###知识体系，触类旁通，列举案例  
  
关于XXX并不太清楚，我只记得YYY是ZZZ。YYY的目的应用场景应该是。  
  
面试官问的技术问题，解释相关的知识点。  
  
###自我介绍  
您好，我叫XXX，本科和硕士都就读于哈尔滨工程大学的一名应届毕业生。热爱前端，对前端的兴趣起源于自己对用ECMAScript代码制作弹幕动画的研究，喜欢JavaScript，喜欢技术与艺术的结合。  
有自己维护的博客，GitHub活跃，每天坚持编程。近几个月也刚刚参加百度前端技术学院的公开训练，受益匪浅。平常学习的方式就是看书，看demo，写demo，做笔记，做项目，也经常关注国内外的前端资讯。自己以全栈为目标，希望能在前端这条路上一直深入下去。  
  
博客: http://ppii.in：  
  
###物联网智能家居控制平台：  
这是一个基于Nodejs的智能家居控制应用。生产环境是嵌入式Linux路由器。主要实现（接收来自用户各类终端的控制信息，）对连接的电子设备(如空调电视照明摄像头等)进行操作控制和监控。  
我主要完成的是，整体软硬件的技术选型，和前后端代码的编写。  
  
我主要讲讲前端部分：  
1. 用knockout.js不用angular.js的原因。  
它们都是MVVM，所以在技术选型上我认为区别应该不大。knockout相对于ng来说，更轻量级，也更容易上手。  
因为物联网应用是这样的，必须要保证，前端显示、后端数据、硬件的状态必须同步。所以，使用MVVM的数据绑定是最好的选择。  
<http://www.cnblogs.com/aaronjs/archive/2013/06/16/3138631.html>  
<http://javascript.ruanyifeng.com/dom/mutationobserver.html#toc0>  
2.什么是MVVM？  
  
3. 什么是web响应式设计？  
  
<file:///Q:/GitRepo/Notes/FrontEnd-Interview/web%E6%8A%80%E6%9C%AF%E4%B8%8E%E5%93%8D%E5%BA%94%E5%BC%8F%E8%AE%BE%E8%AE%A1.html>  
  
后端部分：  
1. PHP Node.js  
因为生产环境上当时只有PHP的软件源，所以一开始是用的PHP，来设计资源的API。  
后来考虑到Node.js实时性强，同时直接可以用JavaScript来写，开发也更快捷。  
拿源码交叉编译了Node.js的软件包，在路由器上搭建了nodejs的环境。  
  
先根据下级电路的特征设计了数据的模型，  
2. 什么是REST？  
  
遇到了什么问题：  
1. 在设计为了让硬件设备状态和前端显示同步的时候，（因为有两种设备，一种是状态设备，就是在前端交互的变化直接传递给连接的硬件）但是，HTTP是无状态的协议，服务器这头不可能实现给指定的设备发送数据，所以我一开始采用的是，设备客户端循环的轮询，一定间隔地发送get请求，可是这样就会造成很大的开销。  
所以我查阅了一些相关的解决方案，包括websocket、长轮询长连接的方式，但是有一个比较大的问题就是，硬件设备的wifi芯片不支持这些协议。后来我找到的一种解决方案是，IBM的一种协议MQTT……(消息队列遥测传输)  
    轻量级的 machine-to-machine 通信协议。  
    publish/subscribe模式。  
    基于TCP/IP  
    适合于低带宽、不可靠连接、嵌入式设备、CPU内存资源紧张。  
    是一种比较不错的Android消息推送方案。  
  
2.什么是websocket？  
3. 有一个需求是摄像头监控，刚好摄像头这部分侦听的是另外一个端口。所以这里涉及到了解决跨域的问题。  
  
4. 前端方面的问题，一级界面有一批正方形的button来链接到对应设备的二级界面。为了实现按钮的宽高比一直保持1比1，当初是，宽度可以用百分比来确定，但是在缩放过程中，高度没法跟着变。后来查阅相关响应式布局的资料，才知道有vw,vh这样的单位，直接设置相同的vw就解决了。  
  
5. 在写CSS的过程中，写的多的时候，全是单个的独立块，都最后都不太好管理。虽然一开始用注释分开模块，或者分文件的方式可以。但是会带来新的问题，比如增加了http的请求。后来了解到一些CSS预处理和后处理器，一点点迁移到了less上，然后写完再做编译加文件合并。  
  
前端：  
1. jsonp,同源策略  
<https://developer.mozilla.org/zh-CN/docs/Web/Security/Same-origin\_policy>  
    a)window.name一直保存，即使刷新、URL跳转  
    b)  
  
3. ajax  
4. nodejs  
5. html5  
6. http协议  
7. jQuery，原生JavaScript  
  
8. websocket  
  
  
  
后端：  
 1. restful→JSON-RPC  
 2. mysql操作  
 3.   
百度前端技术学院的任务项目：  
任务1：  
1. CSS布局 2. 语义化标签  
http://www.zhangxinxu.com/wordpress/2010/01/%E5%AF%B9overflow%E4%B8%8Ezoom%E6%B8%85%E9%99%A4%E6%B5%AE%E5%8A%A8%E7%9A%84%E4%B8%80%E4%BA%9B%E8%AE%A4%E8%AF%86/  
  
水平居中：  
    1. 块级元素 定义宽度， margin:auto  
    2. 父元素：text-align center; display: inline-block  
    3. absolute left:50%, margin-left:-(content-width/2)  
    ↓3. （float: left;）相对父级position: absolute; left: 50%;子级设置relative right：50%  
垂直居中：  
    1. 父级display:table 子集table-cell vertical-align: middle;   
    2. absolute ,top 50%, margin-top:-content高度，(设置overflow:auto防止溢出）  
    3. 外加一个div包裹，height为50%，margin-bottom为-content高度，→content清除浮动  
    4. absolute。设定固定宽高，并且top,bottom都为0，margin:auto会使他居中。（content会被截断）  
    5. height等于line-height  
都居中：  
    3. 父级display:table 子集table-cell vertical-align: middle; text-align: center;  
    4. position: absolute; top: 50%; left: 50%;transform: translate(-50%, -50%);  
    5. display: flex , justify-content: center  
清除浮动：  
    1、<div style=”clear:both”>  
    2. br 有 clear=“all | left | right | none”   
    3. 父元素设置 overflow：hidden （当内容溢出元素框时发生的事情。默认visible，scroll，继承）  
    4. 父元素设置 overflow：auto 属性  
    5. .clearfix:after {  
        content: ".";  
        display: block;  
        height: 0;  
        clear: both;  
        visibility: hidden;  
    }  
什么是 BFC：  
    会计格式上下文。每个BFC是独立布局的，盒子内的元素不会影响到外面的元素  
    垂直方向上的margin会重叠  
会触发BFC：  
    float 除了none以外的值   
    overflow 除了visible 以外的值（hidden，auto，scroll ）   
    display (table-cell，table-caption，inline-block)   
    position（absolute，fixed）  
BFC特性：  
    1. 块级格式化上下文会阻止外边距叠加   
    2)块级格式化上下文不会重叠浮动元素  
    3)块级格式化上下文通常可以包含浮动  
  
当一个元素的hasLayout属性值为true时，它负责对自己和可能的子孙元素进行尺寸计算和定位  
  
任务2：图片轮播、倒计时、输入框即时提示、拖拽交互等任务  
  
图片轮播：  
将所有图片排成一排，然后在跟一张图片一样大小的设置relative,overflow hidden.  
  
任务3：  
  
http://jerryzou.com/posts/cookie-and-web-storage/  
  
http://www.cnblogs.com/beiyuu/archive/2011/07/20/js-localstorage-userdata.html  
  
http://www.hbrc.com/rczx/news-945147.html  
  
LB3：  
那是我召集的一批程序员9和绘画师7，根据自己设计的动画分镜，来创作弹幕动画。  
（我自己也参与进重要模块的制作）  
工作流程的话呢，是让绘画师先根据分镜，先用自己写的矢量绘图工具来制作人物、背景模型，然后程序员依据这些模型根据故事的感情要素来做动画特效和一些滤镜等视觉效果处理，完成整个弹幕动画。整个过程都是在记事本上写类似ECMAScript绘图代码的过程。