zh-CN

我们推荐使用 `Form.useForm` 创建表单数据域进行控制。如果是在 class component 下，你也可以通过 `ref` 获取数据域。

en-US

We recommend use `Form.useForm` to create data control. If you are using class component, you can get it by `ref`.

```tsx
import { Form, Input, Button, Select } from 'antd';
import { FormInstance } from 'antd/es/form';

const { Option } = Select;

const layout = {
  labelCol: { span: 8 },
  wrapperCol: { span: 16 },
};
const tailLayout = {
  wrapperCol: { offset: 8, span: 16 },
};

class Demo extends React.Component {
  formRef = React.createRef<FormInstance>();

  onGenderChange = (value: string) => {
    switch (value) {
      case 'male':
        this.formRef.current!.setFieldsValue({ note: 'Hi, man!' });
        return;
      case 'female':
        this.formRef.current!.setFieldsValue({ note: 'Hi, lady!' });
        return;
      case 'other':
        this.formRef.current!.setFieldsValue({ note: 'Hi there!' });
    }
  };

  onFinish = (values: any) => {
    console.log(values);
  };

  onReset = () => {
    this.formRef.current!.resetFields();
  };

  onFill = () => {
    this.formRef.current!.setFieldsValue({
      note: 'Hello world!',
      gender: 'male',
```

```jsx
      });
    };

  render() {
    return (
      <Form {...layout} ref={this.formRef} name="control-ref" onFinish=
{this.onFinish}>
        <Form.Item name="note" label="Note" rules={[{ required: true }]}>
          <Input />
        </Form.Item>
        <Form.Item name="gender" label="Gender" rules={[{ required: true }]}>
          <Select
            placeholder="Select a option and change input text above"
            onChange={this.onGenderChange}
            allowClear
          >
            <Option value="male">male</Option>
            <Option value="female">female</Option>
            <Option value="other">other</Option>
          </Select>
        </Form.Item>
        <Form.Item
          noStyle
          shouldUpdate={(prevValues, currentValues) => prevValues.gender !==
currentValues.gender}
        >
          {({ getFieldValue }) =>
            getFieldValue('gender') === 'other' ? (
              <Form.Item
                name="customizeGender"
                label="Customize Gender"
                rules={[{ required: true }]}
              >
                <Input />
              </Form.Item>
            ) : null
          }
        </Form.Item>
        <Form.Item {...tailLayout}>
          <Button type="primary" htmlType="submit">
            Submit
          </Button>
          <Button htmlType="button" onClick={this.onReset}>
            Reset
          </Button>
          <Button type="link" htmlType="button" onClick={this.onFill}>
            Fill form
          </Button>
        </Form.Item>
      </Form>
    );
  }
```

```
}

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

```
#components-form-demo-control-ref .ant-btn {
  margin-right: 8px;
}
```