1.声明变量的方法有几种：

ES5只有两种声明变量的方法：var命令和function命令。ES6除了添加let和const命令，后面章节还会提到，另外两种声明变量的方法：import命令和class命令。所以，ES6一共有6种声明变量的方法。

2.其它JS规范了解摘录：

2.1.CommonsJS规范

CommonJS 是以在浏览器环境之外构建 JavaScript 生态系统为目标而产生的项目，比如在服务器和桌面环境中。

这个项目最开始是由 Mozilla 的工程师 Kevin Dangoor 在2009年1月创建的，当时的名字是 ServerJS。

我在这里描述的并不是一个技术问题，而是一件重大的事情，让大家走到一起来做决定，迈出第一步，来建立一个更大更酷的东西。 —— Kevin Dangoor's [What Server Side JavaScript needs](http://www.blueskyonmars.com/2009/01/29/what-server-side-javascript-needs/)

2009年8月，这个项目改名为 CommonJS，以显示其 API 的更广泛实用性。CommonJS 是一套规范，它的创建和核准是开放的。这个规范已经有很多版本和具体实现。CommonJS 并不是属于 ECMAScript TC39 小组的工作，但 TC39 中的一些成员参与 CommonJS 的制定。2013年5月，Node.js 的包管理器 NPM 的作者 Isaac Z. Schlueter 说 [CommonJS 已经过时，Node.js 的内核开发者已经废弃了该规范](https://github.com/nodejs/node-v0.x-archive/issues/5132#issuecomment-15432598)。

CommonJS 规范是为了解决 JavaScript 的作用域问题而定义的模块形式，可以使每个模块在它自身的命名空间中执行。该规范的主要内容是，模块必须通过 module.exports 导出对外的变量或接口，通过require() 来导入其他模块的输出到当前模块作用域中。

2.2.AMD规范

AMD（异步模块定义）是为浏览器环境设计的，因为 CommonJS 模块系统是同步加载的，当前浏览器环境还没有准备好同步加载模块的条件。

AMD 定义了一套 JavaScript 模块依赖异步加载标准，来解决同步加载的问题。

模块通过 define 函数定义在闭包中，格式如下：

define(id?: String, dependencies?: String[], factory: Function|Object);

3.