

21. gui.js库(分组)

当GUI交互界面需要控制的属性比较多的时候，为了避免混合，可以适当分组管理，这样更清晰。

gui交互界面不分组

gui交互界面不分组,只有一个默认的总的菜单。

```
const gui = new GUI(); //创建GUI对象
//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 = {
    color: 0x00ffff, // 材质颜色
    specular: 0x111111, // 材质高光颜色
};

// 材质颜色color
gui.addColor(obj, 'color').onChange(function(value){
    material.color.set(value);
});
// 材质高光颜色specular
gui.addColor(obj, 'specular').onChange(function(value){
    material.specular.set(value);
});

// 环境光强度
gui.add(ambient, 'intensity', 0, 2);
// 平行光强度
gui.add(directionalLight, 'intensity', 0, 2);
// 平行光位置
gui.add(directionalLight.position, 'x', -400, 400);
gui.add(directionalLight.position, 'y', -400, 400);
gui.add(directionalLight.position, 'z', -400, 400);
```

js

`.addFolder()` 分组

`new GUI()` 实例化一个gui对象，默认创建一个总的菜单，通过gui对象的 `.addFolder()` 方法可以创建一个子菜单，当你通过GUI控制的属性比较多时，可以使用 `.addFolder()` 进行分组。

`.addFolder()` 返回的子文件夹对象，同样具有gui对象的 `.add()`、`.onChange()`、`.addColor()` 等等属性。

```
const gui = new GUI(); //创建GUI对象
const obj = {
    color: 0x00ffff, // 材质颜色
};
// 创建材质子菜单
const matFolder = gui.addFolder('材质');
matFolder.close();
// 材质颜色color
matFolder.addColor(obj, 'color').onChange(function(value){
    material.color.set(value);
});
// 材质高光颜色specular
matFolder.addColor(obj, 'specular').onChange(function(value){
    material.specular.set(value);
});
```

js

```
// 环境光子菜单
const ambientFolder = gui.addFolder('环境光');
// 环境光强度
ambientFolder.add(ambient, 'intensity', 0, 2);
```

js

```
// 平行光强度
dirFolder.add(directionalLight, 'intensity', 0, 2);
// 平行光位置
dirFolder.add(directionalLight.position, 'x', -400, 400);
dirFolder.add(directionalLight.position, 'y', -400, 400);
dirFolder.add(directionalLight.position, 'z', -400, 400);
```

js

关闭 `.close()` 和展开 `.open()` 交互界面

gui对象创建的总菜单或 `gui.addFolder()` 创建的子菜单都可以用代码控制交互界面关闭或展开状态。

```
const gui = new GUI(); //创建GUI对象
gui.close(); //关闭菜单
```

js

```
// 创建材质子菜单
const matFolder = gui.addFolder('材质');
matFolder.close(); //关闭菜单
```

js

`.addFolder()` 套娃——子菜单嵌套子菜单

`.addFolder()` 创建的对象，同样也具有 `.addFolder()` 属性，可以继续嵌套子菜单。

```
// 平行光子菜单
const dirFolder = gui.addFolder('平行光');
dirFolder.close(); //关闭菜单
// 平行光强度
dirFolder.add(directionalLight, 'intensity', 0, 2);
const dirFolder2 = dirFolder.addFolder('位置'); //子菜单的子菜单
dirFolder2.close(); //关闭菜单
// 平行光位置
dirFolder2.add(directionalLight.position, 'x', -400, 400);
dirFolder2.add(directionalLight.position, 'y', -400, 400);
dirFolder2.add(directionalLight.position, 'z', -400, 400);
```

js

