注意： “字符串”才加引号，变量不需要加

html/css

* 在 <style> 部分中 class 声明的顺序却非常重要，后面的声明总是比第一个具有优先权
* class属性中类名的顺序不会产生影响
* 选中div内的id：#abc bcd
* background padding border margin font等均有一行写完的方法
* span 创建行内元素,切分网页的元素<p>Top 3 things cats <span class = "text-danger">hate:</span></p>
* 内联元素会给浮动元素让路，普通块元素会在浮动元素的下方。(通常里面的内容都会围绕浮动元素）
* clear 用来指定一个块元素的左右不能有浮动元素，通过下移避开元素。
* <table>表格中，<tr>表示一行，<th>表示某一列的表头，<td>表示单元格 标题<caption>
* text-align属性定义行内内容（例如文字）如何相对它的块父元素对齐,并不控制块元素自己的对齐，只控制它的行内内容的对齐.
* 选择器权重计算共分为4个等级：

第一等：代表内联样式，如: style="xxx"，权值为1000。

第二等：代表ID选择器，如：#content，权值为100。

第三等：代表类，伪类和属性选择器，如.content，:hover，[attribute]，权值为10。

第四等：代表元素选择器和伪元素选择器，如div，p，权值为1。

注意：通用选择器（\*），子选择器（>）和相邻同胞选择器（+）并不在这四个等级中，所以他们的权值都为0。

* [弹性布局](https://juejin.im/post/5ac2329b6fb9a028bf057caf)
* CSS3动画

jquery

* id前缀为#，class前缀为. !important＞行内＞id＞class
* 元素选择器：$("button")、class选择器：$(".btn")、id选择器：$("#target1")
* parent()允许你访问指定元素的父元素。children()子元素。$("#left-well").parent().css("background-color", "blue")
* clone()方法可以拷贝元素。$("#target2").clone().appendTo("#right-well");
* appendTo()方法可以把选中的元素移动到其他元素中
* remove() 的方法可以移除HTML元素
* .html()方法可以添加HTML标签和文字到元素，而元素之前的内容都会被方法的内容所替换掉。
* .text()只能改变文本但不能修改标记
* .prop()调整元素的属性。$("button").prop("disabled", true);
* addClass()给元素添加class,removeClass()去掉元素上的class
* target:nth-child(n) 允许你按照索引顺序(从1开始)选择目标元素的所有子元素。$(".target:nth-child(3)").addClass("animated bounce");
* .target:even选取索引偶数的元素，odd奇数

js

* 在 JavaScript 中，你可以通过在引号前面使用 反斜杠 (\) 来转义引号。
* ||或，均为false才返回false，否则为true。1、只要“||”前面为false,不管“||”后面是true还是false，都返回“||”后面的值。2、只要“||”前面为true,不管“||”后面是true还是false，都返回“||”前面的值。
* &&与，均为true才返回true，否则为false。1、只要“&&”前面是false，无论“&&”后面是true还是false，结果都将返“&&”前面的值。2、只要“&&”前面是true，无论“&&”后面是true还是false，结果都将返“&&”后面的值;
* “&&”的优先级高于“||”。
* 只要你在开始和结束都使用相同类型的引号，单引号和双引号的功能在JavaScript中是相同的。
* .pop() 函数用来“抛出”一个数组末尾的值。我们可以把这个“抛出”的值赋给一个变量存储起来。 抛出后，将会从数组中移除。
* .shift()~与.pop()相反，移除的是第一个元素 push()为添加到结尾
* 严格相等运算符（===）是相对于相等操作符（==）的一种操作符。与相等操作符不同的是，它会同时比较元素的值和 数据类型。
* 注意： =是赋值符号
* 当且仅当运算符的左边和右边都是 true，逻辑与 运算符（&&）才会返回 true
* 如果任何一个操作数是true，逻辑或 运算符 (||) 返回 true。反之，返回 false。
* 嵌套json对象的访问方法例：myStorage.car.inside["glove box"]



* for (*statement 1*;*statement 2*;*statement 3*) {  
    // *code block to be executed*  
  }

语句 1 在循环（代码块）开始前执行

语句 2 定义运行循环（代码块）的条件

语句 3 在循环（代码块）已被执行之后执行

* 用.hasOwnProperty(propname)方法来检查对象是否有该属性。如果有返回true，反之返回 false。
* 生成指定范围内的随机数Math.floor(Math.random() \* (max - min + 1)) + min
* 构造函数使用this.语句间用分号； 普通函数不使用this.语句间用逗号，
* map 方法会迭代数组中的每一个元素，并根据回调函数来处理每一个元素，最后返回一个新数组。注意，这个方法不会改变原始数组。
* 数组方法 reduce 用来迭代一个数组，并且把它累积到一个值中。
* filter ，并且按给出的条件过滤出符合的元素。
* 使用 sort() 方法，你可以很方法用来迭代一个数组容易的按字母顺序或数字顺序对数组中的元素进行排序。sort 方法将改变原数组，返回被排序后的数组（不会生成新的数组）。
* 使用 reverse 方法来翻转数组
* Concat() 方法可以用来把两个数组的内容合并到一个数组中。参数应该是一个数组。参数中的数组会拼接在原数组的后面，并作为一个新数组返回。

注意不是contact！！！！

* 用 split() 方法按指定分隔符将字符串分割为数组。
* join() 方法将一个数组（或一个类数组对象）的所有元素连接成一个字符串并返回这个字符串。不会改变数组
* /i (忽略大小写)

/g (全文查找出现的所有匹配字符)

/m (多行查找)

/gi(全文查找、忽略大小写)

/ig(全文查找、忽略大小写)

* [indexOf()](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/String/indexOf) 方法返回调用String对象中第一次出现的指定值的索引，开始在 fromIndex进行搜索。未找到则返回-1。
* [substring()](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/String/substring) 方法返回一个字符串在开始索引到结束索引之间的一个子集, 或从开始索引直到字符串的末尾的一个子集。括号内为开始/结束位置。Substr()为开始位置/长度。
* .style.display.block 直接添加style语句display=“block”
* 如果使用 === 判断2个对象类型的数据，则直接判断这 2 个对象是否为同一个对象。
* 预解析过程中如果函数变量名相同，函数优先

VUE

* v-cloak方法解决加载慢时插值表达式闪烁问题
* v-text会以纯文本格式覆盖元素中原本的内容，插值表达式不会清空原本的整个元素，v-html以html格式输出内容，也会覆盖原内容
* v-bind: 用来绑定属性data，可以简写为 : ，在v-bind中可以输入合法的JS代码
* v-on：用来绑定事件，简写为@
* 前缀 $，与用户定义的属性区分开，如vm.$data
* 使用 DOM 中的模板时，camelCase (驼峰命名法) 的 prop 名需要使用其等价的 kebab-case (短横线分隔命名) 命名，如props: ['postTitle'](js)→post-title(html)
* components声明有哪些组件，template是使用哪个组件
* 全局组件Vue.component 局部组件new Vue中components: { ToDo : ToDo }