**前端面试题**

**HTML**

1. 请描述一下 cookies、sessionstorage 和 localstorage 的区别 ？

* cookie 有大小限制，一般来说是4k。随着http请求一起发送至服务器端，可以设置过期时间，不设置的情况下在浏览器关闭的时候会清空
* sessionstorage可以保存较多数据，在浏览器关闭的时候会被清空，同时浏览主域名下不同子域名内容不会共享
* localstorage可可以保存较多数据，可以永久保存数据除非用户主动删除或清除缓存

2. 请列举几个项目中常见兼容性问题？

* png24位的图片在ie6浏览器上出现背景，解决方案是做成png8.也可以引用一段脚本处理.
* 浏览器默认的margin和padding不同。解决方案是加一个全局的\*{margin:0;padding:0;}来统一。
* ie6双边距bug:块属性标签float后，又有横行的margin情况下，在ie6显示margin比设置的大。
* 浮动ie产生的双倍距离（ie6双边距问题：在ie6下，如果对元素设置了浮动，同时又设置了margin-left或margin-right，margin值会加倍。）

3. 常见的http状态码

* 404 没找到页面(not found)
* 403 禁止访问(forbidden)
* 500 内部服务器出错(internal service error)
* 200 一切正常(ok)
* 304 没有被修改(not modified)(服务器返回304状态，表示源文件没有被修改）

4. 请说出三种减少页面加载时间的方法。（加载时间指感知的时间或者实际加载时间）

* 1.优化图片
* 2.图像格式的选择（gif：提供的颜色较少，可用在一些对颜色要求不高的地方）
* 3.优化css（压缩合并css，如margin-top,margin-left...)

5. html5中新加入的表单控件

* email 电子邮箱文本框
* tel 电话号码
* url 网页的url
* range 特定范围内的数值选择器

**CSS**

6. css单位中px和em,rem的区别

* px实际上就是像素，用px设置字体大小时，比较稳定和精确。但是px不支持用户进行浏览器缩放或者不同移动端的兼容，因为像素是固定的，屏幕大小是变化的。
* em是根据父元素来对应大小，是一种相对值；em值 = 1/父元素的font-size\*需要转换的像素值。进行任何元素设置，都有可能需要知道他父元素的大小，这很不方便。
* rem是根据根元素html的font-size来对应大小， 1rem = 16px,可以在根元素html中写固定像素也可以写百分比，然后在具体的标签上设置rem。

7. 浮动元素引起的问题和解决办法？

* 父元素的高度无法被撑开，影响与父元素同级的元素
* 与浮动元素同级的非浮动元素（内联元素）会跟随其后
* 若非第一个元素浮动，则该元素之前的元素也需要浮动，否则会影响页面显示的结构
* 解决方法：   
  使用css中的clear:both;属性来清除元素的浮动可解决2、3问题，对于问题1，给父元素添加clearfix样式

8. css3有哪些新特性？

* css3实现圆角（border-radius），阴影（box-shadow），
* 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

**JS**

9. 添加、移除、替换、插入节点的方法

* appendchild() //添加
* removechild() //移除
* replacechild() //替换
* insertbefore() //插入

10. 列举几个数组方法

* push()尾部添加 pop()尾部删除
* unshift()头部添加 shift()头部删除

11. 编写一个去掉数组重复元素的方法

function uniq(array){

var temp = [];

for(var i = 0; i < array.length; i++){

if(temp.indexof(array[i]) == -1){

temp.push(array[i]);

}

}

return temp;

}

var aa = [1,2,2,4,9,6,7,5,2,3,5,6,5];

console.log(uniq(aa));

12. split() join() 的区别

* 前者是将字符串切割成数组的形式，后者是将数组转换成字符串

13. 截取字符串 “abcdefg“ 中的 ”egf”

* var str=”abcdefg”
* console.log(str.substring(4)

14. js的typeof返回哪些数据类型？例举3种强制类型转换和2种隐式类型转换?

* 返回类型:undefined、number、string、boolean、object：
* 强制转换:parseint parsefloat number；
* 隐式转换：== - ===

15. 这个正则表达式用来验证什么的isidcard2=/^[1-9]d{5}[1-9]d{3}((0d)|(1[0-2]))(([0|1|2]d)|3[0-1])d{4}$/;

* 18位的身份证

16. eval是做什么的？

* 它的功能是把对应的字符串解析成js代码并运行；由json字符串转换为json对象的时候可以用eval，var obj =eval('('+ str +')'); 应该避免使用eval，不安全，非常耗性能（2次，一次解析成js语句，一次执行）。

17. es6新特性

* 箭头操作符 inputs=>outputs: 操作符左边是输入的参数，而右边则是进行的操作以及返回的值
* 支持类, 引入了class关键字. es6提供的类实际上就是js原型模式的包装
* 字符串模板: es6中允许使用反引号 ` 来创建字符串，此种方法创建的字符串里面可以包含由美元符号加花括号包裹的变量${vraible}。
* 自动解析数组或对象中的值。比如若一个函数要返回多个值，常规的做法是返回一个对象，将每个值做为这个对象的属性返回。但在es6中，利用解构这一特性，可以直接返回一个数组，然后数组中的值会自动被解析到对应接收该值的变量中。
* 拓展参数则是另一种形式的语法糖，它允许传递数组或者类数组直接做为函数的参数而不用通过apply。
* let和const关键字: 可以把let看成var，只是它定义的变量被限定在了特定范围内才能使用，而离开这个范围则无效。const则很直观，用来定义常量，即无法被更改值的变量。

18. let和const的区别？

* let声明的变量可以改变，值和类型都可以改变，没有限制。
* const声明的变量不得改变值

**vue**

19. 如何理解mvvm模式？

* mvvm全称是model-view-viewmodel；vue是以数据为驱动的，一旦创建dom和数据就保持同步，每当数据发生变化时，dom也会变化。domlisteners和databindings是实现双向绑定的关键。domlisteners监听页面所有view层dom元素的变化，当发生变化，model层的数据随之变化；databindings监听model层的数据，当数据发生变化，view层的dom元素随之变化。

20. 说出至少4种vue当中的指令和它的用法？

* v-if：判断是否隐藏；
* v-for：数据循环出来；
* v-bind:class：绑定一个属性；
* v-model：实现双向绑定

21. computed，mothods，watch的区别？

* computed是计算属性,有getter,setter属性,setter接受,getter返回
* mothods是点击事件触发的,存放一下函数方法
* watch是监听,有newquestion, oldquestion两个参数,一个新值,一个旧值

22. v-if 和 v-show的区别

* v-if是动态的向dom树内添加或者删除dom元素；适合运营条件不大可能改变；
* v-show是通过设置dom元素的display样式属性控制显隐；适合频繁切换。

23. <keep-alive></keep-alive>的作用

* 是vue的内置组件，能在组件切换过程中将状态保留在内存中，防止重复渲染dom。（比如有一个列表和一个详情，那么用户就会经常执行打开详情=>返回列表=>打开详情…这样的话列表和详情都是一个频率很高的页面，那么就可以对列表组件使用<keep-alive></keep-alive>进行缓存，这样用户每次返回列表的时候，都能从缓存中快速渲染，而不是重新渲染）

24. 请说下封装 vue 组件的过程？

* 使用vue.extend方法创建一个组件，然后使用vue.component方法注册组件。子组件需要数据，可以在props中接受定义。而子组件修改好数据后，想把数据传递给父组件。

25. vue路由之间跳转的方法？

* 声明式（标签跳转） <router-link :to="index">
* 编程式（ js跳转）router.push('index')

26. 服务器请求数据的方式

* vue 支持开发者引入 jquery 使用 $.ajax()
* vue.resource

（1）npm 安装 vue-resource

  npm install vue-resource<br>

（2）main.js 中引入

import vueresource from 'vue-resource'

vue.use(vueresource)<br>

（3）使用

this.$http.get('../src/data/a.txt')

    .then(function(res){

          alert(res.data);

    },function(){

          alert('false')

});

* axios

（1）main.js引入迭代补进来的

import axios from 'axios'

（2）配置请求信息

var $http = axios.create({

　baseurl: '请求路径',

　timeout: '3000',  //请求超时时间

　headers: {'x-custom-header': 'foobar'}     //header传值，例如：authorization

})

vue.prototype.$http = $http

（3）组件中使用：

this.$http.post()

**小程序**

27.小程序有哪些参数传值的方法？

* 给html元素添加data-\*属性来传递我们需要的值，然后通过e.currenttarget.dataset或onload的param参数获取。但data-名称不能有大写字母和不可以存放对象
* 设置id 的方法标识来传值通过e.currenttarget.id获取设置的id的值,然后通过设置全局对象的方式来传递数值
* 在navigator中添加参数传值

28. bindtap和catchtap的区别是什么？

* bind事件绑定不会阻止冒泡事件向上冒泡，
* catch事件绑定可以阻止冒泡事件向上冒泡

29.简述下wx.navigateto(), wx.redirectto(), wx.switchtab(), wx.navigateback(),的区别？

* wx.navigateto() 保留当前页面 跳转道应用内的某个页面
* wx.redirectto() 关闭当前页面 跳转到应用内的某个页面
* wx.switchtab() 跳转tabbar 页面 并关闭其他所有tabbar 页面
* wx.navigateback() 返回上页面或多级页面

30. 小程序跳转到外部h5 的方法 及跳转外部h5 遇到的问题

* 使用小程序中的组件 web-view