

🎬 2. 动画播放(暂停、倍速、循环)

上节课对关键帧动画如何**创建**，如何**播放**，做了整体介绍，下面进一步介绍关键帧动画播放的知识，比如关键帧动画停止播放、暂停播放、倍速播放...

动画动作对象 `AnimationAction`

查看文档你可以知道，执行播放器 `AnimationMixer` 的 `.clipAction()` 方法会返回一个 `AnimationAction` 对象。

```
//AnimationMixer的`.clipAction()`返回一个AnimationAction对象
const clipAction = mixer.clipAction(clip);
```

js

`AnimationAction` 对象的功能就是用来控制如何播放关键帧动画，比如是否播放、几倍速播放、是否循环播放、是否暂停播放...

你可以测试注释代码 `clipAction.play();`，看看关键帧动画是否变化。

```
//.play()控制动画播放
clipAction.play();
```

js

`AnimationAction` 的循环属性 `.loop`

通过 `AnimationAction` 的循环属性 `.loop` 可以控制动画是否循环播放。

```
const clipAction = mixer.clipAction(clip);
//.play()控制动画播放，默认循环播放
clipAction.play();
//不循环播放
clipAction.loop = THREE.LoopOnce;
```

js

`AnimationAction` 的 `.clampWhenFinished` 属性

当你通过 `clipAction.loop = THREE.LoopOnce` 设置播放模式为非循环模式的时候，你会发现关键帧动画执行完成一个后，模型回到了关键帧动画开头状态，如果你希望模型停留在关键帧动画结束的状态，可以设置 `.clampWhenFinished` 属性实现，`.clampWhenFinished` 属性默认是 `false`，设置为 `true` 即可

```
//不循环播放，执行一次后默认回到动画开头
clipAction.loop = THREE.LoopOnce;
```

js

```
// 物体状态停留在动画结束的时候
clipAction.clampWhenFinished = true;
```

js

停止结束动画 `.stop()`

执行AnimationAction的 `.stop()` 方法，动画会停止，并结束，模型回到动画开始状态,注意不是暂停，是动画彻底终止，回到初始状态。

```
//动画停止结束，回到开始状态
clipAction.stop();
```

js

```
<div class="pos">
  <div id="stop" class="bu">停止</div>
  <div id="play" class="bu" style="margin-left: 10px;">播放</div>
</div>
```

html

```
document.getElementById('stop').addEventListener('click',function(){
  clipAction.stop();//动画停止结束，回到开始状态
})
document.getElementById('play').addEventListener('click',function(){
  clipAction.play();//播放动画
})
```

js

是否暂停播放 `.paused`

`.paused` 默认值false，动画正常执行，如果你想暂停正在执行的动画可以把 `.paused` 设置为true，对于暂停执行的动画，你把 `.paused` 设置为false，动画会接着暂停的位置继续执行。

```
<div id="bu" class="bu">暂停</div>
```

html

```
const bu = document.getElementById('bu');
bu.addEventListener('click',function(){
  // AnimationAction.paused默认值false，设置为true，可以临时暂停动画
  if (clipAction.paused) { // 暂停状态
    clipAction.paused = false; // 切换为播放状态
    bu.innerHTML='暂停'; // 如果改变为播放状态，按钮文字设置为“暂停”
  } else { // 播放状态
    clipAction.paused = true; // 切换为暂停状态
    bu.innerHTML='继续'; // 如果改变为暂停状态，按钮文字设置为“继续”
  }
})
```

js

倍速播放 `.timeScale`

```
clipAction.timeScale = 1; // 默认
clipAction.timeScale = 2; // 2倍速
```

js

拖动条调整播放速度

gui辅助快速创建一个可交互拖动条，实际开发，可以用vue或react创建的拖动条。

```
const gui = new GUI(); // 创建GUI对象
// 0~6倍速之间调节
gui.add(clipAction, 'timeScale', 0, 6);
```

js

← 1. 关键帧动画

3. 动画播放(拖动任意时间状态) →

