

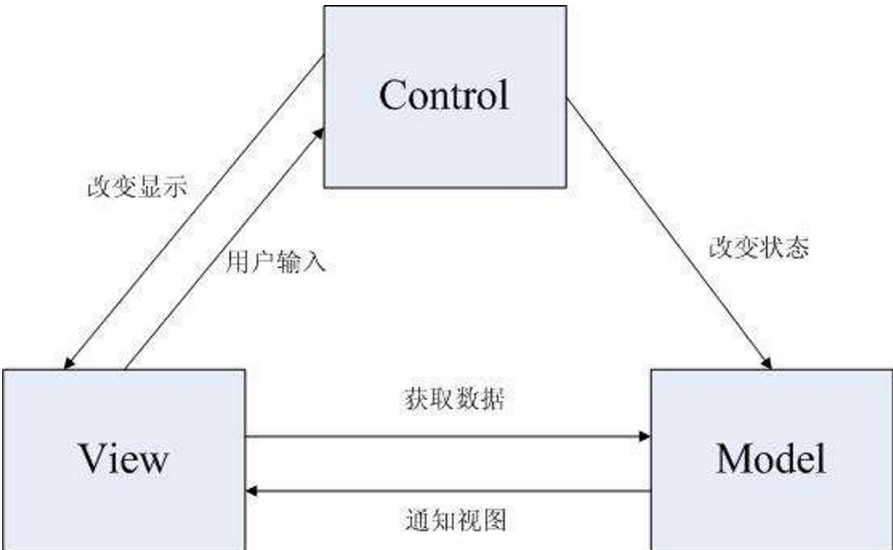
★ 第一题：什么是MVC？

MVC

MVC是一种软件架构模式，也有人叫做设计模式

- M: Model 数据模型（专门用来操作数据，数据的CRUD-增删改查）
- V: View 视图（对于前端来说，就是页面）
- C: Controller 控制器（是视图和数据模型沟通的桥梁，用于处理业务逻辑）

Model（模型）：



MVC vs MVVM

- MVC是Model-View- Controller的简写。即模型-视图-控制器。  
M和V指的意思和MVVM中的M和V意思一样。C即Controller指的是页面业务逻辑。使用MVC的目的就是将M和V的代码分离。MVC是单向通信。也就是View跟Model，必须通过Controller来承上启下。MVC和MVVM的区别并不是VM完全取代了C，只是在MVC的基础上增加了一层VM，只不过是弱化了C的概念，ViewModel存在目的在于抽离Controller中展示的业务逻辑，而不是替代Controller，其它视图操作业务等还是应该放在Controller中实现。也就是说MVVM实现的是业务逻辑组件的重用，使开发更高效，结构更清晰，增加代码的复用性。
- MVVM即Model-View-ViewModel的简写。即模型-视图-视图模型。  
模型（Model）指的是后端传递的数据。视图(View)指的是所看到的页面。视图模型(ViewModel)是mvvm模式的核心，它是连接view和model的桥梁。它有两个方向：一是将模型（Model）转化成视图(View)，即将后端传递的数据转化成所看到的页面。实现的方式是：数据绑定。二是将视图(View)转化成模型(Model)，即将所看到的页面转化成后端的数据。实现的方式是：DOM 事件监听。这两个方向都实现的，我们称之为数据的双向绑定。



- VUE官方，也介绍了MVVM模型相关框架（如Vue 、 React、 AngularJS、 Knockout、 Polymer等）的对比优缺点，可以参考。<https://cn.vuejs.org/v2/guide/comparison.html>

★ 第二题： Ajax与XHR的关系

Ajax的异步的实现是通过new 一个XMLHttpRequest对象， XMLHttpRequest取首字母就是XHR。  
换句话说，Ajax的核心是XMLHttpRequest对象，xhr是老师自定义的对象名。你也可以写成这样：var abc =

```
new XMLHttpRequest()
```

★ 第三题：Ajax请求过程

- 完整的Ajax请求过程：
- (1)创建XHR实例 ( XMLHttpRequest 或 ActiveXObject )
  - (2)发出HTTP请求 ( send()、 open() )
  - (3)接收服务器传回的数据
  - (4)更新网页数据

★ 第四题：常见的HTTP状态码有哪些

常用的状态码：200 成功 302重定向 304缓存 404 找不到 5xx 服务器错误

状态码	响应类别	原因
1xx	信息性状态码(Informational)	服务器正在处理请求
2xx	成功状态码(Success)	请求已正常处理完毕
3xx	重定向状态码(Redirection)	需要进行额外操作以完成请求
4xx	客户端错误状态码(Client Error)	客户端原因导致服务器无法处理请求
5xx	服务端状态码(Server Error)	服务器原因导致处理请求出错

```
// 1. 获取状态，xhr.status 可以获取响应的状态码
// 200 成功 302重定向 304缓存 404 找不到 5xx 服务器错误
```

★ 第五题：cookie和session的区别

- 1、cookie数据存放在客户的浏览器上，session数据放在服务器上
- 2、cookie不是很安全，别人可以分析存放在本地的cookie并进行cookie欺骗，考虑\*到安全应当使用session
- 3、session会在一定时间内保存在服务器上，当访问增多，会比较占用你服务器的性能，考虑到减轻服务器性能方面，应当使用cookie
- 4、单个cookie保存的数\*据不能超过4K，很多浏览器都限制一个站点最多保存20个cookie
- 5、建议将登录信息等重要信息存放为session，其他信息如果需要保留，可以放在cookie中
- 6、session保存在服务器，客户端不知道其中的信心；cookie保存在客户端，服务器能够知道其中的信息
- 7、session中保存的是对象，cookie中保存的是字符串
- 8、session不能区分路径，同一个用户在访问一个网站期间，所有的session在任何一个地方都可以访问到，而cookie中如果设置了路径参数，那么同一个网站中不同路径下的cookie互相是访问不到的\*

★ 第六题：sessionStorage、localStorage和cookie的区别

**共同点：**都是保存在浏览器端、且同源的

**区别：**

- 1、cookie数据始终在同源的http请求中携带（即使不需要），即cookie在浏览器和服务器间来回传递，而sessionStorage和localStorage不会自动把数据发送给服务器，仅在本地保存。cookie数据还有路径（path）的概念，可以限制cookie只属于某个路径下
- 2、存储大小限制也不同，cookie数据不能超过4K，同时因为每次http请求都会携带cookie、所以cookie只适合保存很小的数据，如会话标识。sessionStorage和localStorage虽然也有存储大小的限制，但比cookie大得多，

可以达到5M或更大

3、数据有效期不同，sessionStorage：仅在当前浏览器窗口关闭之前有效；localStorage：始终有效，窗口或浏览器关闭也一直保存，因此用作持久数据；cookie：只在设置的cookie过期时间之前有效，即使窗口关闭或浏览器关闭

4、作用域不同，sessionStorage不在不同的浏览器窗口中共享，即使是同一个页面；localStorage在所有同源窗口中都是共享的；cookie也是在所有同源窗口中都是共享的

5、web Storage支持事件通知机制，可以将数据更新的通知发送给监听者

6、web Storage的api接口使用更方便