

# 2018 BAT 最新《前端必考面试题》

## 1、Doctype 作用？严格模式与混杂模式如何区分？它们有何意义？

- (1)、声明位于文档中的最前面，处于 标签之前。告知浏览器的解析器，用什么文档类型 规范来解析这个文档。
- (2)、严格模式的排版和 JS 运作模式是 以该浏览器支持的最高标准运行。
- (3)、在混杂模式中，页面以宽松的向后兼容的方式显示。模拟老式浏览器的行为以防止站点无法工作。
- (4)、DOCTYPE 不存在或格式不正确会导致文档以混杂模式呈现。

## 2、行内元素有哪些？块级元素有哪些？

- (1) CSS 规范规定，每个元素都有 display 属性，确定该元素的类型，每个元素都有默认的 display 值，比如 div 默认 display 属性值为 “block” ,成为 “块级” 元素； span 默认 display 属性值为 “inline” ，是 “行内” 元素。
- (2) 行内元素有：a b span img input select strong ( 强调的语气 ) 块级元素有：div ul ol li dl dt dd h1 h2 h3 h4...p

## 3、link 和@import 的区别是？

- (1) link 属于 XHTML 标签，而@import 是 CSS 提供的;
- (2) 页面被加载的时，link 会同时被加载，而@import 引用的 CSS 会等到页面被加载完再加载;
- (3) import 只在 IE5 以上才能识别，而 link 是 XHTML 标签，无兼容问题;
- (4) link 方式的样式的权重 高于@import 的权重.

## 4、浏览器的内核分别是什么？

IE 浏览器的内核 Trident、Mozilla 的 Gecko、Chrome 的 Blink( WebKit 的分支 )、Opera 内核原为 Presto，现为 Blink；

## 5、HTML5 有哪些新特性？如何处理 HTML5 新标签的浏览器兼容问题？如何区

## 分 HTML 和 HTML5 ?

HTML5 现在已经不是 SGML 的子集，主要是关于图像，位置，存储，多任务等功能的增加。

绘画 canvas 用于媒介回放的 video 和 audio 元素 本地离线存储 localStorage 长期存储数据，浏览器关闭后数据不丢失； sessionStorage 的数据在浏览器关闭后自动删除 语义化更好的内容元素，比如 article、footer、header、nav、section 表单控件 ,calendar、date、time、email、url、search 新的技术 webworker, websocket, Geolocation

获取更多资料欢迎加群：866109386



## 6、对语义化如何理解？

用正确的标签做正确的事情！

HTML 语义化就是让页面的内容结构化，便于对浏览器、搜索引擎解析；在没有样式 CCS 情况下也以一种文档格式显示，并且是容易阅读的。搜索引擎的爬虫依赖于标记来确定上下文和各个关键字的权重，利于 SEO。使阅读源代码的人对网站更容易将网站分块，便于阅读维护理解。

## 7、HTML5 的离线储存有几种方式？

localStorage 长期存储数据，浏览器关闭后数据不丢失；sessionStorage 数据在浏览器关闭后自动删除。

## 8、iframe 有那些缺点？

iframe 会阻塞主页面的 Onload 事件；

iframe 和主页面共享连接池，而浏览器对相同域的连接有限制，所以会影响页面的并行加载。使用 iframe 之前需要考虑这两个缺点。如果需要使用 iframe，最好是通过 javascript 动态给 iframe 添加 src 属性值，这样可以可以绕开以上两个问题。

## 9、请描述一下 cookies , sessionStorage 和 localStorage 的区别？

cookie 在浏览器和服务器间来回传递。 sessionStorage 和 localStorage 不会  
sessionStorage 和 localStorage 的存储空间更大；sessionStorage 和  
localStorage 有更多丰富易用的接口；sessionStorage 和 localStorage 各自独立的存储空间；

## 10、CSS 选择符有哪些？哪些属性可以继承？优先级算法如何计算？ CSS3 新增伪类有那些？

1.id 选择器 ( # myid )

2.类选择器 ( .myclassname )

3.标签选择器 ( div, h1, p )

4.相邻选择器 ( h1 + p )

5.子选择器 ( ul < li )

6.后代选择器 ( li a )

7.通配符选择器 ( \* )

8.属性选择器 ( a[rel = "external"] )

可继承的样式： font-size font-family color, UL LI DL DD DT;

不可继承的样式： border padding margin width height ;

优先级就近原则，同权重情况下样式定义最近者为准;

载入样式以最后载入的定位为准;

优先级为:

!important > id > class > tag    important 比 内联优先级高

## 11、CSS3 新增伪类举例：

p:first-of-type 选择属于其父元素的首个元素的每个元素。

p:last-of-type 选择属于其父元素的最后元素的每个元素。

p:only-of-type 选择属于其父元素唯一的元素的每个元素。

p:only-child 选择属于其父元素的唯一子元素的每个元素。

p:nth-child(2) 选择属于其父元素的第二个子元素的每个元素。

:enabled :disabled 控制表单控件的禁用态。

:checked 单选框或复选框被选中。

## 12、如何居中 div ? 如何居中一个浮动元素 ?

给 div 设置一个宽度，然后添加 margin:0 auto 属性

```
div{ width:200px; margin:0 auto; }
```

居中一个浮动元素

确定容器的宽高 宽 500 高 300 的层 设置层的外边距 .div { Width:500px ; height:300px;//高度可以不设 Margin: -150px 0 0 -250px; position:relative; 相对定位 background-color:pink;//方便看效果 left:50%; top:50%;}

列出 display 的值，说明他们的作用。position 的值，relative 和 absolute 定位原点是？

1.block 象块类型元素一样显示。 none 缺省值。象行内元素类型一样显示。 inline-block 象行内元素一样显示，但其内容象块类型元素一样显示。 list-item 象块类型元素一样显示，并添加样式列表标记。

2. absolute 生成绝对定位的元素，相对于 static 定位以外的第一个父元素进行定位。

fixed （老 IE 不支持） 生成绝对定位的元素，相对于浏览器窗口进行定位。

relative 生成相对定位的元素，相对于其正常位置进行定位。

static 默认值。没有定位，元素出现在正常的流中 \*( 忽略 top, bottom, left, right z-index 声明 )。

inherit 规定从父元素继承 position 属性的值。

### 13、为什么要初始化 CSS 样式？

因为浏览器的兼容问题，不同浏览器对有些标签的默认值是不同的，如果没对 CSS 初始化往往会出现浏览器之间的页面显示差异。当然，初始化样式会对 SEO 有一定的影响，但鱼和熊掌不可兼得，但力求影响最小的情况下初始化。最简单的初始化方法就是：`* {padding: 0; margin: 0;}`（不建议）淘宝的样式初始化：`body, h1, h2, h3, h4, h5, h6, hr, p, blockquote, dl, dt, dd, ul, ol, li, pre, form, fieldset, legend, button, input, textarea, th, td { margin:0; padding:0; }body, button, input, select, textarea { font:12px/1.5tahoma, arial, /5b8b/4f53; }h1, h2, h3, h4, h5, h6{ font-size:100%; }address, cite, dfn, em, var { font-style:normal; }code, kbd, pre, samp { font-family:couriernew, courier, monospace; }small{ font-size:12px; }ul, ol { list-style:none; }a { text-decoration:none; }a:hover { text-decoration:underline; }sup { vertical-align:text-top; }sub{ vertical-align:text-bottom; }legend { color:#000; }fieldset, img { border:0; }button, input, select, textarea { font-size:100%; }table { border-collapse:collapse; border-spacing:0; }`

### 14、css 定义的权重

以下是权重的规则：标签的权重为 1，class 的权重为 10，id 的权重为 100，以下例子是演示各种定义的权重值：`/*权重为 1*/div{}``/*权重为 10*/.class1{}``/*权重为 100*/#id1{}``/* 权重为 100+1=101*/#id1 div{}``/* 权重为 10+1=11*/.class1 div{}``/*权重为 10+10+1=21*/.class1 .class2 div{}` 如果权重相同，则最后定义的风格会起作用，但是应该避免这种情况出现

### 15、CSS3 有哪些新特性？

CSS3 实现圆角（`border-radius:8px`），阴影（`box-shadow:10px`），对文字加特效（`text-shadow`、），线性渐变（`gradient`），旋转

( transform ) transform:rotate(9deg) scale(0.85,0.90) translate(0px,-30px) skew(-9deg,0deg);//旋转,缩放,定位,倾斜 增加了更多的 CSS 选择器 多背景 rgba

## 16、介绍一下 CSS 的盒子模型？

( 1 ) 有两种，IE 盒子模型、标准 W3C 盒子模型；IE 的 content 部分包含了 border 和 padding;

( 2 ) 盒模型：内容(content)、填充(padding)、边界(margin)、边框(border).

## 17.对 WEB 标准以及 W3C 的理解与认识？

标签闭合、标签小写、不乱嵌套、提高搜索机器人搜索几率、使用外链 css 和 js 脚本、结构行为表现的分离、文件下载与页面速度更快、内容能被更多的用户所访问、内容能被更广泛的设备所访问、更少的代码和组件，容易维护、改版方便，不需要变动页面内容、提供打印版本而不需要复制内容、提高网站易用性;

## 18.XHTML 和 HTML 有什么区别？

HTML 是一种基本的 WEB 网页设计语言，XHTML 是一个基于 XML 的置标语言最主要的不同：

XHTML 元素必须被正确地嵌套。

XHTML 元素必须被关闭。标签名必须用小写字母。

XHTML 文档必须拥有根元素。

## 19.Doctype? 严格模式与混杂模式-如何触发这两种模式，区分它们有何意义？

用于声明文档使用那种规范(HTML/XHTML)一般为 严格 过度 基于框架的 html 文档

加入 XML 声明可触发，解析方式更改为 IE5.5 拥有 IE5.5 的 bug

## 20.行内元素有哪些?块级元素有哪些?CSS 的盒模型?

块级元素：div p h1 h2 h3 h4 form ul

行内元素: a b br i span input select



## 25.你做的页面在哪些浏览器测试过?这些浏览器的内核分别是什么?

IE(IE 内核) 火狐(Gecko) 谷歌(webkit) opera(Presto)

## 26.写出几种 IE6 BUG 的解决方法

- 1.双边距 BUG float 引起的 使用 display
- 2.3 像素问题 使用 float 引起的 使用 display:inline -3px
- 3.超链接 hover 点击后失效 使用正确的书写顺序 link visited hover active
- 4.IE z-index 问题 给父级添加 position:relative
- 5.Png 透明 使用 js 代码 改
- 6.Min-height 最小高度 !Important 解决
- 7.select 在 ie6 下遮盖 使用 iframe 嵌套
- 8.为什么没有办法定义 1px 左右的宽度容器(IE6 默认的行高造成的, 使用 over:hidden,zoom:0.08 line-height:1px)
- 9.ie 6 不支持!important

## 27.img 标签上 title 与 alt 属性的区别是什么?

Alt 当图片不显示是 用文字代表。

Title 为该属性提供信息

## 28.描述 css reset 的作用和用途。

Reset 重置浏览器的 css 默认属性 浏览器的品种不同, 样式不同, 然后重置, 让他们统一

## 29.解释 css sprites , 如何使用。

Css 精灵 把一堆小的图片整合到一张大的图片上, 减轻服务器对图片的请求数量

## 30.浏览器标准模式和怪异模式之间的区别是什么?



盒子模型 渲染模式的不同

使用 `window.top.document.compatMode` 可显示为什么模式

### 31.你如何对网站的文件和资源进行优化?期待的解决方案包括：

文件合并

文件最小化/文件压缩

使用 CDN 托管

缓存的使用

### 32.什么是语义化的 HTML?

直观的认识标签 对于搜索引擎的抓取有好处

### 33.清除浮动的几种方式，各自的优缺点

- 1.使用空标签清除浮动 `clear:both`(理论上能清楚任何标签，增加无意义的标签)
- 2.使用 `overflow:auto`(空标签元素清除浮动而不得不增加无意义代码的弊端,,使用 `zoom:1` 用于兼容 IE)
- 3.是用 `afert` 伪元素清除浮动(用于非 IE 浏览器)

### 34.javascript 的 `typeof` 返回哪些数据类型

Object number function boolean underfind

### 35.例举 3 种强制类型转换和 2 种隐式类型转换?

强制 ( `parseInt,parseFloat,number` )

隐式 ( `== - ===` )

### 36.split() join() 的区别

前者是切割成数组的形式，后者是将数组转换成字符串

### 37.数组方法 `pop()` `push()` `unshift()` `shift()`

Push()尾部添加 pop()尾部删除  
Unshift()头部添加 shift()头部删除

### 38.事件绑定和普通事件有什么区别

### 39.IE 和 DOM 事件流的区别

- 1.执行顺序不一样、
- 2.参数不一样
- 3.事件加不加 on
- 4.this 指向问题

### 40.IE 和标准下有哪些兼容性的写法

```
Var ev = ev || window.event  
document.documentElement.clientWidth || document.body.clientWidth  
Var target = ev.srcElement||ev.target
```

### 41.ajax 请求的时候 get 和 post 方式的区别

一个在 url 后面 一个放在虚拟载体里面  
有大小限制  
安全问题  
应用不同 一个是论坛等只需要请求的，一个是类似修改密码的

### 42.call 和 apply 的区别

```
Object.call(this,obj1,obj2,obj3)  
Object.apply(this,arguments)
```

### 43.ajax 请求时，如何解释 json 数据

使用 eval parse 鉴于安全性考虑 使用 parse 更靠谱

#### 44.闭包是什么，有什么特性，对页面有什么影响

闭包就是能够读取其他函数内部变量的函数。

#### 45.如何阻止事件冒泡和默认事件

cancelBubble return false

#### 46.添加 删除 替换 插入到某个接点的方法

obj.appendChild()

obj.insertBefore

obj.replaceChild

obj.removeChild

#### 47.解释 jsonp 的原理，以及为什么不是真正的 ajax

动态创建 script 标签，回调函数

Ajax 是页面无刷新请求数据操作

#### 48.javascript 的本地对象，内置对象和宿主对象

本地对象为 array obj regexp 等可以 new 实例化

内置对象为 global Math 等不可以实例化的

宿主为浏览器自带的 document,window 等

#### 49.document load 和 document ready 的区别

Document.onload 是在结构和样式加载完才执行 js

Document.ready 原生没有这个方法，jquery 中有 \$.ready(function)

#### 50."==" 和 "===" 的不同

前者会自动转换类型

后者不会

## 51.javascript 的同源策略

一段脚本只能读取来自于同一样源的窗口和文档的属性，这里的同一样源指的是主机名、协议和端口号的组合

## 52.编写一个数组去重的方法

```
function oSort(arr)
{
  var result = {};
  var newArr=[];
  for(var i=0;i<arr.length;i++)
  {
    if(!result[arr])
    {
      newArr.push(arr)
      result[arr]=1
    }
  }
  return newArr
}
</arr.length;i++)
```

## 53.你认为最常遇到的兼容 Bug 有哪些?有哪些问题是你认为解决起来最麻烦的?

IE6 PNG

IE6 Fixed

## 54.CSS 定位方式有哪些?position 属性的值有哪些?他们之间的区别是什么?

在 CSS 中关于定位的内容是：position:relative | absolute | static | fixed  
static 没有特别的设定，遵循基本的定位规定，不能通过 z-index 进行层次分级。  
relative 不脱离文档流，参考自身静态位置通过 top,bottom,left,right 定位，并

且可以通过 z-index 进行层次分级。

absolute 脱离文档流,通过 top,bottom,left,right 定位。选 53D6 其最近的父级定位元素,当父级 position 为 static 时,absolute 元素将以 body 坐标原点进行定位,可以通过 z-index 进行层次分级。

fixed 固定定位,这里他所固定的对象是可视窗口而并非是 body 或是父级元素。可通过 z-index 进行层次分级。

### 55.函数的几种定义方法?

```
function a(){},  
var a = function(){}
```

### 56.对象的定义方法?

```
a = new Object(), a = {}
```

### 57.类的定义方法 ( prototype ) ( 继承 )

```
Var a = function(){}  
a.prototype = {}  
new a();
```

获取更多资料欢迎加群: 866109386

### 58.this 关键字的指向

```
obj.foo() == obj      //方法调用模式,this 指向 obj  
foo() == window;      //函数调用模式,this 指向 window  
new obj.foo() == obj   //构造器调用模式, this 指向新建立对象  
foo.call(obj) == obj;  //APPLY 调用模式,this 指向 obj
```

### 59.异步 ajax 的优缺点都有什么?

优点:

相对于同步 ajax: 不会造成 UI 卡死, 用户体验好。

相对于刷新页面, 省流量

缺点:

后退按钮无效；

多个请求同时触发时，由于回调时间不确定，会造成混乱，避免这种混乱需要复杂的判断机制。

搜索引擎不友好

数据安全获取更多资料欢迎加群：866109386

## 60、介绍 js 的基本数据类型。

number,string,boolean,object,undefined

## 61、Javascript 如何实现继承？

通过原型和构造器

## 62、如何创建一个对象？（画出此对象的内存图）

```
function Person(name, age) {  
    this.name = name;    this.age =  
age;    this.sing = function() { alert(this.name) } }  
}
```

## 63、谈谈 This 对象的理解。

this 是 js 的一个关键字，随着函数使用场合不同，this 的值会发生变化。但是有一个总原则，那就是 this 指的是调用函数的那个对象。this 一般情况下：是全局对象 Global。作为方法调用，那么 this 就是指这个对象

## 64、事件是什么？IE 与火狐的事件机制有什么区别？如何阻止冒泡？

（1）我们在网页中的某个操作（有的操作对应多个事件）。例如：当我们点击一个按钮就会产生一个事件。是可以被 JavaScript 侦测到的行为。

（2）事件处理机制：IE 是事件冒泡、火狐是 事件捕获；

（3）ev.stopPropagation();获取更多资料欢迎加群：866109386

## 65、如何判断一个对象是否属于某个类？

使用 instanceof （待完善） `if(a instanceof Person){ alert('yes'); }`

## 66、Javascript 中，有一个函数，执行时对象查找时，永远不会去查找原型，这个函数是？

`hasOwnProperty`

## 67、对 JSON 的了解？

JSON(JavaScript Object Notation) 是一种轻量级的数据交换格式。它是基于 JavaScript 的一个子集。数据格式简单，易于读写，占用带宽小 `{'age':'12', 'name':'back'}`

## 68、简述一下 src 与 href 的区别

`href` 是指向网络资源所在位置，建立和当前元素（锚点）或当前文档（链接）之间的链接，用于超链接。

`src` 是指向外部资源的位置，指向的内容将会嵌入到文档中当前标签所在位置；在请求 `src` 资源时会将其指向的资源下载并应用到文档内，例如 `js` 脚本，`img` 图片和 `frame` 等元素。当浏览器解析到该元素时，会暂停其他资源的下载和处理，直到将该资源加载、编译、执行完毕，图片和框架等元素也如此，类似于将所指向资源嵌入当前标签内。这也是为什么将 `js` 脚本放在底部而不是头部。

## 69、简述同步和异步的区别

同步是阻塞模式，异步是非阻塞模式。

同步就是指一个进程在执行某个请求的时候 若该请求需要一段时间才能返回信息，那么这个进程将会一直等待下去，直到收到返回信息才继续执行下去；

异步是指进程不需要一直等下去，而是继续执行下面的操作，不管其他进程的状态。当有消息返回时系统会通知进程进行处理，这样可以提高执行的效率。

## 70、px 和 em 的区别

`px` 和 `em` 都是长度单位，区别是，`px` 的值是固定的，指定是多少就是多少，计算比较容易。`em` 得值不是固定的，并且 `em` 会继承父级元素的字体大小。

浏览器的默认字体高都是 16px。所以未经调整的浏览器都符合: 1em=16px。那么 12px=0.75em, 10px=0.625em

## 71、什么叫优雅降级和渐进增强？

渐进增强 progressive enhancement：

针对低版本浏览器进行构建页面，保证最基本的功能，然后再针对高级浏览器进行效果、交互等改进和追加功能达到更好的用户体验。

优雅降级 graceful degradation：

一开始就构建完整的功能，然后再针对低版本浏览器进行兼容。

区别：

- a. 优雅降级是从复杂的现状开始，并试图减少用户体验的供给
- b. 渐进增强则是从一个非常基础的，能够起作用的版本开始，并不断扩充，以适应未来环境的需要
- c. 降级（功能衰减）意味着往回看；而渐进增强则意味着朝前看，同时保证其根基处于安全地带

## 72、浏览器的内核分别是什么？

IE: trident 内核

Firefox：gecko 内核

Safari：webkit 内核

Opera：以前是 presto 内核，Opera 现已改用 Google Chrome 的 Blink 内核

Chrome：Blink(基于 webkit，Google 与 Opera Software 共同开发)

## 73、如何消除一个数组里面重复的元素？

// 方法一：

```
var arr1 = [1,2,2,2,3,3,3,4,5,6],  
    arr2 = [];  
for(var i = 0, len = arr1.length; i < len; i++){  
    if(arr2.indexOf(arr1[i]) < 0){
```



```
        arr2.push(arr1[i]);
    }
}
document.write(arr2); // 1,2,3,4,5,6
```

#### 74、在 Javascript 中什么是伪数组？如何将伪数组转化为标准数组？

伪数组（类数组）：无法直接调用数组方法或期望 length 属性有什么特殊的行为，但仍可以对真正数组遍历方法来遍历它们。典型的是函数的 argument 参数，还有像调用 getElementByTagName, document.childNodes 之类的，它们都返回 NodeList 对象都属于伪数组。可以使用 Array.prototype.slice.call(fakeArray) 将数组转化为真正的 Array 对象。获取更多资料欢迎加群：866109386

```
function log(){
    var args = Array.prototype.slice.call(arguments);
    //为了使用 unshift 数组方法，将 argument 转化为真正的数组
    args.unshift('(app)');

    console.log.apply(console, args);
};
```

#### 75、Javascript 中 callee 和 caller 的作用？

caller 是返回一个对函数的引用，该函数调用了当前函数；  
callee 是返回正在被执行的 function 函数，也就是所指定的 function 对象的正文。

#### 76、请描述一下 cookies，sessionStorage 和 localStorage 的区别

sessionStorage 用于本地存储一个会话（session）中的数据，这些数据只有在同一个会话中的页面才能访问并且当会话结束后数据也随之销毁。因此 sessionStorage 不是一种持久化的本地存储，仅仅是会话级别的存储。而 localStorage 用于持久化的本地存储，除非主动删除数据，否则数据是永远不会过期的。

web storage 和 cookie 的区别

Web Storage 的概念和 cookie 相似，区别是它是为了更大容量存储设计的。Cookie 的大小是受限的，并且每次你请求一个新的页面的时候 Cookie 都会被发送过去，这样无形中浪费了带宽，另外 cookie 还需要指定作用域，不可以跨域调用。除此之外，Web Storage 拥有 setItem,getItem,removeItem,clear 等方法，不像 cookie 需要前端开发者自己封装 setCookie，getCookie。但是 Cookie 也是不可或缺的：Cookie 的作用是与服务器进行交互，作为 HTTP 规范的一部分而存在，而 Web Storage 仅仅是为了在本地“存储”数据而生。

## 77、手写数组快速排序

关于快排算法的详细说明，可以参考阮一峰老师的文章快速排序

“快速排序”的思想很简单，整个排序过程只需要三步：

- (1) 在数据集之中，选择一个元素作为“基准”（pivot）。
- (2) 所有小于“基准”的元素，都移到“基准”的左边；所有大于“基准”的元素，都移到“基准”的右边。
- (3) 对“基准”左边和右边的两个子集，不断重复第一步和第二步，直到所有子集只剩下一个元素为止。

## 78、统计字符串“aaaabbbccccdddfgh”中字母个数或统计最多字母数。

```
var str = "aaaabbbccccdddfgh";
var obj = {};
for(var i=0;i<str.length;i++){
    var v = str.charAt(i);
    if(obj[v] & obj[v].value == v){
        obj[v].count = ++ obj[v].count;
    }else{
        obj[v] = {};
        obj[v].count = 1;
        obj[v].value = v;
    }
}
```

```
for(key in obj){  
    document.write(obj[key].value + '=' + obj[key].count + ' ');  
    //  
    a=4 b=3 c=4 d=2 f=1 g=1 h=1  
}
```

## 79、一次完整的 HTTP 事务是怎样的一个过程？

基本流程：

- a. 域名解析
- b. 发起 TCP 的 3 次握手
- c. 建立 TCP 连接后发起 http 请求
- d. 服务器端响应 http 请求，浏览器得到 html 代码
- e. 浏览器解析 html 代码，并请求 html 代码中的资源
- f. 浏览器对页面进行渲染呈现给用户

## 80、对前端工程师这个职位你是怎么样理解的？

- a. 前端是最贴近用户的程序员，前端的能力就是能让产品从 90 分进化到 100 分，甚至更好
- b. 参与项目，快速高质量完成实现效果图，精确到 1px；
- c. 与团队成员，UI 设计，产品经理的沟通；
- d. 做好的页面结构，页面重构和用户体验；

e. 处理 hack , 兼容、写出优美的代码格式 ;

f. 针对服务器的优化、拥抱最新前端技术。

获取更多资料欢迎加群：866109386

