

前端面试题

前端技术 前端 面试

前端面试题

CSS

简要手绘盒子模型，并对边框和边距进行标注？

界面div三平分除了flex布局外，简写其它两种布局代码？

- 1、绝对定位法
- 2、margin负值法
- 3、自身浮动法

javascript

es6中let const var的区别是什么？

对数组[5,6,6,7,8,8]进行去重，es5或者es6方法

谈谈对es6中=>的理解

点击一个按钮，发出ajax请求，如何防止 用户在此请求方式返回之前再次点击？

编写一个双层回调函数？

封闭一个深度克隆方法es5或es6？

请用正则校验身份号码，要求符合15位或18位，末尾可能是X

html5&css3

html5和css3新增了哪些标签和属性？

用css3写一个开关按钮，按下时置灰变色，并改变按钮文字(如按下之前是on,按下之后是off)

http

http和https请求的区别是什么？post请求和get请求的区别是什么？

前端框架和模板

使用过的前端框架有哪些？

使用过的模板引擎有哪些？

使用过的模块加载器有哪些？

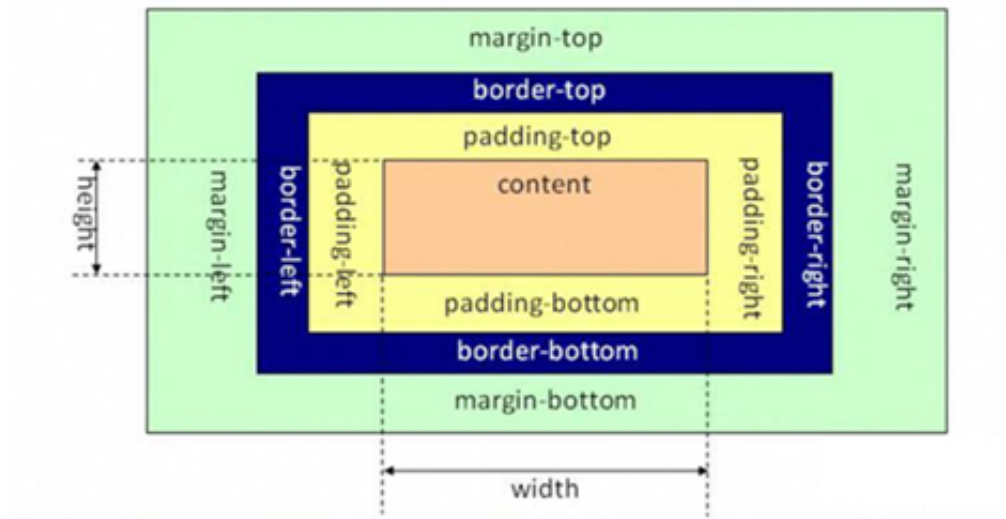
简写ReactNative中使用es5和es6如何绑定的this？

如何理解 前端MVC架构，简要画出流程图

CSS

简要手绘盒子模型，并对边框和边距进行标注？

■ 标准盒子模型



界面div三平分除了flex布局外，简写其它两种布局代码？

1、绝对定位法

这或许是三种方法里最直观，最容易理解的：左右两栏采用绝对定位，分别固定于页面的左右两侧，中间的主体栏用左右margin值撑开距离。于是实现了三栏自适应布局。

css代码如下（为截图）：

```
<style type="text/css">
html,body{margin:0; height:100%;}
#left,#right{position:absolute; top:0; width:200px; height:100%;}
#left{left:0; background:#a0b3d6;}
#right{right:0; background:#a0b3d6;}
#main{margin:0 20px; background:#ffe6b8; height:100%;}
</style>
```

HTML代码为（图片）：

```
<body>
<div id="left"></div>
<div id="main"></div>
<div id="right"></div>
</body>
```

2、margin负值法

这种方法是在实际的网站中应用的最多的，我个人感觉多少有些跟风的嫌疑。此方法很难用一句话概括。首先，中间的主体要使用双层标签。外层div宽度100%显示，并且浮动（本例左浮动，下面所述依次为基础），内层div为真正的主体内容，含有左右210像素的margin值。左栏与右栏都是采用margin负值定位的，左栏左浮动，margin-left为-100%，由于前面的div宽度100%与浏览器，所以这里的-100%margin值正好使左栏div定位到了页面的左侧；右侧栏也是左浮动，其margin-left也是负值，大小为其本身的宽度即200像素。

见下面的css代码：

```
<style type="text/css">
html,body{margin:0; height:100%;}
#main{width:100%; height:100%; float:left;}
#main #body{margin:0 210px; background:#ffe6b8; height:100%;}
#left,#right{width:200px; height:100%; float:left; background:#a0b3d6;}
#left{margin-left:-100%;}
#right{margin-left:-200px;}
</style>
```

HTML代码：

```
<body>
<div id="main">
    <div id="body"></div>
</div>
<div id="left"></div>
<div id="right"></div>
</body>
```

3、自身浮动法

此方法代码最简单。应用了标签浮动跟随的特性。左栏左浮动，右栏右浮动，主体直接放后面，就实现了自适应。

css代码如下：

```
<style type="text/css">
html,body{margin:0; height:100%;}
#main{height:100%; margin:0 210px; background:#ffe6b8;}
#left,#right{width:200px; height:100%; background:#a0b3d6;}
#left{float:left;}
#right{float:right;}
</style>
```

HTML代码：

```
<body>
<div id="left"></div>
<div id="right"></div>
<div id="main"></div>
</body>
```

javascript

es6中let const var的区别是什么？

- **var** : 声明全局变量
- **let** : 声明块级变量，即局部变量
- **const** : 用于声明常量，就是定义后不能再修改值或者引用值的常量，也具有块级作用域

对数组[5,6,6,7,8,8]进行去重，es5或者es6方法

ES5数组去重

```
Array.prototype.unique1 = function()
{
    var n = []; //一个新的临时数组
    for(var i = 0; i < this.length; i++) //遍历当前数组
    {
        //如果当前数组的第i已经保存进了临时数组，那么跳过，
        //否则把当前项push到临时数组里面
        if (n.indexOf(this[i]) == -1) n.push(this[i]);
    }
    return n;
}

Array.prototype.unique2 = function()
{
    var n = {}, r=[]; //n为hash表，r为临时数组
    for(var i = 0; i < this.length; i++) //遍历当前数组
    {
        if (!n[this[i]]) //如果hash表中没有当前项
        {
            n[this[i]] = true; //存入hash表
            r.push(this[i]); //把当前数组的当前项push到临时数组里面
        }
    }
    return r;
}

Array.prototype.unique3 = function()
{
    var n = [this[0]]; //结果数组
    for(var i = 1; i < this.length; i++) //从第二项开始遍历
    {
        //如果当前数组的第i项在当前数组中第一次出现的位置不是i，
```

```

        //那么表示第i项是重复的，忽略掉。否则存入结果数组
        if (this.indexOf(this[i]) == i) n.push(this[i]);
    }
    return n;
}
Array.prototype.unique4 = function()
{
    this.sort();
    var re=[this[0]];
    for(var i = 1; i < this.length; i++)
    {
        if( this[i] !== re[re.length-1])
        {
            re.push(this[i]);
        }
    }
    return re;
}

```

ES6去重

```

Array.prototype.remDub = Array.prototype.remDub || function () {
    //return Array.from(new Set(this));
    return [...new Set(this)];
};

```

谈谈对es6中=>的理解

箭头函数相当于匿名函数，并且简化了函数定义,箭头左边是参数，右边是返回值。

箭头函数看上去是匿名函数的一种简写，但实际上，箭头函数和匿名函数有个明显的区别：箭头函数内部的this是词法作用域，由上下文确定。

点击一个按钮，发出ajax请求，如何防止 用户在此请求方式返回之前再次点击？

1. 当点击提交按钮的时候，把这个提交这个处理函数给解绑掉，请求完成的时候在绑定回来

```
function clickHandler(){
    $(this).unbind('click', clickHandler);
    $.ajax({
        url : 'url',
        dataType : 'json',
        type : 'post',
        success : function (data) {
            if (data.success) {
                //提交成功做跳转处理
            } else {
                //处理失败，重新绑定点击事件
                alert(data.message);
                $(self).click(clickHandler);
            }
        }
    });
}
$('#buntton').click(clickHandler);
```

2.可以点击后让按钮不可用,如果提交失败可以再次设置为可用

```
$("#button1").attr("disabled","disabled");//按钮不可用
$.ajax({
    url : 'url',
    dataType : 'json',
    type : 'post',
    success : function (data) {
        if (data.success) {
            //提交成功做跳转处理
        } else {
            //处理失败，重新绑定点击事件
            alert(data.message);
            $("#button1").removeAttr("disabled");//按钮可用
        }
    }
});
```

编写一个双层回调函数？

封闭一个深度克隆方法es5或es6？

```
let json = {a:6,b:5,c:[1,2,3,4]};
```

```
function es5Clone(obj) {
    var str, newObj = obj.constructor === Array ? [] : {};
    if (typeof obj !== 'object') {
        return;
    } else if (window.JSON) {
        str = JSON.stringify(obj), //系列化对象
        newObj = JSON.parse(str); //还原
    } else {
        for (var i in obj) {
            newObj[i] = typeof obj[i] === 'object' ?
                es5Clone(obj[i]) : obj[i];
        }
    }
    return newObj;
};
```

请用正则校验身份号码，要求符合15位或18位，末尾可能是X

以黑龙江省的身份证为例

- 15位
 - 1-6位是你所在的地方
 - 7-12位是你的生日
 - 13-15位是你的身份证号
- 18位
 - 1-6位是你所在的地方
 - 7-14位是你的生日
 - 15-17位是你的身份证号
 - 18位是校验码

身份证号1 3 5 7 9是男性 2 4 6 8 0是女性
校验码不按男女算,有的还是X.
- 如:
 - 230102800808002(15位)
 - 23黑龙江省01哈尔滨市02道里区800808生日002身份证号
 - 230102198008080021
 - 23黑龙江省01哈尔滨市02道里区19800808生日002身份证号1校验码

正则表达式

```
//身份证正则表达式(15位)
isIDCard1=/^[1-9]\d{7}((0\d)|(1[0-2]))((0[1|2]\d)|3[0-1])\d{3}$/;
//身份证正则表达式(18位)
isIDCard2=/^[1-9]\d{5}[1-9]\d{3}((0\d)|(1[0-2]))((0[1|2]\d)|3[0-1])\d{3}([0-9]|X)$/;
```

html5&css3

html5和css3新增了些哪些标签和属性？

• HTML5语法

- 大部分延续了html的语法
- 不同之处：开头的

```
<html lang="zh-CN">
<meta charset="utf-8">
```

- 字符编码变得简洁，
- 不区分大小写，
- 添加了布尔值，类似checked，selected
- 引号可以省略，但是编码规范来说，不建议，
- 有可以省略结束符的标签，和完全省略的标签应用。

• 增加标签

- 1、结构标签
 - (1) section：独立内容区块，可以用h1~h6组成大纲，表示文档结构，也可以有章节、页眉、页脚或页眉的其他部分；
 - (2) article：特殊独立区块，表示这篇页眉中的核心内容；
 - (3) aside：标签内容之外与标签内容相关的辅助信息；
 - (4) header：某个区块的头部信息/标题；
 - (5) hgroup：头部信息/标题的补充内容；
 - (6) footer：底部信息；
 - (7) nav：导航条部分信息
 - (8) figure：独立的单元，例如某个有图片与内容的新闻块。
- 2、表单标签
 - (1) email：必须输入邮件；
 - (2) url：必须输入url地址；

- (3) number : 必须输入数值 ;
 - (4) range : 必须输入一定范围内的数值 ;
 - (5) Date Pickers : 日期选择器 ;
 - a.date : 选取日、月、年
 - b.month : 选取月、年
 - c.week : 选取周和年
 - d.time : 选取时间 (小时和分钟)
 - e.datetime : 选取时间、日、月、年 (UTC时间)
 - f.datetime-local : 选取时间、日、月、年 (本地时间)
 - (6) search : 搜索常规的文本域 ;
 - (7) color : 颜色
- 3、媒体标签
- (1) video : 视频
 - (2) audio : 音频
 - (3) embed : 嵌入内容 (包括各种媒体) , Midi、Wav、AU、MP3、Flash、AIFF等。
- 4、其他功能标签
- (1) mark : 标注 (像荧光笔做笔记)
 - (2) progress : 进度条 ;
 - (3) time : 数据标签 , 给搜索引擎使用 ; 发布日期9 : 00更新日期<time datetime="2015-01-23T04:00" pubdate>4:00
 - (5) wbr : 软换行 , 页面宽度到需要换行时换行 ;
 - (6) canvas : 使用JS代码做内容进行图像绘制 ;
 - (7) command : 按钮 ;
 - (8) details : 展开菜单 ;
 - (9) dateilst : 文本域下拉提示 ;
 - (10) keygen:加密 ;
- 5、新增的属性 :
- 对于js进行添加的属性。

```
<script defer src=".....js" onload="alert('a')"></script>
<script async src=".....js" onload="alert('b')"></script>
```

如果没有以上两个属性的话，执行顺序为先加载（下载）第一个src，然后在执行其onload，然后在向下依次同步执行defer属性在h5之前就已经有了，输入延迟加载（推迟执行），它会先加载（下载）src中文件内容，然后等页面全部加载完成后，再加载onload中js.async属性属于异步加载，它会在加载src后，立即执行onload,同时还会继续加载页面上执行顺序，alert显示会先显示b然后再显示a

- 网页中标签中加入小图标的样式代码

```
<link rel="icon" href="url..." type="图片名称" size="16*16">
```

有序列表ol:新增start（列表起始值），reversed（是否倒置）
menu菜单type属性（3个菜单类型）内嵌css样式：在标签内部来定义一个样式区块（scoped），只对样式标签内部才有效内嵌框架：iframe元素，新增了seamless无边距无边框，srcdoc定义了内嵌框架的内容

用css3写一个开关按钮，按下时置灰变色，并改变按钮文字(如按下之前是on,按下之后是off)

<https://github.com/justforuse/HTML-CSS-JS/blob/master/customize-radio/index.html>

http

http和https请求的区别是什么？post请求和get请求的区别是什么？

- http是超文本传输协议，信息是明文传输，https 则是具有安全性的ssl加密传输协议
- http和https使用的是完全不同的连接方式用的端口也不一样,前者是80,后者是443。
- http的连接很简单,是无状态的，HTTPS协议是由SSL+HTTP协议构建的可进行加密传输、身份认证的网络协议，要比http协议安全

前端框架和模板

使用过的前端框架有哪些？

Angular、Vue、React、Backbone、Ember、Meteor

使用过的模板引擎有哪些？

ejs、handlebar、jquery template、ejs、arttemplate

使用过的模块加载器有哪些？

requirejs、sea.js

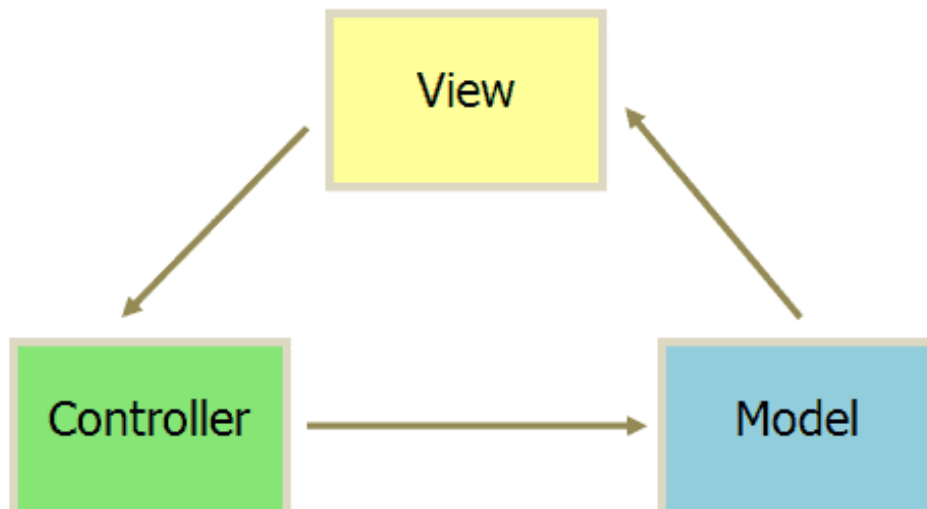
简写ReactNative中使用es5和es6如何绑定的this？

1. 箭头函数
2. 外层缓存this变量
3. 使用bind

如何理解 前端MVC架构，简要画出流程图

MVC模式的意思是，软件可以分成三个部分：

- 视图（View）：用户界面。
- 控制器（Controller）：业务逻辑
- 模型（Model）：数据保存



- View 传送指令到 Controller
- Controller 完成业务逻辑后，要求 Model 改变状态
- Model 将新的数据发送到 View，用户得到反馈