郭隆邦 2023-06-05

1. 向量点乘dot

百科词条你可以看到关于点乘 的介绍，不过这些理论不要求你掌握，换句话说，你有没有相关的数学基础，都不影响本节课内容的学习。

点乘是向量的一种运算规则，点乘也有其它称呼，比如点积、数量积、标量积。

threejs三维向量`Vector3`封装了一个**点乘**相关的方法`.dot()`，本节课主要目的就是让大家能够灵活应用**点乘**方法`.dot()`。

已知向量a和向量b

已知两个向量a和b，默认夹角是45度。

```js
const a = new THREE.Vector3(10, 10, 0);
const b = new THREE.Vector3(20, 0, 0);
```

向量点乘`.dot()`语法

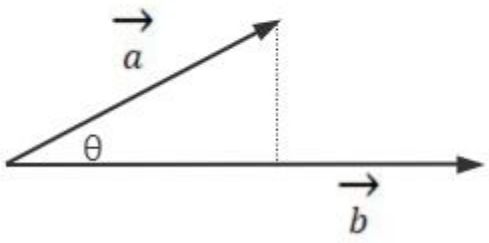
下面先给大家说下点乘`.dot()`的语法，然后再讲解它的用途。

`a.dot(b)`表示向量`a`与向量`b`点乘，返回结果是一个数字(标量)。

```js
//向量a与向量b点乘，返回结果是一个数字
const dot = a.dot(b);
console.log('点乘结果',dot);
```

点乘`.dot()`几何含义

点乘 $\vec{a} \bullet \vec{b} = |\vec{a}||\vec{b}| \cos\theta$



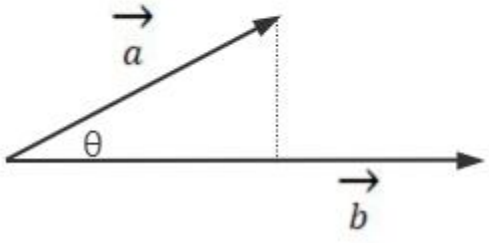
你只需要记住 `a.dot(b)` 的几何含义是向量a在向量b上投影长度与向量b相乘，或者说 `向量a长度 * 向量b长度 * cos(ab夹角)` 。

```js
const a = new THREE.Vector3(10, 10, 0);
const b = new THREE.Vector3(20, 0, 0);
// dot几何含义：向量a长度 * 向量b长度 * cos(ab夹角)
const dot = a.dot(b);
console.log('点乘结果',dot);//判断结果是不是200
```

单位向量点乘含义(计算向量夹角余弦值)

假设两个向量的夹角是θ，两个向量的单位向量进行点乘 `.dot()` ，返回的结果就是夹角θ的余弦值 `cos(θ)`

点乘 $\vec{a} \bullet \vec{b} = |\vec{a}||\vec{b}| \cos\theta$




```js
const a = new THREE.Vector3(10, 10, 0);
const b = new THREE.Vector3(20, 0, 0);
// a、b向量归一化后点乘
const cos =  a.normalize().dot(b.normalize());
console.log('向量夹角余弦值',cos);
```

如果不希望向量a和b被改变，注意克隆 `.clone()`

```js
const cos =  a.clone().normalize().dot(b.clone().normalize());
```

夹角余弦值转角度值

```
//反余弦计算向量夹角弧度
const rad = Math.acos(cos);
```

```
// 弧度转角度
const angle = THREE.MathUtils.radToDeg(rad);
console.log('向量夹角角度值',angle);
```

修改向量a和b垂直，验证下，代码计算夹角是否正确。

```js
const a = new THREE.Vector3(0, 10, 0);
const b = new THREE.Vector3(20, 0, 0);
// 打印结果90度
console.log('向量夹角角度值',angle);
```

向量相反方向，夹角180度

```js
const a = new THREE.Vector3(-10, 0, 0);
const b = new THREE.Vector3(20, 0, 0);
```

向量同一个方向，夹角0度。

```js
const a = new THREE.Vector3(10, 0, 0);
const b = new THREE.Vector3(20, 0, 0);
```

← 2. 物体下落动画(重力加速度)　　2. 点乘练习-计算三角形夹角→