

# 过渡动画

主题键使您能够自定义使用跨 Material-UI 组件的各种过渡动画的时长和缓动效果，并且它也提供了一个用于创建自定义过渡动画的工具集。

## API

```
theme.transitions.create(props, options) => transition
```

### 参数

1. `props` (*string* | *string[]*): Defaults to `['all']` . 这将提供一个 CSS 属性，或者提供一个应该具有动画效果的 CSS 属性列表。 这将提供一个 CSS 属性，或者提供一个应该具有动画效果的 CSS 属性列表。
2. `options` (*object* [optional]):
  - `options.easing` (*string* [optional]): 默认为 `theme.transitions.easing.easeInOut` 。 这将为动画提供缓动效果。
  - `options.duration` (*string* | *number* [optional]): Defaults to `theme.transitions.duration.standard` . 这将为动画提供动画效果的时长。 Provides the easing for the transition.
  - `options.delay` (*string* | *number* [optional]): Defaults to `0` . 这将为动画提供延迟效果。 这将为动画提供延迟效果。

### 返回结果

、  
、

```
theme.transitions.create(['background-color', 'transform']);
```

### 示例

```
{["demo": "TransitionHover.js", "defaultCodeOpen": false]}
```

```
theme.transitions.getAutoHeightDuration(height) => duration
```

### 参数

1. `height` (*number*): The height of the component.

### 返回结果

`duration` : 基于高度计算出来的时长。

## 时长

你可以更改其中部分或全部的时长，或者提供你想要的时长（供 `create()` 助手使用）。此示例显示了所有默认值（以毫秒为单位），但你只需要提供你想要更改或添加的键。

```
const theme = createTheme({
  transitions: {
    duration: {
      shortest: 150,
```

```

    shorter: 200,
    short: 250,
    // most basic recommended timing
    standard: 300,
    // this is to be used in complex animations
    complex: 375,
    // recommended when something is entering screen
    enteringScreen: 225,
    // recommended when something is leaving screen
    leavingScreen: 195,
  },
},
});

```

## 缓动

你可以通过提供一个自定义的 CSS `transition-timing-function` 值来改变部分或全部的缓动值，或者提供你自己的缓动值。

```

const theme = createTheme({
  transitions: {
    easing: {
      // This is the most common easing curve.
      easeInOut: 'cubic-bezier(0.4, 0, 0.2, 1)',
      // 物体以全速从屏幕外进入屏幕，并在屏幕上以全速前进。
      easeInOut: 'cubic-bezier(0.4, 0, 0.2, 1)',
      // Objects enter the screen at full velocity from off-screen and
      // slowly decelerate to a resting point.
      easeOut: 'cubic-bezier(0.0, 0, 0.2, 1)',
      // 物体以全速离开屏幕。 它们在屏幕外不会减速。
      easeIn: 'cubic-bezier(0.4, 0, 1, 1)',
      // 锐化曲线是由可能随时返回屏幕的对象使用的。
      sharp: 'cubic-bezier(0.4, 0, 0.6, 1)',
    },
  },
});

```

## 参考

你可以通过提供一个自定义的 CSS `transition-timing-function` 值来改变部分或全部的缓动值，或者提供你自己的缓动值。