

🔗 18. gui.js库(可视化改变三维场景)

gui.js库(可视化改变三维场景)

dat.gui.js说白了就是一个前端js库，对HTML、CSS和JavaScript进行了封装，学习开发的时候，借助dat.gui.js可以快速创建控制三维场景的UI交互界面，你打开课件中案例源码体验一下就能感受到。

课程学习dat.gui.js也不仅仅是为了学习dat.gui.js，也是建立一种思想，就是threejs三维空间的很多参数，不是心算出来的，往往需要可视化的方式调试出来。

引入dat.gui.js

github地址：<https://github.com/dataarts/dat.gui>

npm地址：<https://www.npmjs.com/package/dat.gui>

你可以通过npm或github方式获得dat.gui.js库，当然为了学习方便，threejs官方案例扩展库中也提供了gui.js，你可以直接使用。

```
// 引入dat.gui.js的一个类GUIjs  
import { GUI } from 'three/addons/libs/lil-gui.module.min.js';
```

创建一个GUI对象

创建一个GUI对象，你可以看到浏览器右上角多了一个交互界面，GUI本质上就是一个前端js库。

```
// 实例化一个gui对象js  
const gui = new GUI();
```

.domElement：改变GUI界面默认的style属性

通过 `.domElement` 属性可以获取gui界面的HTML元素，那就意味着你可以改变默认的style样式，比如位置、宽度等。

```
//改变交互界面style属性
gui.domElement.style.right = '0px';
gui.domElement.style.width = '300px';
```

js

`.add()` 方法

执行gui的 `.add()` 方法可以快速创建一个UI交互界面，比如一个拖动条，可以用来改变一个JavaScript对象属性的属性值。

格式： `.add(控制对象, 对象具体属性, 其他参数)`

其他参数，可以一个或多个，数据类型也可以不同，gui会自动根据参数形式，自动生成对应的交互界面。

参数3和参数4，分别是一个**数字**，交互界面是一个鼠标可以拖动的**拖动条**，可以在一个区间改变属性的值

执行 `gui.add(obj, 'x', 0, 100)`；你可以发现右上角gui界面增加了新的内容，可以控制obj对象x属性的新交互界面。

```
//创建一个对象，对象属性的值可以被GUI库创建的交互界面改变
const obj = {
  x: 30,
};
// gui增加交互界面，用来改变obj对应属性
gui.add(obj, 'x', 0, 100);
```

js

体验 `.add()` 功能——改变对象属性值

为了方便观察 `.add()` 是如何改变JavaScript对象属性的，可以浏览器控制台不停地打印obj的值，这样通过gui界面拖动改变obj对象属性的的时候，便于观察obj的变化。

```
const obj = {x: 30};
setInterval(function () {
  console.log('x', obj.x);
}, 10)
```

js

gui改变js对象多个属性

```
const obj = {  
  x: 30,  
  y: 60,  
  z: 300,  
};  
// gui界面上增加交互界面, 改变obj对应属性  
gui.add(obj, 'x', 0, 100);  
gui.add(obj, 'y', 0, 50);  
gui.add(obj, 'z', 0, 60);
```

js

gui改变threejs光照强度测试

three.js在调试场景渲染效果的时候, 比如光照的强度, 人大脑的CPU是没有能力通过光照参数算出来模型渲染效果的, 一般来说你先大概给一个经验值, 然后通过gui在这个大概值的基础上上下浮动可视化调试。

光源对象具有一个光照强度属性 `.intensity` , 可以通过gui拖动条改变该属性。

```
// 光照强度属性.intensity  
console.log('ambient.intensity', ambient.intensity);  
// 通过GUI改变mesh.position对象的xyz属性  
gui.add(ambient, 'intensity', 0, 2.0);
```

js

gui改变threejs模型位置测试

`mesh.position` 是JavaScript对象, 具有 `x` 、 `y` 、 `z` 属性, 这三个属性分别表示模型的xyz坐标, 这就是说, gui改变 `mesh.position` 的 `x` 、 `y` 、 `z` 属性, 就可以可视化改变mesh的位置。

```
gui.add(mesh.position, 'x', 0, 180);  
gui.add(mesh.position, 'y', 0, 180);  
gui.add(mesh.position, 'z', 0, 180);
```

js

