1. JavaScript模块化的概念解读（面试必问）

2009年，Nodejs 诞生！这个服务器端的 JavaScript 采用模块化的写法很快征服了浏览器端的 JSer 。牛人们纷纷仿效，各种写模块的规范也是层出不穷。CommonJS 想统一前后端的写法，AMD 则认为自己是适合浏览器端的。国内的 CMD 非常成熟。

起源于nodejs，把代码打包成模块，采用require引用。

AMD是RequireJS在推广过程中对模块定义的规范化产出。

* 异步加载模块，以来前置，提前执行。
* Define定义模块define([‘require’,’foo’],function(){return});
* Require加载模块（以来前置）require([‘foo’,’bar’],function(foo,bar){});

CMD是SeaJS在推广过程中对模块定义的规范化产出。

* Define定义exports导出define(function(require,exports,module){});

Module上存储了当前模块的一些对象。

* Require(./a)直接引入。Requeir.async异步引入。
* 同步加载，依赖就近，延迟执行。

类似的还有CommonJS Modules/2.0规范。

1. JavaScript数据推送

数据推送在当下非常火热，我们的手机每天都能收到来自APP的各种消息，JavaScript数据推送主要致力于webapp的在线推送服务，不用我们每次都像服务器去发送Ajax请求而主动从Server端推送数据到本地。在早期必须保持着一个链接不放，或者通过前端不停的向后端发送请求，H5更新后有了WebSocket大大改善了双向和单向推送数据的便利性。

1. Comet：基于HTTP长链接的服务器推送技术

基于 HTTP 长连接的服务器推送技术，Comet 是一种 Web 应用架构。服务器端会主动以异步的方式向客户端程序推送数据，而不需要客户端显式的发出请求。Comet 架构非常适合事件驱动的 Web 应用，以及对交互性和实时性要求很强的应用，如股票交易行情分析、聊天室和 Web 版在线游戏等。

1. 基于WebSocket的推送方案

基于WebSocket 的推送方案，在浏览器中通过http仅能实现单向的通信,comet可以一定程度上模拟双向通信,但效率较低,并需要服务器有较好的支持; flash中的socket和xmlsocket可以实现真正的双向通信,通过 flex ajax bridge,可以在javascript中使用这两项功能. 可以预见,如果websocket一旦在浏览器中得到实现,将会替代上面两项技术,得到广泛的使用.面对这种状况，HTML5定义了WebSocket协议，能更好的节省服务器资源和带宽并达到实时通讯。

1. SSE(Server-Send Event)：服务器推送数据的新方式

服务器推送数据的新方式SSE，传统的网页都是浏览器向服务器"查询"数据，但是很多场合，最有效的方式是服务器向浏览器"发送"数据。比如，每当收到新的电子邮件,服务器就向浏览器发送一个"通知",这要比浏览器按时向服务器查询（polling）更有效率。服务器发送事件（Server-Sent Events，简称SSE）就是为了解决这个问题，而提出的一种新API，部署在EventSource对象上。目前，除了IE，其他主流浏览器都支持。