClassName：
onclick 当点击的时候
onmouseover 鼠标移入
onmouseout 鼠标移出
onblur：失去焦点事件
onkeydown：键盘按下事件（前）
onkeypress：键盘按下事件（后）
onload：当页面加载完成之后触发
onchange; 内容改变事件
onkeydown; 键盘按下事件（前）
onkeyup; 键盘松开事件
onkeypress; 键盘按下事件（后）
onmouseout：鼠标移出事件
onmouseover：鼠标移入事件
onselect：内容被选中事件
ondblclick：鼠标双击事件
onfocus：获得焦点事件
onkeyup：键盘松开事件
复选框操作 checked=" "
checked 默认选中
false 不选中
true 选中
typeof 查看数据类型
object 对象
number 数字
string 字符串
function 函数
boolean 布尔值 true false

undefined 未定义

parseInt 整数

parseFloat 浮点数/ 小数

Number 数字

hide 隐藏

show 显示

skin 皮肤

code 代码

even 事件

array 数组

integer 整数

veruant 变量

common 公用

switch

case 情况

break 中断整个循环

continue 跳出本次循环

default 不履行/ 所有情况都不满足的时候走这个

null 没有/ 空对象

computed 计算后/ 生效的

eval (JS 可以读懂的代码：如 ' 1+1' -> 字符串) 这个代码会把 () 里面的字符串换成计算机读懂的代码来计算 + - * % 都行 (计算用的)

return 返回

write 输入

find 查找

interval 间隔

setInterval (函数名 , 1000/) 间隔定时器

clearInterval 清除间隔定时器

Timeout 超时

setTimeout 延迟定时器

clearTimeout 清除延迟定时器

obj.disabled 让按钮/ 变成不可使用

true 不可用

false 可用

date 时间

get 获取

set 设置

Fullyear 满年

month 月 (从 0 月开始)

day 日 获取需要 date

week 星期 获取需要 day (从 0 到 6)

hours 小时

minutes 分钟

seconds 秒

single 一个/ 一倍

dubble 两个/ 两倍

triple 三个/ 三倍

arguments 函数传参- 实参的集合
arguments.length 函数传参- 实参的长度
arr.push() 向数组最后添加一项
arr.unshift() 向数组前面添加一项
arr.pop() 删除数组最后一位
arr.shift() 删除数组第一位
arr.splice(开始的位置, 删除的个数, 添加的东西 1。。。)
arr.join('') 变字符串 是中间的隔开东西 如- , 空格 等等。。。.
arr.concat(arr1, arr2, arr... 要拼接的数组...); 数组拼接/ 合并数组
arr.reverse() 反转/ 颠倒数组
arr.sort() 数组排序 按照字符串比大小的方法来排序
arr.sort(function(a1, a2){ return a1- a2 // 从小到大 return a2- a1 // 从大到小 }) a1 a2 随便起
代表的是数组中随机的某两项
str.charAt() 字符串里面的下标
str.indexOf(' ') 下字符串对应的下标数 (相同的就取第一个)
str.lastIndexOf(' ') 下字符串对应的下标数 (相同的就取最后一个)
str.substring(开始位置, 结束位置)/ (开始位置- 不写结束位置默认就从开始位置到最后) 截取子串
str.toUpperCase() 大写字母
str.toLowerCase() 小写字母
str.split(切割的方式 ' ' ' ' ' ' 空格' 等) 切割变成数组
str.charCodeAt('a') 输入字符查找对应的编码
String.fromCharCode(编码) 输入编码查找对应的字符
json.name= value; 添加
delete json.name 删除
token 括号
try catch(e) 异常
Math.random() 求随机数
parseInt (Math.random() (m- n) + n) 求 n- m之间随机的整数, 前包括后不包括
Math.abs() 求绝对值
Math.max(, , ,) 求最大值
Math.min(, , ,) 求最小值
Math.pow(n, m) 求 n 的 m 次方
Math.sqrt(a) 开方
Math.PI π
Math.ceil(12.3) 向上取整 13
Math.floor(12.3) 向下取整 12
Math.round(a) 四舍五入
!important Css 样式值后面加这个 权重最大
navigator.userAgent UA 例子判断浏览器的类型
navigator.language 语言
navigator.platform 操作系统

Sibling 兄弟
next 下一个
previous 上一个
获取元素/ 节点:
(父级) obj.children 找: 儿子/ 子级 子节点
(子级) obj.parentNode 找: 父亲/ 父级 父节点
(同级) obj.nextElementSibling || obj.nextSibling 下一个同级 / 节点 (同级)

obj.previousElementSibling || obj.previousSibling 上一个同级/节点

(父级) obj.firstElementChild || obj.firstChild 首节点

(父级) obj.lastElementChild || obj.lastChild 尾节点

创建元素：

document.createElement('标签名') 创建元素

添加插入

(父级) obj.appendChild(谁) 往元素里面最后添加一个

带剪切效果

(父级) obj.insertBefore(把谁, 加在谁的前面) 往元素里面添加一个

带剪切效果

删除元素：

(父级) obj.removeChild(子级) 删除子级

window.open() 打开

window.close() 关闭

a 标签: target = ""

不填 默认新页面打开

_blank 新页面打开

_self 本页面打开

window.location 地址栏信息

window.location.href 链接信息

window.location.search 数据信息

window.location.hash 锚点信息

*了解就行 面试背住

window.location.protocol 协议信息

window.location.hostname 域名信息

window.location.port 端口号信息

window.location.pathname 路径信息

back 返回

forward 向前

history 历史

window.history.back 返回历史记录 退

window.history.forward 返回历史记录 前

alert() 弹

confirm 询问提示框

prompt 问答提示框

console.log() 在 F12 中显示;

document.write(*) 在页面中输入

scroll 滚动 有兼容问题

client 客户端/可视区

offsetHeight/Width 物体盒子模型的宽高

offsetTop/Bottom/方向 物体距上/下/左/右的距离

offsetParent 定位父级

window.onScroll 当滚动的时候 滚动事件

window.onresize 当缩放的时候 缩放事件

onfocus 当获取焦点的时候

onblur 当失去焦点的时候

obj.getAttribute('属性名') 获取属性名对应的属性值

obj.setAttribute('属性名', '属性值') 设置属性

obj.removeAttribute('属性名') 删除属性

event 事件

onmousemove 鼠标移动事件

onload 当图片加载完成时

onerror 当图片加载失败时

cancelBubble 取消冒泡

onkeydown 当按键按下时

onkeyup 当按键抬起时

oEvent.keyCode 键盘编码

oncontextmenu 当按右键的时候

return false 阻止浏览器右键菜单

cache 缓存

onmousedown 鼠标按下时

onmouseup 鼠标抬起时

onmousemove 鼠标移动事件

obj.setCapture() 开始捕获

obj.releaseCapture() 取消捕获

obj.cloneNode() 克隆复制一个节点

document / obj.addEventListener('不加 on 的事件名', 函数名, false) 添加事件绑定 兼容高级

document / obj.attachEvent('加 on 的事件名', 函数名) 添加事件绑定 兼容 IE 系列

document / obj.removeEventListener('不加 on 的事件名', 函数名, false) 解除事件绑定

document / obj.detachEvent('加 on 的事件名', 函数名) 解除事件绑定

onmouseenter 鼠标移入

onmouseleave 鼠标移出

onmousewheel 鼠标滚轮事件

DOMMouseScroll 鼠标滚动事件

oEvent.wheelDelta 非火狐 测试鼠标滚轮向上向下弹得数字

oEvent.detail 火狐 测试鼠标滚轮向上向下弹得数字

return false 阻止浏览器默认事件 但是在事件绑定中失效

oEvent.preventDefault 在事件绑定中用 阻止浏览器默认事件 (如果单独用只兼容高级浏览器)

oninput 当输入的时候 ie9- 没有此方法

onpropertychange 当属性改变的时候 IE 系列

oEvent.srcElement 事件源 非火狐

oEvent.target 事件源 高级浏览器

duration 期间

start 开始

count 次数

complete 完成

var bFlag= false/ ture 开关 自定义

easing 速度类型 :

linear 匀速

ease- in 加速

ease- out 减速

window.onload 页面加载完毕后执行 (代码、资源)

DOMReady 页面加载完毕后执行 (代码)
DOMContentLoaded DOM加载事件
obj.onreadystatechange 监控事件 (模拟 DOM加载事件)
readyState 加载状态
document / obj.readyState 加载状态
ondblclick 鼠标双击
querySelector 新的获取元素方法

window.location.reload 重新加载

cookie
expires 期限
session 一个会话 (浏览器打开和关闭)
path 路径
define 定义
require 引用其他模块
exports 导出模块
module 批量导出模块 ,
data-main 主要数据 / 初始化接口

success 成功
error 失败

Request 请求
ActiveX 浏览器插件
XMLHttpRequest () ajax 创建 / 不兼容 ie6-
ActiveXObject ('Microsoft.XMLHTTP') 创建 / (插件) ie 系列

oAjax.open() 打开
oAjax.setRequestHeader (' content - type ' , ' application / x - www - form - urlencoded ');
oAjax.send() 发送
onreadystatechange 接收 / 当通信状态变化

oAjax.abort() 使中止
oAjax.readyState 是否完成 / 加载状态
oAjax.status 是否成功
response 回答 / 响应
oAjax.responseText 返回结果

JSON.parse() 可以把字符串 json 变成 json 不过 json 要正规些 双引号
JSON.stringify() json 打印出来的方法

encode 编码
decode 解码
Component 组件 / 元件
encodeURIComponent (' 中文 ') 中文转化成编码
decodeURIComponent (' 编码 ') 编码转化成中文

str.replace 替换
new RegExp() 正则表达式
str.search() 跟 indexOf 一样 在正则用

str.match 匹配

正则.test (你要校验的 str) true 正确满足 false 不满足 用这个的时候一定配合行首和行尾转译

\d 数字 [0- 9]

\w 数字、英文、下划线 [0- 9a- z]

\s 空白字符 所有的空白字符

. 代表任意字符

\D 非数字 [^0- 9]

\W 非数字、英文、下划线 [^0- 9a- z]

\S 非空白字符

\u4e00- \u9fa5 中文

量词

{n} n 个

{n, m} n- m 个

{n, } 最少 n 个 , 最多随便

{0, m} 最少随便 最多 m 个

{1, } 就是 若干个

{0, } 任意个 可以没有

? {0, 1} 最少没有 最多 1 个

\ 转译特殊字符

| 或的意思

i 忽略大小写

g 全局匹配

m 多行模式

^ 行首 如果在方括号里面是除了

\$ 行尾

*jq

hide 隐藏

show 显示

toggle 点击

hover 移入

fadeIn 淡入

fadeOut 淡出

animate 运动函数 (move)

slideDown 滑动从上往下出来

slideUp 滑动从下往上隐藏

eq(下标) 标签获取的下标

li> a li 下面的第一级 a

伪类选择器:

li: last li 最后一个

li: first li 第一个

li: eq(2) 第三个 li

obj : even 偶数

obj : odd 奇数

obj : has("p/ 标签名") 里面包含' p' 这个标签的就会选取

obj : contains("p/ 内容") 内容含有得' p' 就都会选取

event / 自己起行参名字.pageX X 轴坐标
event / 自己起行参名字.pageY Y 轴坐标
offset().top 物体绝对的位置 -> getPos()函数
offset().left 物体绝对的位置
position().top 物体距定位父级距离 -> offsetTop
position().left 物体距定位父级距离
document.scrollTop 和原生一样
document.scrollLeft 和原生一样
.width / .height 物体本身的宽高
innerHeight / Width 物体内部的宽高 包括 padding 不包括边框
outerHeight / Width 物体盒子模型宽高
父级.append(子级) 在父级内的后部追加
父级.prepend(子级) 在父级内的前面追加
子级.appendTo(父级) 在父级后部追加
子级.prependTo(父级) 在父级前部追加
obj1.after(obj2) 把 2 放在 1 的后面
obj1.before(obj2) 把 2 放在 1 的前面
bind() 事件绑定
unbind() 事件解绑
live() 事件委托
die() 解除事件委托
**以上四种不推荐 因为原理还是昂 所以推荐 on
obj.on() 可以事件绑定/ 委托

each 循环
\$.trim() 去除首尾空格

extend() 延伸
height() 就是宽度
innerHeight 包括 padding、不包括边框
outerHeight 盒子模型的
offset().left -> getPos()
position().left -> offsetLeft()
scrollTop() -> scrollTop
eve.pageX / Y 坐标
maxlength 表单元素最大长度 属性
\$.fn.函数名= function(){} 建立 jq 封装函数
\$.fn.extend({函数名 1: function(){}, 函数名 2: function(){}, ...})

汉字 : [0x4e00, 0x9fa5] (或十进制[19968, 40869])
数字 : [0x30, 0x39] (或十进制[48, 57])
小写字母 : [0x61, 0x7a] (或十进制[97, 122])
大写字母 : [0x41, 0x5a] (或十进制[65, 90])
parent() 父级
offsetParent() 定位父级
parents() 所有父级 括号里面可以筛选 比如放 class 名字或...
str / arr.slice (开始位置 , 结束位置) 切 从开始到结束 如果只有开始 就从开始到最后。 前包后不包

prototype 原型

form 表单中 submit 提交 reset 重置
xxx instanceof xxx 检测一个对象属于某个类
xxx.constructor == xxx 看构造函数 双等是判断 直接父级属不属于 xxx 一个等号是赋值让直接父级等于 xxx
fn.call(this 的指向, 参数一, 参数二,) 改变 this 的指向 fn 中的 this=
fn.apply(this 的指向, [参数一, 参数二,]) 改变 this 的指向 fn 中的 this=
init 开始

localStorage 本地存储
一般的规范
localStorage.name = value 存
localStorage.name 取
delete localStorage.name 删除一条
localStorage.clear() 删除全部
规范的存取
localStorage.setItem(name, value)
localStorage.getItem(name)
localStorage.removeItem(name)
sessionStorage 临时存储 和 localStorage 一样用法
include() 包含
template.default.openTag = ''; 自定义模板开始样子
template.default.closeTag = ''; 自定义模板结束样子
template.config('openTag', '')
template.config('closeTag', '')
template.config('escape', true/false) true 不默认转译 false 默认转译

m model 模型
v view 视图
c controller 控制器
scope 范围 作用域
then 然后
model 模型
bind 绑定
ng-app = ""; 引用其他模块
ng-model = "" 数据从哪来
ng-bind = "" 或 model 或 {{xxx}} 数据到哪去
ng-init 初始化
ng-repeat 循环
angular.module('名字', [依赖模块]).controller('名字', function(依赖项){})
\$scope
\$http
\$interval
\$timeout
` 反单引号 包住字符串 拼接是这样 \${}

|currency: "" 货币
date: "" 时间戳转化处理
route 路由/ 状态
filter 过滤器
directive 指令
restrict 类型 E: 'element' C: 'class' M: 'comment' A: 'attribute'

service 依赖
oDate.toString() 获取正常时间
creator.ionic.io 做手机端的框架网站不错

ng-show true false 显示 隐藏
ng-hide true false 隐藏 显示
ng-click
comment 注释
bundle 打包
transclude 嵌入
params 参数
provider 提供者
.config() 配置
\$scope.\$watch(数据, fn)
template 模板
templateUrl: '模板地址' 模板
\$bower install * 下载东西用的
ReactDOM.render(什么东西, 渲染到哪)
type= 'text/ babel'
state 状态
extends 延伸 继承
constructor 创建
super 超类/ 父类
component 组件
React.Component
this.props.属性
组件中 ref 和正常的 ID 类似

vue
el: 'id'
data: {}
\$index
capitalize 首字母大写
uppercase 变大写字母
lowercase 变小写字母
ev.stopPropagation() 阻止冒泡
ev.preventDefault() 阻止默认事件 angular
angular.bind(this 的指向, 要改变指向的函数名, 参数)
bootstrap 开启应用模式
copy(复制谁, 复制到哪里) 复制/ 克隆
equals(a, b) 比较
forEach 循环
fromJson 字符串 json 转化成 json
extend(obj, obj1, obj2。。。) 扩展

标签里面属性 transition= "名字" 过渡 动画
.名字-
transition 过渡
transition: 时间 让谁变/ all ease;
enter 进来

leave 出去
route 路由/ 状态
watch 监控
props 存东西 可在替换模板身上用
Router- view 状态视图
v- link= "{path: 名字
animation: 8s test/ 名字 linear infinite/ 数值/ 次数;
background- size 背景尺寸
contain 以最合适的状态展示 可能会留白
cover 尽可能填满
nth- child(下标 注意：从 1 开始) 注意: 项目不用这个。禁用
nth- of- type(下标 注意：从 1 开始) 注意: 项目不用这个。禁用
background- origin： 背景从哪里开始
content- box
border- box
padding- box
background- clip： 背景从哪里开切
content- box
border- box
padding- box
text 注意：这个 background- clip 前面要加上 webkit-