zh-CN

`name` 属性支持嵌套数据结构。通过 `validateMessages` 或 `message` 自定义校验信息模板，模板内容可参考此处。

en-US

`name` prop support nest data structure. Customize validate message template with `validateMessages` or `message` . Ref here about message template.

```
import { Form, Input, InputNumber, Button } from 'antd';

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

/* eslint-disable no-template-curly-in-string */
const validateMessages = {
  required: '${label} is required!',
  types: {
    email: '${label} is not a valid email!',
    number: '${label} is not a valid number!',
  },
  number: {
    range: '${label} must be between ${min} and ${max}',
  },
};
/* eslint-enable no-template-curly-in-string */

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

  return (
    <Form {...layout} name="nest-messages" onFinish={onFinish} validateMessages={validateMessages}>
      <Form.Item name={['user', 'name']} label="Name" rules={[{ required: true }]}>
        <Input />
      </Form.Item>
      <Form.Item name={['user', 'email']} label="Email" rules={[{ type: 'email' }]}>
        <Input />
      </Form.Item>
      <Form.Item name={['user', 'age']} label="Age" rules={[{ type: 'number', min: 0, max: 99 }]}>
        <InputNumber />
      </Form.Item>
      <Form.Item name={['user', 'website']} label="Website">
        <Input />
      </Form.Item>
      <Form.Item name={['user', 'introduction']} label="Introduction">
        <Input.TextArea />
      </Form.Item>
      <Form.Item wrapperCol={{ ...layout.wrapperCol, offset: 8 }}>
        <Button type="primary" htmlType="submit">
          Submit
        </Button>
      </Form.Item>
    </Form>
  );
};

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