zh-CN

使用 `confirm()` 可以快捷地弹出确认框。onCancel/onOk 返回 promise 可以延迟关闭。

en-US

Use `confirm()` to show a confirmation modal dialog. Let onCancel/onOk function return a promise object to delay closing the dialog.

```jsx
import { Modal, Button, Space } from 'antd';
import { ExclamationCircleOutlined } from '@ant-design/icons';

const { confirm } = Modal;

function showConfirm() {
  confirm({
    title: 'Do you Want to delete these items?',
    icon: <ExclamationCircleOutlined />,
    content: 'Some descriptions',
    onOk() {
      console.log('OK');
    },
    onCancel() {
      console.log('Cancel');
    },
  });
}

function showPromiseConfirm() {
  confirm({
    title: 'Do you want to delete these items?',
    icon: <ExclamationCircleOutlined />,
    content: 'When clicked the OK button, this dialog will be closed after 1
second',
    onOk() {
      return new Promise((resolve, reject) => {
        setTimeout(Math.random() > 0.5 ? resolve : reject, 1000);
      }).catch(() => console.log('Oops errors!'));
    },
    onCancel() {},
  });
}

function showDeleteConfirm() {
  confirm({
    title: 'Are you sure delete this task?',
    icon: <ExclamationCircleOutlined />,
    content: 'Some descriptions',
    okText: 'Yes',
    okType: 'danger',
    cancelText: 'No',
```

```
    onOk() {
      console.log('OK');
    },
    onCancel() {
      console.log('Cancel');
    },
  });
}

function showPropsConfirm() {
  confirm({
    title: 'Are you sure delete this task?',
    icon: <ExclamationCircleOutlined />,
    content: 'Some descriptions',
    okText: 'Yes',
    okType: 'danger',
    okButtonProps: {
      disabled: true,
    },
    cancelText: 'No',
    onOk() {
      console.log('OK');
    },
    onCancel() {
      console.log('Cancel');
    },
  });
}

export default () => (
  <Space wrap>
    <Button onClick={showConfirm}>Confirm</Button>
    <Button onClick={showPromiseConfirm}>With promise</Button>
    <Button onClick={showDeleteConfirm} type="dashed">
      Delete
    </Button>
    <Button onClick={showPropsConfirm} type="dashed">
      With extra props
    </Button>
  </Space>
);
```