**案例3：AMD方式的模块化开发**

学习内容：

学习基于ADM方式的模块化Javascript开发方式，定义AMD模块和对象; 学习利用模块化的方式完成一个网页部件(Widget)。

基本概念：

1. AMD模块: 是”Asynchronous Module Definition”的缩写，意思就是”异步模块定义”。它采用异步方式加载模块，模块的加载不影响它后面语句的运行。所有依赖这个模块的语句，都定义在一个回调函数中，等到加载完成之后，这个回调函数才会运行。
2. CommonJS模块，一般用于node.js的服务端。CommonJS属于动态加载，核心思想是通过require方法来同步加载其它模块，是一种动态导入方式。
3. ES6模块，一般需要配合node.js一起使用。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。
4. Dojo框架：ArcGIS Server for Javascript API是基于Dojo框架的，Dojo采用AMD方式管理模块。本课程样例采用AMD方式管理模块。

学习步骤：

1. 在index.html文件所在位置，新建app文件，用于存放AMD模块。
2. 添加一个下拉菜单

<div class="dropdown dropdown-menu-end">

<button type="button" class="btn btn-primary dropdown-toggle" data-bs-toggle="dropdown">地  图

</button>

<ul class="dropdown-menu">

<li><a class="dropdown-item" href="#" id="load">装载...</a></li>

    <li><hr class="dropdown-divider"></hr></li>

    <li><a class="dropdown-item" href="#" id="config">配置...</a></li>

</ul>

</div>

1. 配置dojo模块路径文件

var dojoConfig = {

isDebug: true,

async: true,

//在此，引入自定义模块的路径

packages: [

{

"name": "app",//包的名称

"location": location.pathname.replace(/\/[^/]+$/,"")+"/app"   //包的路径

}

]

}

1. 利用define和declare声明AMD模块和对象。

在app文件下创建Menu.js，声明一个模块，其中定义的方法用于处理菜单响应事件；

在app文件夹下，创建一个dialogs文件，在widgets文件下创建LoadMapService.js，在文件中声明一个LoadMapServices对象，该对象动态创建一个模式对话框(dialoget)，会弹出一个Modal对话框。用户可以通过该对话框选择输入地图服务的URL地址和地图服务类型。

Menu.js和LoadMapService.js展示了AMD模块化编程的优势。

LaodMapService.js中也展示了如何动态添加网页的DOM元素，这样比直接写在index.html文件中具有较好的独立性和封装性。

