

Notification 通知

悬浮出现在页面角落，显示全局的通知提醒消息。

基本用法

适用性广泛的通知栏

:::demo Notification 组件提供通知功能，Element 注册了 `$notify` 方法，接收一个 `options` 字面量参数，在最简单的情况下，你可以设置 `title` 字段和 `message` 字段，用于设置通知的标题和正文。默认情况下，经过一段时间后 Notification 组件会自动关闭，但是通过设置 `duration`，可以控制关闭的时间间隔，特别的是，如果设置为 `0`，则不会自动关闭。注意：`duration` 接收一个 `Number`，单位为毫秒，默认为 `4500`。

```
<template>
  <el-button
    plain
    @click="open1">
    可自动关闭
  </el-button>
  <el-button
    plain
    @click="open2">
    不会自动关闭
  </el-button>
</template>

<script>
  export default {
    methods: {
      open1() {
        const h = this.$createElement;

        this.$notify({
          title: '标题名称',
          message: h('i', { style: 'color: teal' }, '这是提示文案这是提示文案这是提示文案这是提示文案这是提示文案这是提示文案这是提示文案')
        });
      },

      open2() {
        this.$notify({
          title: '提示',
          message: '这是一条不会自动关闭的消息',
          duration: 0
        });
      }
    }
  }
</script>
```

...

带有倾向性

带有 icon，常用来显示「成功、警告、消息、错误」类的系统消息

:::demo Element 为 Notification 组件准备了四种通知类型： `success`，`warning`，`info`，`error`。通过 `type` 字段来设置，除此以外的值将被忽略。同时，我们也为 Notification 的各种 type 注册了方法，可以在不传入 `type` 字段的情况下像 `open3` 和 `open4` 那样直接调用。

```
<template>
  <el-button
    plain
    @click="open1">
    成功
  </el-button>
  <el-button
    plain
    @click="open2">
    警告
  </el-button>
  <el-button
    plain
    @click="open3">
    消息
  </el-button>
  <el-button
    plain
    @click="open4">
    错误
  </el-button>
</template>

<script>
export default {
  methods: {
    open1() {
      this.$notify({
        title: '成功',
        message: '这是一条成功的提示消息',
        type: 'success'
      });
    },

    open2() {
      this.$notify({
        title: '警告',
        message: '这是一条警告的提示消息',
        type: 'warning'
      });
    },

    open3() {
      this.$notify.info({
```

```

        title: '消息',
        message: '这是一条消息的提示消息'
    });
},

open4() {
    this.$notify.error({
        title: '错误',
        message: '这是一条错误的提示消息'
    });
}
}
}
}
</script>

```

...

自定义弹出位置

可以让 Notification 从屏幕四角中的任意一角弹出

demo 使用 position 属性定义 Notification 的弹出位置，支持四个选项：top-right、top-left、bottom-right、bottom-left，默认为 top-right。

```

<template>
  <el-button
    plain
    @click="open1">
    右上角
  </el-button>
  <el-button
    plain
    @click="open2">
    右下角
  </el-button>
  <el-button
    plain
    @click="open3">
    左下角
  </el-button>
  <el-button
    plain
    @click="open4">
    左上角
  </el-button>
</template>

<script>
  export default {
    methods: {
      open1() {
        this.$notify({

```

```

        title: '自定义位置',
        message: '右上角弹出的消息'
    });
},

open2() {
    this.$notify({
        title: '自定义位置',
        message: '右下角弹出的消息',
        position: 'bottom-right'
    });
},

open3() {
    this.$notify({
        title: '自定义位置',
        message: '左下角弹出的消息',
        position: 'bottom-left'
    });
},

open4() {
    this.$notify({
        title: '自定义位置',
        message: '左上角弹出的消息',
        position: 'top-left'
    });
}
}
}
</script>

```

...

带有偏移

让 Notification 偏移一些位置

:::demo Notification 提供设置偏移量的功能，通过设置 `offset` 字段，可以使弹出的消息距屏幕边缘偏移一段距离。注意在同一时刻，所有的 Notification 实例应当具有一个相同的偏移量。

```

<template>
  <el-button
    plain
    @click="open">
    偏移的消息
  </el-button>
</template>

<script>
  export default {
    methods: {

```

```

    open() {
      this.$notify({
        title: '偏移',
        message: '这是一条带有偏移的提示消息',
        offset: 100
      });
    }
  }
}
</script>

```

...

使用 HTML 片段

`message` 属性支持传入 HTML 片段

:::demo 将 `dangerouslyUseHTMLString` 属性设置为 `true`, `message` 就会被当作 HTML 片段处理。

```

<template>
  <el-button
    plain
    @click="open">
    使用 HTML 片段
  </el-button>
</template>

<script>
  export default {
    methods: {
      open() {
        this.$notify({
          title: 'HTML 片段',
          dangerouslyUseHTMLString: true,
          message: '<strong>这是 <i>HTML</i> 片段</strong>'
        });
      }
    }
  }
}
</script>

```

...

:::warning `message` 属性虽然支持传入 HTML 片段，但是在网站上动态渲染任意 HTML 是非常危险的，因为容易导致 [XSS 攻击](#)。因此在 `dangerouslyUseHTMLString` 打开的情况下，请确保 `message` 的内容是可信的，**永远不要**将用户提交的内容赋值给 `message` 属性。:::

隐藏关闭按钮

可以不显示关闭按钮

:::demo 将 `showClose` 属性设置为 `false` 即可隐藏关闭按钮。

```
<template>
  <el-button
    plain
    @click="open">
    隐藏关闭按钮
  </el-button>
</template>

<script>
  export default {
    methods: {
      open() {
        this.$notify.success({
          title: 'Info',
          message: '这是一条没有关闭按钮的消息',
          showClose: false
        });
      }
    }
  }
</script>
```

...

全局方法

Element 为 `Vue.prototype` 添加了全局方法 `$notify` 。因此在 vue instance 中可以采用本页面中的方式调用 `Notification`。

单独引用

单独引入 `Notification`：

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

此时调用方法为 `Notification(options)` 。我们也为每个 type 定义了各自的方法，如 `Notification.success(options)` 。并且可以调用 `Notification.closeAll()` 手动关闭所有实例。

Options

参数	说明	类型	可选值	默认值
title	标题	string	—	—
message	说明文字	string/Vue.VNode	—	—
dangerouslyUseHTMLString	是否将 message 属性作为 HTML 片段处理	boolean	—	false

type	主题样式，如果不在可选值内将被忽略	string	success/warning/info/error	—
iconClass	自定义图标的类名。若设置了 type，则 iconClass 会被覆盖	string	—	—
customClass	自定义类名	string	—	—
duration	显示时间，毫秒。设为 0 则不会自动关闭	number	—	4500
position	自定义弹出位置	string	top-right/top-left/bottom-right/bottom-left	top-right
showClose	是否显示关闭按钮	boolean	—	true
onClose	关闭时的回调函数	function	—	—
onClick	点击 Notification 时的回调函数	function	—	—
offset	偏移的距离，在同一时刻，所有的 Notification 实例应当具有一个相同的偏移量	number	—	0

方法

调用 `Notification` 或 `this.$notify` 会返回当前 `Notification` 的实例。如果需要手动关闭实例，可以调用它的 `close` 方法。

方法名	说明
close	关闭当前的 Notification