**一、HTML&CSS**

●基本的布局的几种实现方法（部分固定其余自适应等）？

1、水平居中:

1)子元素margin:0 auto

2)父元素text-align:center,子元素inline-block

3)父元素position:relative,子元素position:absolute;left:50%;transform:translateX(-50%)

4)父元素display:flex;justify-content:center;(需要子元素有宽度)

2、垂直居中:

1)display:inline-block;vertical-align:middle

2)父元素position:relative,子元素position:absolute;top:50%;transform:translateY(-50%)

3)line-height:200px

2、水平垂直居中:

1)父元素position:relative,

子元素position:absolute;margin:auto;top:0;left=0;right:0;bottom:0

2)父元素display:tablecell;vertical-align:center;text-align:center

3)父元素display:flex;align-items:center;justify-content:center

3、左侧固定右侧自适应

1)#left{float:left;width:200px}

#right{margin-left:200px}

2)父元素display:flex

子元素左flex:0 0 200px

子元素右flex:1

3)元素左float:left;

元素右:float:left;width:calc(100vw-200px)

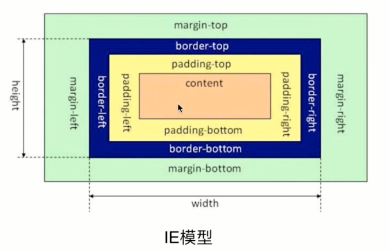
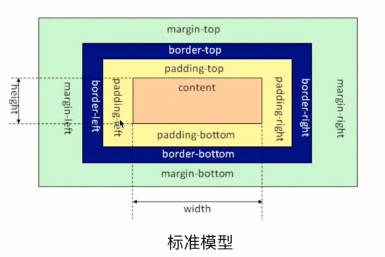
4、两边固定中间自适应的三列布局

1)元素左float:left;width:200px;

元素右float:right;width:100px;

元素中margin:0 120px 0 220px

●盒模型的了解？



1、如何设置两种模型：

/\* 标准模型 \*/

box-sizing:content-box;

/\*IE模型\*/

box-sizing:border-box;

2、通过JS获取盒模型对应的宽高：

1)dom.offsetWidth/offsetHeight

2)window.getComputedStyle(dom).width/height

●阐述块级格式化上下文（Block Formatting Context）及其工作原理。

1、具有 BFC 特性的元素可以看作是隔离了的独立容器，容器里面的元素不会在布局上影响到外面的元素，并且 BFC 具有普通容器所没有的一些特性。

2、触发BFC：只要元素满足下面任一条件即可触发 BFC 特性：

1)body 根元素

2)浮动元素：float 除 none 以外的值

3)绝对定位元素：position (absolute、fixed)

4)display 为 inline-block、table-cells、flex

5)overflow 除了 visible 以外的值 (hidden、auto、scroll)

3、应用：

1)同一个 BFC 下外边距会发生折叠

2)BFC 可以包含浮动的元素（清除浮动）

3)BFC 可以阻止元素被浮动元素覆盖

●几种position之间的区别？

常用四种：static、relative、absolute、fixed

1、static:默认属性。

2、relative:相对定位，相对定位相对的是它原本在文档流中的位置而进行的偏移，没有脱离文档流，但是它的top/left/right/bottom属性是生效的

3、absolute:绝对定位，根据祖先类元素(父类以上)第一个position非static进行定位，脱离文档流，如果使用absolute或fixed定位的话，必须指定 left、right、top、bottom属性中的至少一个，否则left/right/top/bottom属性会使用它们的默认值 auto。

4、fixed:以浏览器窗口定位，脱离文档流。

●CSS层次关系的理解？

1、在没有定位属性的兄弟元素中，一般在html结构下面的元素层级高于结构上面的元素。

2、兄弟元素中有定位属性的元素层级高于没有这两个属性的其他兄弟元素。

3、兄弟元素中有position:relative和position:absolute定位属性的元素，结构下面的元素层级高于结构上面的元素，但是有z-index设置时，z-index值大者居上。

4、非兄弟元素，任意一者或其祖元素拥有动态定位时，同时各自向上寻找动态定位的祖元素，并分别从中拿出具备最高级别的祖元素（或其本身）进行比较（子元素的z-index无论多大，父元素大者居上）。

5、父子关系是无法用z-index来设定上下关系的，一定是子级在上父级在下。

6、使用static 定位或无position定位的元素z-index属性是无效的。

●伪元素伪类的了解？

●cookie、localstorage、session区别？

●display: none和visibility:hidden的区别？Display的各个属性值的区别？

●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？

●HTML5的新特性？

●css3新增的特性？ CSS3新增伪类有那些？

●移动端兼容性考虑过哪些？

●http状态码有那些？分别代表是什么意思？

●一个页面从输入 URL 到页面加载显示完成，这个过程中都发生了什么？

●说说你对Flex的理解，rem你怎么理解的？

●HTTP与HTTPS的区别？

●HTTP响应头组成结构？

●只允许使用css，让一个页面原有dom不可见？

●为什么cookie不安全？

●浏览器CSS Hack手段有哪些？

●css选择器有哪些？CSS 选择器的优先级是如何计算的？

●为什么最好把 CSS 的<link>标签放在<head></head>之间？为什么最好把 JS 的<script>标签恰好放在</body>之前，有例外情况吗？

●有哪些清除浮动的技术，都适用哪些情况？

●请阐述Float定位的工作原理。

●使用 CSS 预处理的优缺点分别是什么？

**二、Javascript**

●数据类型有哪些？

●简述原型链？

●使用new操作符调用函数创建实例的执行步骤？

●正则表达式

●promise对象？

●如何识别不同硬件设备？

●原生ajax的实现，具体思路

●变量提升？

●ES6的了解

●DOM操作——怎样添加、移除、移动、复制、创建和查找节点?

●那些操作会造成内存泄漏？

●网站性能优化的方法

●对象属性的遍历？

●Dom0级和dom2级事件之间的区别

●深度拷贝和浅拷贝？

●排序算法？

●数组去重？

●为什么浏览器需要同源策略？浏览器跨域的几种方式？

●怎样判断一个数组是数组？

●怎么用JS实现一个队列？

●怎么发送一个跨域的POST请求？

●jpg格式和png格式怎么区别？gif呢？什么情况下分别选择？

●闭包介绍一下你的理解？用处，缺点？

●堆和栈有什么区别？

●请描述事件冒泡，怎么取消事件冒泡？

●[请解释事件委托（event delegation）](https://github.com/yangshun/front-end-interview-handbook/blob/master/Translations/Chinese/questions/javascript-questions.md" \l "%E8%AF%B7%E8%A7%A3%E9%87%8A%E4%BA%8B%E4%BB%B6%E5%A7%94%E6%89%98event-delegation)

●请简述JavaScript中的this。

●说说你对 AMD 和 CommonJS 的了解

●请解释下面代码为什么不能用作 IIFE：function foo(){ }();，需要作出哪些修改才能使其成为 IIFE？

●请说明.forEach循环和.map()循环的主要区别，它们分别在什么情况下使用？

●匿名函数的典型应用场景是什么？

●document 中的load事件和DOMContentLoaded事件之间的区别是什么？

●创建一个循环，从 1 迭代到 100，3的倍数时输出 "fizz"，5的倍数时输出 "buzz"，同时为3和5的倍数时输出 "fizzbuzz"。

●为什么不要使用全局作用域？

●你使用什么工具和技巧调试 JavaScript 代码？

●你使用什么语句遍历对象的属性和数组的元素？

●请解释同步和异步函数之间的区别。

●什么是事件循环？调用堆栈和任务队列之间有什么区别？

●使用let、var和const创建变量有什么区别？

●请解释function foo() {}和var foo = function() {}之间foo的用法上的区别。

●你能给出一个使用箭头函数的例子吗，箭头函数与其他函数有什么不同？在构造函数中使用箭头函数有什么好处？

●documen.write和 innerHTML的区别

●.call() 和 .apply() 的区别？

●如何编写高性能的Javascript？

**三、VUE**

●vue如何实现双向数据绑定？能实现一下双向绑定么

vue.js 是采用数据劫持结合发布者-订阅者模式的方式，通过Object.defineProperty()来劫持各个属性的setter，getter，在数据变动时发布消息给订阅者，触发相应的监听回调。

具体步骤：

第一步：需要observe的数据对象进行递归遍历，包括子属性对象的属性，都加上 setter和getter

这样的话，给这个对象的某个值赋值，就会触发setter，那么就能监听到了数据变化

第二步：compile解析模板指令，将模板中的变量替换成数据，然后初始化渲染页面视图，并将每个指令对应的节点绑定更新函数，添加监听数据的订阅者，一旦数据有变动，收到通知，更新视图

第三步：Watcher订阅者是Observer和Compile之间通信的桥梁，主要做的事情是:

1、在自身实例化时往属性订阅器(dep)里面添加自己

2、自身必须有一个update()方法

3、待属性变动dep.notice()通知时，能调用自身的update()方法，并触发Compile中绑定的回调，则功成身退。

第四步：MVVM作为数据绑定的入口，整合Observer、Compile和Watcher三者，通过Observer来监听自己的model数据变化，通过Compile来解析编译模板指令，最终利用Watcher搭起Observer和Compile之间的通信桥梁，达到数据变化 -> 视图更新；视图交互变化(input) -> 数据model变更的双向绑定效果。

●mvvm框架是什么？它和其它框架（jquery）的区别是什么？哪些场景适合？

●导航钩子有哪些？介绍一下他们

●请详细说下你对vue生命周期的理解？

●vue的组件是怎么定义的？父组件怎么给子组件传值？组件间如何通信？

●请说出vue.cli项目中src目录每个文件夹和文件的用法？

●你觉得哪些项目适合vue框架？

●怎么定义vue-router的动态路由？怎么获取传过来的动态参数？

●axios是什么？怎么使用？

●vuex是什么？怎么使用？哪种功能场景使用它？