zh-CN

此处列出 Ant Design 中需要国际化支持的组件，你可以在演示里切换语言。

en-US

Components which need localization support are listed here, you can toggle the language in the demo.

```
import {
  ConfigProvider,
  Pagination,
  DatePicker,
  TimePicker,
  Calendar,
  Popconfirm,
  Table,
  Modal,
  Button,
  Select,
  Transfer,
  Radio,
} from 'antd';
import enUS from 'antd/lib/locale/en_US';
import zhCN from 'antd/lib/locale/zh_CN';
import moment from 'moment';
import 'moment/locale/zh-cn';

moment.locale('en');

const { Option } = Select;
const { RangePicker } = DatePicker;

const columns = [
  {
    title: 'Name',
    dataIndex: 'name',
    filters: [
      {
        text: 'filter1',
        value: 'filter1',
      },
    ],
  },
  {
    title: 'Age',
    dataIndex: 'age',
  },
];

class Page extends React.Component {
  state = {
```

```jsx
      visible: false,
  };

  showModal = () => {
    this.setState({ visible: true });
  };

  hideModal = () => {
    this.setState({ visible: false });
  };

  render() {
    const info = () => {
      Modal.info({
        title: 'some info',
        content: 'some info',
      });
    };
    const confirm = () => {
      Modal.confirm({
        title: 'some info',
        content: 'some info',
      });
    };
    return (
      <div className="locale-components">
        <div className="example">
          <Pagination defaultCurrent={1} total={50} showSizeChanger />
        </div>
        <div className="example">
          <Select showSearch style={{ width: 200 }}>
            <Option value="jack">jack</Option>
            <Option value="lucy">lucy</Option>
          </Select>
          <DatePicker />
          <TimePicker />
          <RangePicker style={{ width: 200 }} />
        </div>
        <div className="example">
          <Button type="primary" onClick={this.showModal}>
            Show Modal
          </Button>
          <Button onClick={info}>Show info</Button>
          <Button onClick={confirm}>Show confirm</Button>
          <Popconfirm title="Question?">
            <a href="#">Click to confirm</a>
          </Popconfirm>
        </div>
        <div className="example">
          <Transfer dataSource={[]} showSearch targetKeys={[]} render={item =>
item.title} />
        </div>
```

```jsx
        <div className="site-config-provider-calendar-wrapper">
          <Calendar fullscreen={false} value={moment()} />
        </div>
        <div className="example">
          <Table dataSource={[]} columns={columns} />
        </div>
        <Modal title="Locale Modal" visible={this.state.visible} onCancel=
{this.hideModal}>
          <p>Locale Modal</p>
        </Modal>
      </div>
    );
  }
}

class App extends React.Component {
  constructor() {
    super();
    this.state = {
      locale: enUS,
    };
  }

  changeLocale = e => {
    const localeValue = e.target.value;
    this.setState({ locale: localeValue });
    if (!localeValue) {
      moment.locale('en');
    } else {
      moment.locale('zh-cn');
    }
  };

  render() {
    const { locale } = this.state;
    return (
      <div>
        <div className="change-locale">
          <span style={{ marginRight: 16 }}>Change locale of components: </span>
          <Radio.Group value={locale} onChange={this.changeLocale}>
            <Radio.Button key="en" value={enUS}>
              English
            </Radio.Button>
            <Radio.Button key="cn" value={zhCN}>
              中文
            </Radio.Button>
          </Radio.Group>
        </div>
        <ConfigProvider locale={locale}>
          <Page
            key={locale ? locale.locale : 'en' /* Have to refresh for production
environment */}
```

```
          />
        </ConfigProvider>
      </div>
    );
  }
}

export default () => <App />;
```

```css
.site-config-provider-calendar-wrapper {
  width: 319px;
  border: 1px solid #d9d9d9;
  border-radius: 2px;
}

.locale-components {
  padding-top: 16px;
  border-top: 1px solid #d9d9d9;
}

.code-box-demo .example {
  margin: 16px 0;
}

.code-box-demo .example > * {
  margin-right: 8px;
}

.change-locale {
  margin-bottom: 16px;
}
```