

## Loading 加载

加载数据时显示动效。

### 区域加载

在表格等容器中加载数据时显示。

::demo Element 提供了两种调用 Loading 的方法：指令和服务。对于自定义指令 `v-loading`，只需要绑定 `Boolean` 即可。默认状况下，Loading 遮罩会插入到绑定元素的子节点，通过添加 `body` 修饰符，可以使遮罩插入至 DOM 中的 `body` 上。

```
<template>
  <el-table
    v-loading="loading"
    :data="tableData"
    style="width: 100%">
    <el-table-column
      prop="date"
      label="日期"
      width="180">
    </el-table-column>
    <el-table-column
      prop="name"
      label="姓名"
      width="180">
    </el-table-column>
    <el-table-column
      prop="address"
      label="地址">
    </el-table-column>
  </el-table>
</template>

<style>
  body {
    margin: 0;
  }
</style>

<script>
  export default {
    data() {
      return {
        tableData: [{
          date: '2016-05-03',
          name: '王小虎',
          address: '上海市普陀区金沙江路 1518 弄'
        }, {
          date: '2016-05-02',
          name: '王小虎',
          address: '上海市普陀区金沙江路 1518 弄'
        }
      ]
    }
  }
}
```

```

      }, {
        date: '2016-05-04',
        name: '王小虎',
        address: '上海市普陀区金沙江路 1518 弄'
      }],
      loading: true
    };
  }
};
</script>

```

...

## 自定义

可自定义加载文案、图标和背景色。

`::demo` 在绑定了 `v-loading` 指令的元素上添加 `element-loading-text` 属性，其值会被渲染为加载文案，并显示在加载图标的下方。类似地，`element-loading-spinner` 和 `element-loading-background` 属性分别用来设定图标类名和背景色值。

```

<template>
  <el-table
    v-loading="loading"
    element-loading-text="拼命加载中"
    element-loading-spinner="el-icon-loading"
    element-loading-background="rgba(0, 0, 0, 0.8)"
    :data="tableData"
    style="width: 100%">
    <el-table-column
      prop="date"
      label="日期"
      width="180">
    </el-table-column>
    <el-table-column
      prop="name"
      label="姓名"
      width="180">
    </el-table-column>
    <el-table-column
      prop="address"
      label="地址">
    </el-table-column>
  </el-table>
</template>

<script>
  export default {
    data() {
      return {
        tableData: [{
          date: '2016-05-03',

```

```

      name: '王小虎',
      address: '上海市普陀区金沙江路 1518 弄'
    }, {
      date: '2016-05-02',
      name: '王小虎',
      address: '上海市普陀区金沙江路 1518 弄'
    }, {
      date: '2016-05-04',
      name: '王小虎',
      address: '上海市普陀区金沙江路 1518 弄'
    }
  ],
  loading: true
};
}
};
</script>

```

...

## 整页加载

页面数据加载时显示。

:::demo 当使用指令方式时，全屏遮罩需要添加 `fullscreen` 修饰符（遮罩会插入至 `body` 上），此时若需要锁定屏幕的滚动，可以使用 `lock` 修饰符；当使用服务方式时，遮罩默认即为全屏，无需额外设置。

```

<template>
  <el-button
    type="primary"
    @click="openFullScreen1"
    v-loading.fullscreen.lock="fullscreenLoading">
    指令方式
  </el-button>
  <el-button
    type="primary"
    @click="openFullScreen2">
    服务方式
  </el-button>
</template>

<script>
export default {
  data() {
    return {
      fullscreenLoading: false
    }
  },
  methods: {
    openFullScreen1() {
      this.fullscreenLoading = true;
      setTimeout(() => {
        this.fullscreenLoading = false;

```

```

    }, 2000);
  },
  openFullScreen2() {
    const loading = this.$loading({
      lock: true,
      text: 'Loading',
      spinner: 'el-icon-loading',
      background: 'rgba(0, 0, 0, 0.7)'
    });
    setTimeout(() => {
      loading.close();
    }, 2000);
  }
}
}
</script>

```

...

## 服务

Loading 还可以以服务的方式调用。引入 Loading 服务：

```
import { Loading } from 'element-ui';
```

在需要调用时：

```
Loading.service(options);
```

其中 `options` 参数为 Loading 的配置项，具体见下表。`LoadingService` 会返回一个 Loading 实例，可通过调用该实例的 `close` 方法来关闭它：

```

let loadingInstance = Loading.service(options);
this.$nextTick(() => { // 以服务的方式调用的 Loading 需要异步关闭
  loadingInstance.close();
});

```

需要注意的是，以服务的方式调用的全屏 Loading 是单例的：若在前一个全屏 Loading 关闭前再次调用全屏 Loading，并不会创建一个新的 Loading 实例，而是返回现有全屏 Loading 的实例：

```

let loadingInstance1 = Loading.service({ fullscreen: true });
let loadingInstance2 = Loading.service({ fullscreen: true });
console.log(loadingInstance1 === loadingInstance2); // true

```

此时调用它们中任意一个的 `close` 方法都能关闭这个全屏 Loading。

如果完整引入了 Element，那么 `Vue.prototype` 上会有一个全局方法 `$loading`，它的调用方式为：

```
this.$loading(options)，同样会返回一个 Loading 实例。
```

Options

参数	说明	类型	可 选 值	默认值
target	Loading 需要覆盖的 DOM 节点。可传入一个 DOM 对象或字符串；若传入字符串，则会将其作为参数传入 <code>document.querySelector</code> 以获取到对应 DOM 节点	object/string	—	document.body
body	同 <code>v-loading</code> 指令中的 <code>body</code> 修饰符	boolean	—	false
fullscreen	同 <code>v-loading</code> 指令中的 <code>fullscreen</code> 修饰符	boolean	—	true
lock	同 <code>v-loading</code> 指令中的 <code>lock</code> 修饰符	boolean	—	false
text	显示在加载图标下方的加载文案	string	—	—
spinner	自定义加载图标类名	string	—	—
background	遮罩背景色	string	—	—
customClass	Loading 的自定义类名	string	—	—