AMD 规范在这里：[https://github.com/amdjs/amdjs-api/wiki/AMD](//link.zhihu.com/?target=https%3A//github.com/amdjs/amdjs-api/wiki/AMD)  
CMD 规范在这里：[https://github.com/seajs/seajs/issues/242](//link.zhihu.com/?target=https%3A//github.com/seajs/seajs/issues/242)  
  
AMD 是 RequireJS 在推广过程中对模块定义的规范化产出。  
CMD 是 SeaJS 在推广过程中对模块定义的规范化产出。  
类似的还有 CommonJS Modules/2.0 规范，是 BravoJS 在推广过程中对模块定义的规范化产出。  
还有不少⋯⋯  
  
这些规范的目的都是为了 JavaScript 的模块化开发，特别是在浏览器端的。  
目前这些规范的实现都能达成**浏览器端模块化开发的目的**。  
  
区别：  
  
1. 对于依赖的模块，AMD 是**提前执行**，CMD 是**延迟执行**。不过 RequireJS 从 2.0 开始，也改成可以延迟执行（根据写法不同，处理方式不同）。CMD 推崇 as lazy as possible.  
  
2. CMD 推崇**依赖就近**，AMD 推崇**依赖前置**。看代码：  
  
// CMD  
define(function(require, exports, module) {  
var a = require('./a')  
a.doSomething()  
// 此处略去 100 行  
var b = require('./b') // 依赖可以就近书写  
b.doSomething()  
// ...   
})  
  
// AMD 默认推荐的是  
define(['./a', './b'], function(a, b) { // 依赖必须一开始就写好  
a.doSomething()  
// 此处略去 100 行  
b.doSomething()  
...  
})   
  
虽然 AMD 也支持 CMD 的写法，同时还支持将 require 作为依赖项传递，但 RequireJS 的作者默认是最喜欢上面的写法，也是官方文档里默认的模块定义写法。  
  
  
3. AMD 的 API 默认是**一个当多个用**，CMD 的 API 严格区分，推崇**职责单一**。比如 AMD 里，require 分全局 require 和局部 require，都叫 require。CMD 里，没有全局 require，而是根据模块系统的完备性，提供 seajs.use 来实现模块系统的加载启动。CMD 里，每个 API 都**简单纯粹**。  
  
  
4. 还有一些细节差异，具体看这个规范的定义就好，就不多说了。  
  
另外，SeaJS 和 RequireJS 的差异，可以参考：[https://github.com/seajs/seajs/issues/277](//link.zhihu.com/?target=https%3A//github.com/seajs/seajs/issues/277)