jsTree

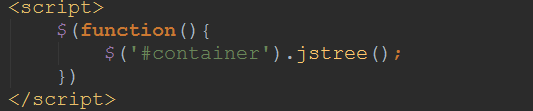
1. 引用文件

Js与css文件都已下载到本地





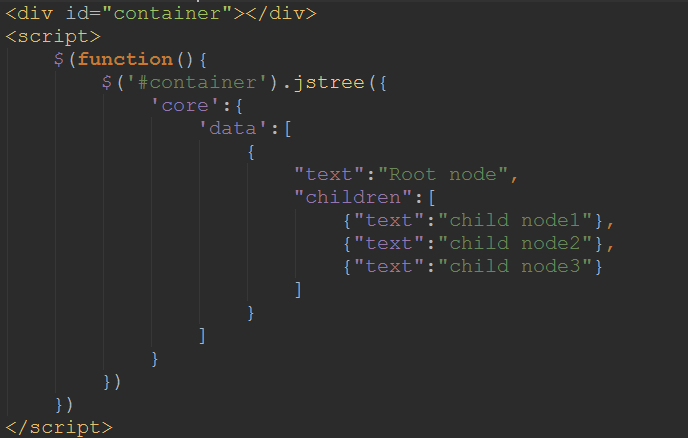
2. 简单调用



3. 支持使用data-\*属性来设置选项，如:



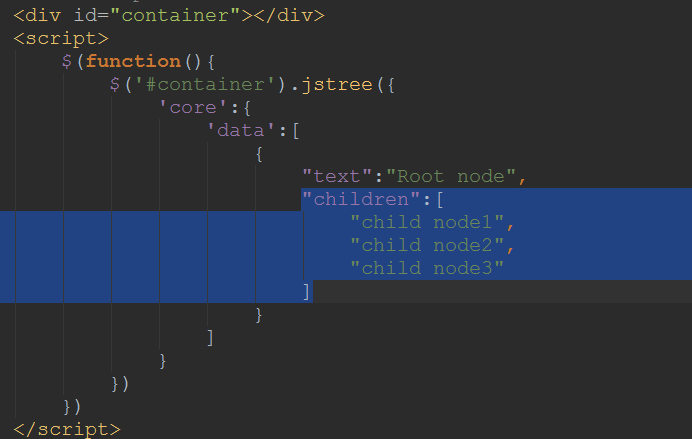
4. HTML定义方法比较简单但不够灵活，所以可使用JS方法：



Js中传入了一个config参数对象，jstree会试图解析core.data定义的任何数据，用来构造节点。如果不设置core.data属性，jstree就会解析HTML树节点

提供的json数据必须符合一定的格式，每个对象对应一个节点，如果该节点还有子节点，还需要提供children属性。Children属性是一个相同格式的对象数组

\*\*注意：如果一个对象只有text属性时，也可以使用字符串替换该对象，而该字符串就是text属性的值，所以上图代码可以改成这样：



节点对象中还提供了一些可选的配置：

id - 这个id会在对应的 li 节点上设置html标签的ID属性

icon - 节点图标，可以设置表示路径、一个或者多个css类名、字体图标的字符串

data - 任何数据，设置此属性没有任何UI上的效果，任何时候都可以读写这个数据

state - 对象类型，一个节点的状态有以下几种：

selected - 节点处于被选中的状态

opend - 节点处于被打开的状态

disabled - 节点不可选

checked - 用于checkbox插件，勾选该checkbox（只有当tie\_selection处于false时有效）

undetermined - 用于checkbox插件，状态待定（只有启用懒加载并且节点没有被加载时生效）

type - 用于types插件，用来定义节点的类型，默认为default类型

li\_attr - 包含DOM属性的对象，会追加到该节点对应的LI标签

a\_attr - 包含DOM属性的对象，会追加到该节点对象的A标签

包含可选属性的DEMO:



5. 使用AJAX定义树



除了给core.data设置数组类型的数据外，还可以设置一个jquery AJAX confifg对象

jsTree会获取URL数据，并显示

6. 使用AJAX和lazy loading（懒加载）定义树

Lazy loading（懒加载）意味只按需加载节点。比如当有大量的数据展示，而一个请求会很占资源的时候，lazy loading可以更快速的加载数据，jsTree会在用户浏览到的时候发送ajax请求获取节点数据



解析：

首先data对象中的 ”data” 配置选项，在jquery中，应该是string或者object类型，而jsTree拓展了一下，使其可以设置为function

每次jsTree需要请求AJAX的时候，就会调用这个function，并将正在打开的节点作为参数传给这个function，该方法的返回值作为AJAX的 “data” 参数

查看控制台请求的信息，可以看到首次请求的链接为：

<http://www.jstree.com/fiddle?lazy&id=#> ，其中 # 是一个比较特别的ID，这个ID表示jsTree当前要加载的根节点

打开root根节点，可以看见两个子节点，这个操作不会触发AJAX请求，因为这两个子节点的信息早已包含在首次请求中

”child node1” 节点处于关闭状态，因为该节点的 ”children” 属性设置了true 这相当于告诉jsTree，该节点将会触发lazy load

打开 “child node1” 节点，出现第二个请求：

<http://www.jstree.com/fiddle?lazy&id=2>

ID=2，对应了 “child node1“ 节点的ID，也是data属性设置的function返回的结果

也可以把 “url” 设置为一个function效果和上面的 “data” 属性一样，每一次请求都调用该方法，并且将该方法的返回值（字符串）作为API地址，进行访问，此方法比较适合这样的URL：<http://example.com/get_children/1>

7. 用回调函数定义树

如不希望使用jsTree自带的AJAX请求数据，希望自己写AJAX或者其他的方式请求数据，就需要使用回调函数



回调函数会接收两个参数，当前需要加载的节点以及获取完数据后，结束加载的回调方法，数据格式相同，用上AJAX后也有lazy load的效果

8. jstree 事件

Jstree提供了许多事件，可以观察节点树发生了什么变化，不论用什么方式定义树，包含的事件都是一样的，例如changed事件，该事件在选项改变的时候触发：



所有的jstree事件都有一个叫 “.jstree” 的命名空间，比如我们监听的 “changed.jstree”,另外，事件回调处理函数多了一个参数，这个参数包含了该事件的所有信息，比如当前这个例子， data.selected是一个数组，它包含了所有选中的节点id（如果没有自定义id，这些id是自动生成的）

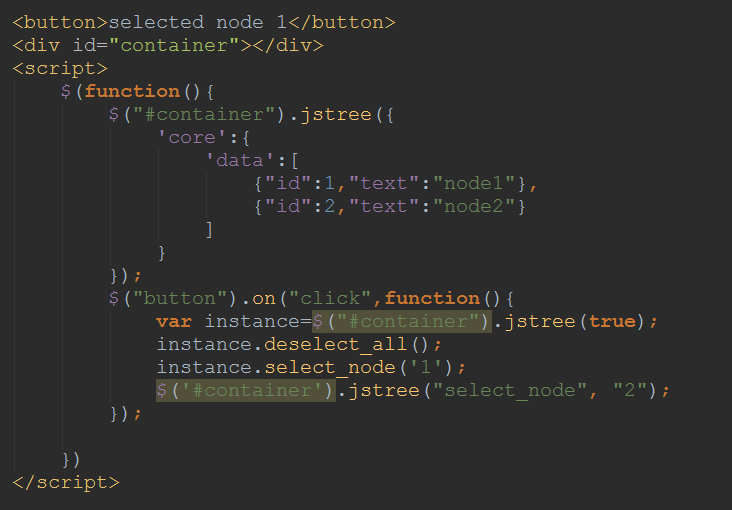


此两行代码都是获取并打印第一个选中节点的text属性

为了方便 jstree的方法，data参数对象中有一个instance的属性，这个是属性代表了这棵树的实例对象

9. jstree API

获取jstree的实例，并调用实例的方法



例子中说明了怎么获取jstree的实例（还是用选择器，只不过参数变成了布尔值true）以及调用了两个方法，最后一个方法是选中id=1的节点

也可以这样调用方法：



10. 更多配置

之前定义树的时候，已经覆盖了大部分的配置场景

$(“#tree”).jstree({/\* config object goes here \*/})

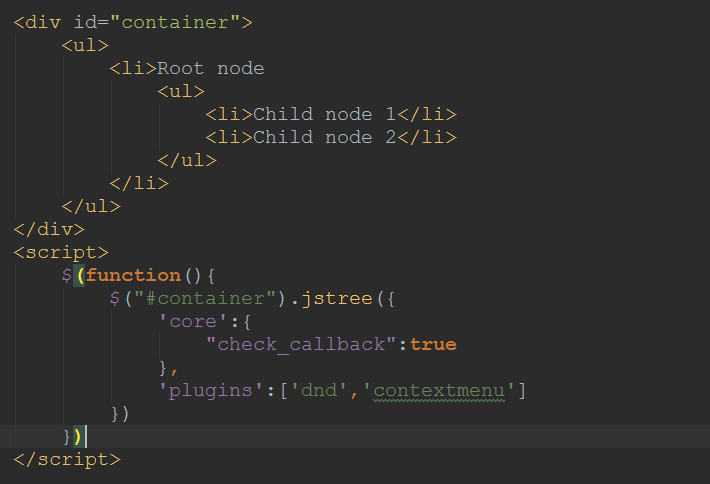
config对象的每一个属性，都对应着一种插件，而属性的值的配置，是专门给该插件提供的，此外还有两个别的属性 ”core” 和 ”plugins”:

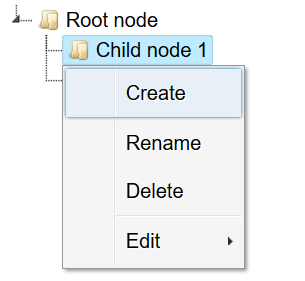
Core 存放着核心的配置

Plugins 是一个数组，包含了当前要激活的各个插件的字符串名称



11. 默认情况下，所有的修改操作是不启用的，像拖放、新增、修改、删除等，需要配置启用

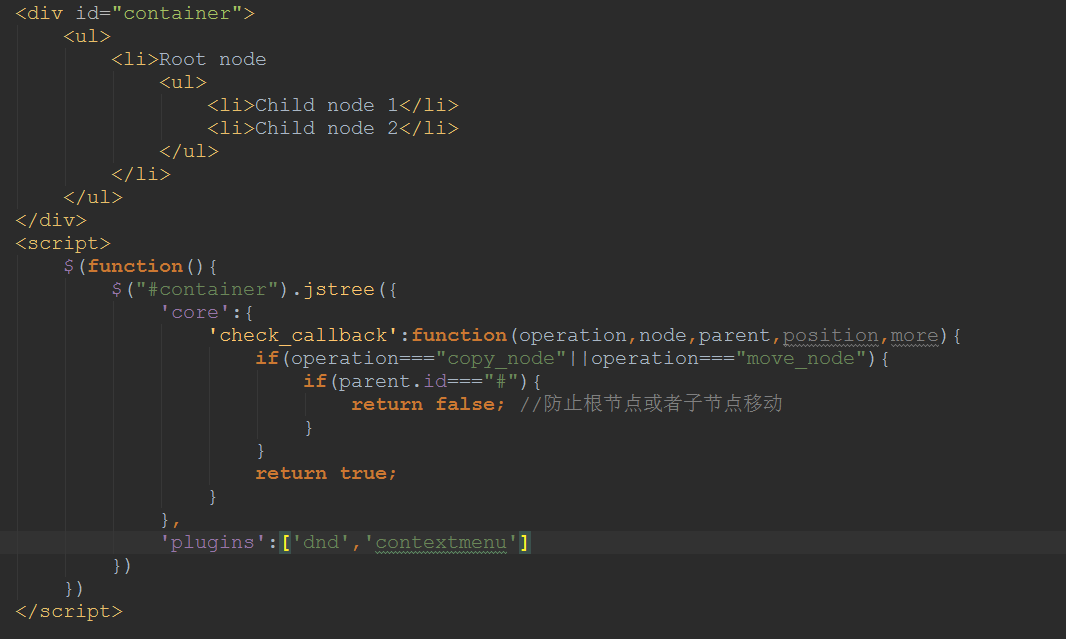




core.check\_callback 也可以设置成一个函数，每次尝试修改都会触发该函数，函数的返回值为 true 时，才允许修改，返回false意味着拒绝修改

operation（操作）参数有 create\_node,rename\_node,delete\_node,move\_node和copy\_node几种

more参数由触发修改确认函数的插件所决定。比如DND插件的more是一个包含move或者copy操作（前、后、操作中）的信息，像是否多选，移动到哪个节点的上方，插入点在哪



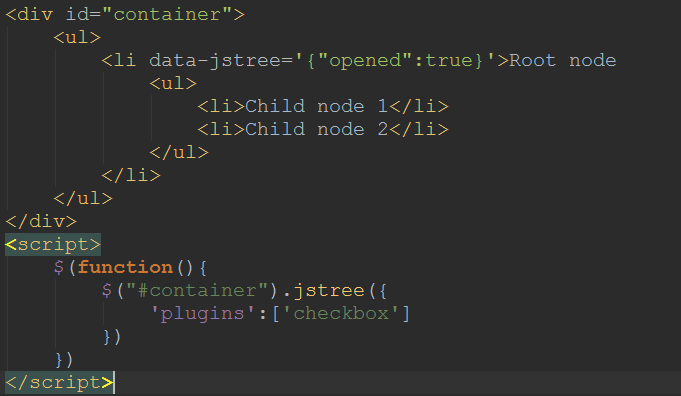
more参数还包含了确认操作相关的其他信息，例如：

在拖拽节点的时候move\_node 和 copy\_node 操作会重复触发，假如确认操作是dnd插件触发的，more会包含一个dnd属性，其值会被设置为true

可以通过检查more.dnd属性，来判断是不是dnd触发了确认操作

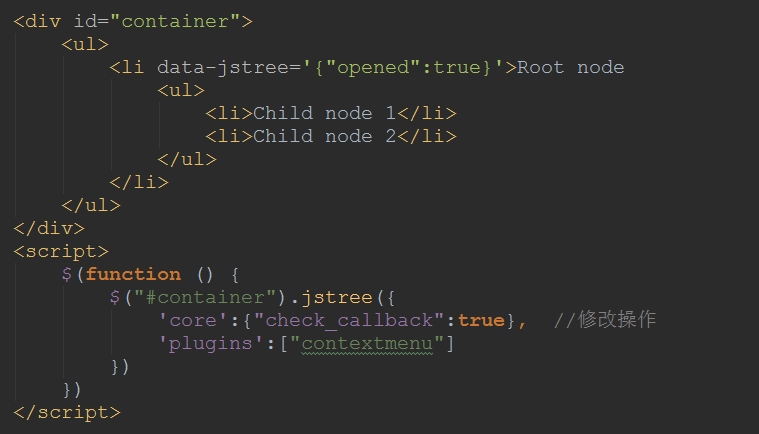
12. checkbox

在每个节点前渲染一个checkbox的图标，用来跟好的标记多选。插件提供了 tri-state 的选项，当该节点的子节点只有部分被选中的时候，会变成一个 square 方格图标



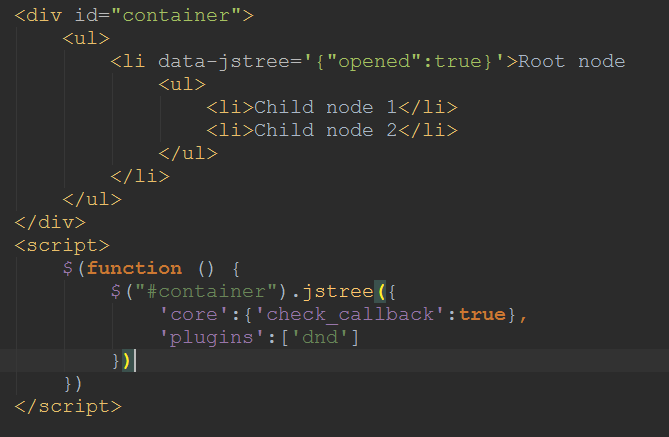
13. contextmenu

当点击鼠标右键时，显示右键菜单，菜单中的操作都是可以配置的



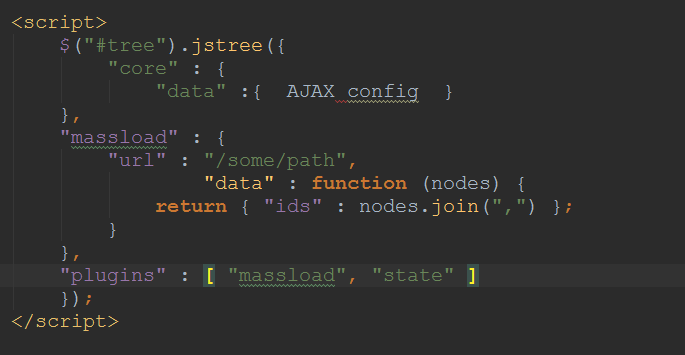
14. dnd

提供拖放功能，重新整理树节点



15. massload

Lazy loaded的一种，为了将一个AJAX中加载的数据放到多个节点。只用core.data时，AJAX每次加载的数据都只放在一个节点下



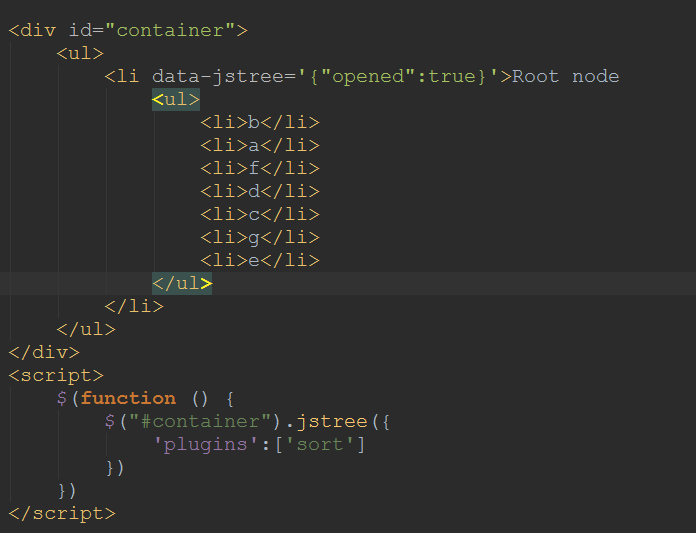
16. search

搜索功能，并只显示匹配到的节点。因为可以使用AJAX/callback构造，所以搜索可以支持lazy loaded的节点



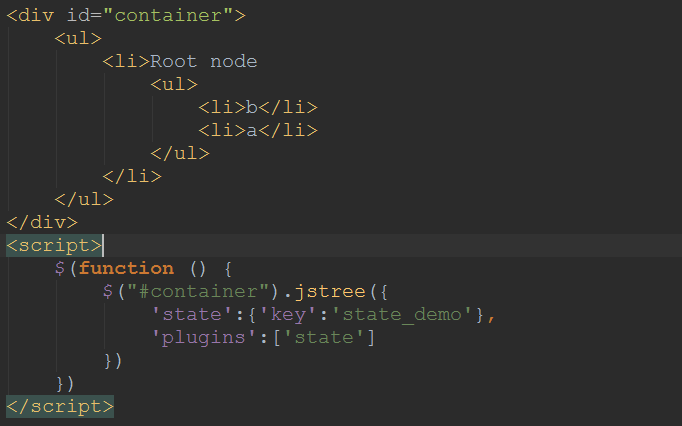
17. sort

节点排序功能，可以在配置中提供自定义的排序方法或者默认启用字母表排序



18. state

保存用户所浏览节点的状态，便于切换回相同的树时，恢复选中的状态



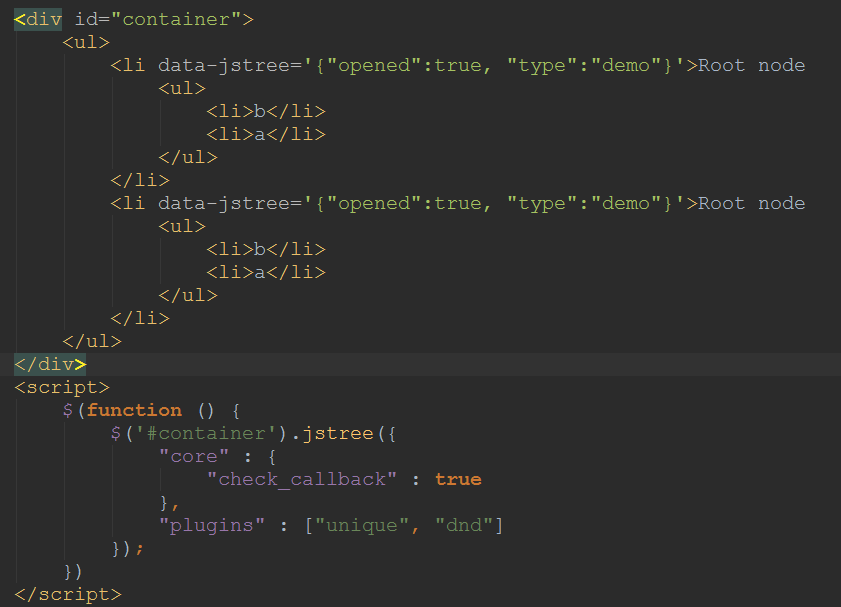
19. types

给节点增加一个类型。主要是为了便于控制和维护相同的type节点的配置。节点设置type属性时，匹配types对象相同的属性



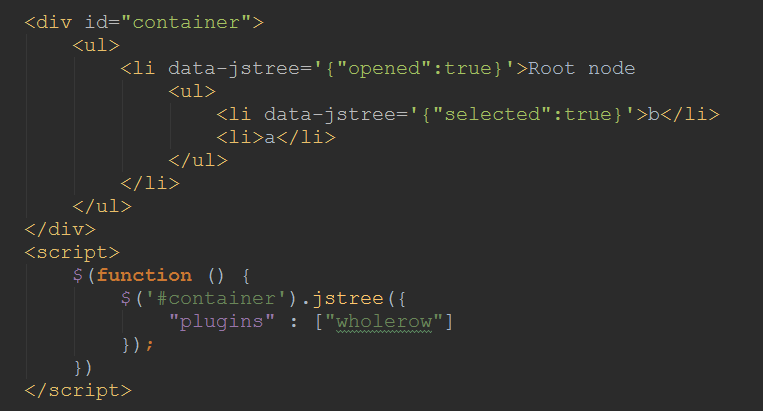
20. unique

命名唯一，不允许在同一级下有命名相同的节点，当移到到parent节点下，存在同名的节点时，不允许移到



21. wholerow

把每个节点作为一行，便于选中。节点过多时，在旧的浏览器上会有性能问题



22. 其他插件 ----- 不懂

如果需要自定义插件或者下载第三方插件，需要在响应的页面引入该资源文件，并且在 plugins 属性中激活



当新建了一个插件时，可以定义一个默认的配置，给jstree增加你的方法，或者覆盖现有的方法

23. PHP demos

[https://github.com/vakata/jstree-php-demos](https://github.com/vakata/jstree-php-demos" \t "http://blog.csdn.net/qq_24472595/article/details/_blank)

不要直接编辑 dist 目录下的文件，他们是grunt生成的。相应的源文件可以在 src 目录找到