

后盾人 人人做后盾

www.houdunren.com

ES6模块化

后盾人 2010-2018

历史上，JavaScript 一直没有模块（module）体系，无法将一个大程序拆分成互相依赖的小文件，再用简单的方法拼装起来。其他语言都有这项功能，比如 Ruby 的require、Python 的import，甚至就连 CSS 都有@import，但是 JavaScript 任何这方面的支持都没有，这对开发大型的、复杂的项目形成了巨大障碍。

在 ES6 之前，社区制定了一些模块加载方案，最主要的有 CommonJS 和 AMD 两种。前者用于服务器，后者用于浏览器。ES6 在语言标准的层面上，实现了模块功能，而且实现得相当简单，完全可以取代 CommonJS 和 AMD 规范，成为浏览器和服务器通用的模块解决方案。

ES6 模块的设计思想，是尽量的静态化，使得编译时就能确定模块的依赖关系，以及输入和输出的变量。CommonJS 和 AMD 模块，都只能在运行时确定这些东西。比如，CommonJS 模块就是对象，输入时必须查找对象属性。

ES6模块化

- 模块功能主要由两个命令构成：export和import。export命令用于规定模块的对外接口，import命令用于输入其他模块提供的功能。
- 一个模块就是一个独立的文件。该文件内部的所有变量，外部无法获取。如果你希望外部能够读取模块内部的某个变量，就必须使用export关键字输出该变量。

ES6模块化

浏览器中引入模块文件的时候要注意：

1. 需要对script标签指定 `type='module'` 属性
2. 需要通过http协议访问页面才可以正常引入模块文件
3. 最好通过最新版谷歌浏览器（chrome）进行测试

```
<script src="a.js" type="module"> </script>
```

浏览器中引入模块文件

- export用法1:

```
export var hd = '后盾人';
```

- export用法2:

```
var url = 'www.houdunren.com';
```

```
export {url};
```

- export用法3:

```
var hd = '后盾人';
```

```
var url = 'www.houdunren.com';
```

```
export {hd,url};
```

export用法示例

export命令除了输出变量，还可以输出函数或类（class）。

```
export function foo(x, y) {  
  return x * y;  
};
```

导出函数

- 通常情况下，export输出的变量就是本来的名字，但是可以使用as关键字重命名。

```
var hd = '后盾人';
```

```
var url = 'www.houdunren.com';
```

```
export {hd as houdun,url as website};
```

别名

- export语句输出的接口，与其对应的值是动态绑定关系，即通过该接口，可以取到模块内部实时的值

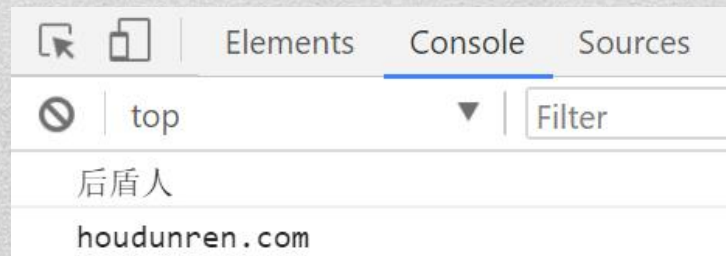
//a.js

```
import {hd} from './b.js';  
console.log(hd);  
setTimeout(function(){  
    console.log(hd);  
},2000)
```

//b.js

```
let hd = '后盾人';  
setTimeout(function(){  
    hd = 'houdunren.com';  
},1000)  
export {hd};
```

结果：



导出函数

- 使用export命令定义了模块的对外接口以后，其他 JS 文件就可以通过import命令加载这个模块。注意，导入时路径要完整，./不可省略

```
import {hd,url,foo} from './b.js';
```

- 如果想为输入的变量重新取一个名字，import命令要使用as关键字，将输入的变量重命名。

```
import {hd as houdun,url} from './b.js';
```

import导入模块

- 从前面的例子可以看出，使用import命令的时候，用户需要知道所要加载的变量名或函数名，否则无法加载。但是，用户肯定希望快速上手，未必愿意阅读文档，去了解模块有哪些属性和方法。
- 为了给用户提供方便，让他们不用阅读文档就能加载模块，就要用到export default命令，为模块指定默认输出。

```
export default hd() {  
  console.log('foo');  
}
```

export default命令

- `export default`命令用于指定模块的默认输出。显然，一个模块只能有一个默认输出，因此`export default`命令只能使用一次。所以，`import`命令后面才不用加大括号，因为只可能对应一个方法。
- 本质上，`export default`就是输出一个叫做`default`的变量或方法，然后系统允许你为它取任意名字。

export default命令

通过export default命令导出的数据不属于数据接口，在导入的时候必须指定名称，且不需要加花括号。

//a.js

```
import y,{hd,url} from './b.js';  
console.log(y);  
console.log(hd);  
console.log(url);
```

//b.js

```
let hd = '后盾人';  
let url = 'houdunren.com';  
let year = 2018;  
export {hd,url};  
export default year;
```

导入