

相机

`v2.10.x` 版本之后，相机所有的方法都基于 `controls` 实现。

getCameraViewpoint

获取当前相机视角数据

定义：

```
function getCameraViewpoint(): CameraViewpointData;
```

ts

用法：

```
const cameraViewpointData = ssp.getCameraViewpoint();  
console.log('cameraViewpointData', cameraViewpointData);
```

js

setCameraViewpoint

设置当前相机视角数据，数据由 `getCameraViewpoint` 获取。

定义：

```
// v2.9.x  
function setCameraViewpoint(data: CameraViewpointData): void;
```

ts

☰ SoonSpace.js 2.x

用法

```
// v2.9.x
```

js

```
ssp.setCameraViewpoint({
```

```
  position: {
    x: 100,
    y: 100,
    z: 100,
  },
  rotation: {
    x: 0,
    y: Math.PI / 2,
    z: 0,
  },
});
```

```
// v2.10.x
```

```
await ssp.setCameraViewpoint(
  {
    position: {
      x: 100,
      y: 100,
      z: 100,
    },
    target: {
      x: 0,
      y: 0,
      z: 0,
    },
  },
  true
);
```

参数：

data

- **描述:** 由 `getCameraViewpoint` 获取到的相机视角数据。

☰ SoonSpace.js 2.x

CameraViewpointData

属性	描述	类型	必填	默认值
<code>position</code>	相机位置	<code>Position</code>	✓	
<code>rotation</code>	相机旋转弧度	<code>Rotation</code>	✓	

`v2.10.x` 版本之后数据格式发生变化，但也兼容了旧版的 `rotation` 参数

属性	描述	类型	必填	默认值
<code>position</code>	相机位置	<code>Position</code>	✓	
<code>target</code>	相机朝向位置	<code>Position</code>	✓	

enableTransition

- 描述: 是否开启过渡效果，默认值为 `false`
- 必填: ✗
- 类型: `boolean`

提示

也可以直接使用 `setLookAt` 方法

flyMainViewpoint

相机飞向主场景视角

定义:

```
function flyMainViewpoint(viewpoint: FlyToViewpoint = 'frontTop', options: FlyToViewpointOptionsts)
```

☰ SoonSpace.js 2.x

用法:

```
ssp
  .flyMainViewpoint('top')
  .then((object) => console.log('flyMainViewpoint done', object))
  .catch((err) => console.error(err));
```

js

参数:

viewpoint

- 描述: 相机朝向主场景的视角面
- 必填: **✗**
- 类型: **FlyToViewpoint**
- 默认值: `frontTop`

options

- 描述: 可配置参数
- 必填: **✗**
- 类型: **FlyToObjOptions**

flyToObj

相机飞向对象

样例:

☰ SoonSpace.js 2.x



定义:

```
interface FlyToObjOptions extends AnimationOptions {  
  padding?: number | string;  
}  
  
function flyToObj(object: BaseObject3D | BaseMesh, viewpoint: FlyToViewpoint
```

v2.10.x 版本之后更新了类型

```
interface FlyToObjOptions {  
  enableTransition?: boolean;  
  padding?: number | string;  
}  
  
function flyToObj(object: Object3D | Box3, viewpoint?: FlyToViewpoint, option
```

用法:

☰ SoonSpace.js 2.x

```
    .flyToObj(  
      // object  
      sbm,  
      // viewpoint  
      'top',  
      // option  
      {  
        padding: '30%',  
        duration: 1000,  
      }  
    )  
    .then((object) => console.log('flyToObj done', object))  
    .catch((err) => console.error(err));  
  
// v2.10.x  
ssp.flyToObj(sbm, 'left', { padding: '30%', enableTransition: false });
```

参数：

object

- 描述: 相机飞向的空间对象
- 必填: ✓
- 类型: `Object3D` | `Box3`

viewpoint

- 描述: 相机朝向物体对象的视角面，**可选枚举**。
- 必填: ✗
- 类型: `FlyToViewpoint`
- 默认值: `frontTop`

option

- 描述: 可配置项
- 必填: ✗
- 类型: `FlyToObjOptions`

FlyToObjOptions

☰ SoonSpace.js 2.x

属性	描述	类型	必填	默认值
padding	视角后飞向的偏移量	number string	✗	30%
duration	补间执行时长 (ms)	number	✗	1000
delay	补间开始前延时 (ms)	number	✗	0
repeat	循环	number boolean	✗	false

v2.10.x

属性	描述	类型	必填	默认值
padding	视角后飞向的偏移量	number string	✗	30%
enableTransition	是否开启过渡效果	boolean	✗	true

提示

内部是调用 `controls.fitToBox` 方法

flyTo

相机飞向固定位置

样例：

飞向 Poi
飞向模型

Close Controls



定义:

```
function flyTo(position: Position, rotation: FlyToViewpoint | Rotation | Euler, options?: FlyToOptions) {
```

v2.10.x 版本之后更新了类型

```
interface FlyToOptions {
  enableTransition?: boolean;
}

function flyTo(position: Position, rotation: FlyToViewpoint | Rotation | Euler, options?: FlyToOptions) {
```

用法:

```
// v2.9.x
ssp.flyTo(
  // position
```


☰ SoonSpace.js 2.x

```
        y: 1000,  
        z: 1000  
    },  
    // rotation  
    {  
        x: 0,  
        y: 0,  
        z: 0  
    },  
    // option  
    {  
        duration: 1000  
    }  
)  
    .then(res => console.log(res))  
    .catch(err => console.error(err))  
  
// v2.10.x  
await ssp.flyTo(  
    // position  
    {  
        x: 0,  
        y: 1000,  
        z: 1000  
    },  
    // rotation  
    {  
        x: 0,  
        y: 0,  
        z: 0  
    },  
    // option  
    {  
        enableTransition: true  
    }  
)
```

参数:

position

- **描述:** 相机飞向的坐标点

☰ SoonSpace.js 2.x

rotation

- 描述: 相机的旋转弧度
- 必填: **✗**
- 类型: **FlyToViewpoint** | **Rotation**
- 默认值: `frontTop`

options

v2.9.x

- 描述: 可配置项
- 类型: **AnimationOptions**
- 必填: **✗**
- 默认值: `{}`

v2.10.x

- 描述: 可配置项
- 类型: `FlyToOptions`
- 必填: **✗**
- 默认值: `{ enableTransition: true }`

注意

`flyTo` 以向后兼容保留, 你也可以使用 `setCameraViewpoint` 替代

surroundOnTarget

围绕一个目标点旋转

样例:

环绕暂停
环绕继续

Close Controls

objects 29
triangles 28734
vertices 86202
frametime 11.00 ms



定义:

```
interface SurroundOptions {  
  duration?: number;  
  startAngle?: number;  
  endAngle?: number;  
  end?: number;  
  onStart?: (tween: Tween<{ radian: number }>) => void;  
}  
  
function surroundOnTarget(target: Position, options: SurroundOptions = {}): P
```

用法:

```
ssp  
  .surroundOnTarget(  
    // target  
    {  
      x: 0,  
      y: 0,  
      z: 0,
```

☰ SoonSpace.js 2.x


```
    startAngle: 0,
    endAngle: 360,
    duration: 2000,
    onStart: (tween) => {
      console.log('旋转动画对象', tween);
    },
  },
}
)
.then(() => console.log('surroundOnTarget done'))
.catch((err) => console.error(err));
```

参数:

target

- 描述: 目标点
- 类型: **Position**
- 必填: 

options

- 描述: 可配置项
- 必填: 
- 类型: SurroundOptions
- 默认值: `{}`

SurroundOptions

属性	描述	类型	必填	默认值
startAngle	起始旋转角度	number		0
endAngle	结束旋转角度	number		360
duration	旋转动画过渡时间	number		3000
onStart	旋转开始回调函数	function(tween){}		

☰ SoonSpace.js 2.x

surroundOnObject

围绕一个目标对象旋转

定义:

```
function surroundOnObject(object: BaseObject3D | BaseMesh, options: SurroundOptions) {}
```

用法:

```
ssp
  .surroundOnObject(
    // object
    ssp.getObjectById('xxx'),
    // option
    {
      startAngle: 0,
      endAngle: 360,
      duration: 2000,
      onStart: (tween) => {
        console.log('旋转动画对象', tween);
      },
    }
  )
  .then(() => console.log('surroundOnObject done'))
  .catch((err) => console.error(err));
```

参数:

object

- 描述: 目标对象

☰ SoonSpace.js 2.x

options

- 描述: 可配置项
- 必填: ✖
- 类型: SurroundOptions
- 默认值: `{}`

← 场景操作

灯光 →