

zh-CN

我们提供了 `validateStatus` `help` `hasFeedback` 等属性，你可以不通过 Form 自己定义校验的时机和内容。

1. `validateStatus` : 校验状态，可选 'success', 'warning', 'error', 'validating'。
2. `hasFeedback` : 用于给输入框添加反馈图标。
3. `help` : 设置校验文案。

en-US

We provide properties like `validateStatus` `help` `hasFeedback` to customize your own validate status and message, without using Form.

1. `validateStatus` : validate status of form components which could be 'success', 'warning', 'error', 'validating'.
2. `hasFeedback` : display feed icon of input control
3. `help` : display validate message.

```
import { SmileOutlined } from '@ant-design/icons';
import {
  Form,
  Input,
  DatePicker,
  TimePicker,
  Select,
  Cascader,
  InputNumber,
  Mentions,
  TreeSelect,
} from 'antd';

const { Option } = Select;

const formItemLayout = {
  labelCol: {
    xs: { span: 24 },
    sm: { span: 6 },
  },
  wrapperCol: {
    xs: { span: 24 },
    sm: { span: 14 },
  },
};

export default () => (
  <Form {...formItemLayout}>
    <Form.Item
      label="Fail"
      validateStatus="error"
      help="Should be combination of numbers & alphabets"
    >
```

```

    <Input placeholder="unavailable choice" id="error" />
  </Form.Item>

  <Form.Item label="Warning" validateStatus="warning">
    <Input placeholder="Warning" id="warning" prefix={<SmileOutlined />} />
  </Form.Item>

  <Form.Item
    label="Validating"
    hasFeedback
    validateStatus="validating"
    help="The information is being validated..."
  >
    <Input placeholder="I'm the content is being validated" id="validating" />
  </Form.Item>

  <Form.Item label="Success" hasFeedback validateStatus="success">
    <Input placeholder="I'm the content" id="success" />
  </Form.Item>

  <Form.Item label="Warning" hasFeedback validateStatus="warning">
    <Input placeholder="Warning" id="warning2" />
  </Form.Item>

  <Form.Item
    label="Fail"
    hasFeedback
    validateStatus="error"
    help="Should be combination of numbers & alphabets"
  >
    <Input placeholder="unavailable choice" id="error2" />
  </Form.Item>

  <Form.Item label="Success" hasFeedback validateStatus="success">
    <DatePicker style={{ width: '100%' }} />
  </Form.Item>

  <Form.Item label="Warning" hasFeedback validateStatus="warning">
    <TimePicker style={{ width: '100%' }} />
  </Form.Item>

  <Form.Item label="Error" hasFeedback validateStatus="error">
    <DatePicker.RangePicker style={{ width: '100%' }} />
  </Form.Item>

  <Form.Item label="Error" hasFeedback validateStatus="error">
    <Select placeholder="I'm Select" allowClear>
      <Option value="1">Option 1</Option>
      <Option value="2">Option 2</Option>
      <Option value="3">Option 3</Option>
    </Select>
  </Form.Item>

```

```

<Form.Item
  label="Validating"
  hasFeedback
  validateStatus="error"
  help="Something breaks the rule."
>
  <Cascader placeholder="I'm Cascader" options={[{ value: 'xx', label: 'xx' }]}
allowClear />
</Form.Item>

<Form.Item label="Warning" hasFeedback validateStatus="warning" help="Need to be
checked">
  <TreeSelect
    placeholder="I'm TreeSelect"
    treeData={[{ value: 'xx', label: 'xx' }]}
    allowClear
  />
</Form.Item>

<Form.Item label="inline" style={{ marginBottom: 0 }}>
  <Form.Item
    validateStatus="error"
    help="Please select right date"
    style={{ display: 'inline-block', width: 'calc(50% - 12px)' }}
  >
    <DatePicker />
  </Form.Item>
  <span
    style={{ display: 'inline-block', width: '24px', lineHeight: '32px',
textAlign: 'center' }}
  >
    -
  </span>
  <Form.Item style={{ display: 'inline-block', width: 'calc(50% - 12px)' }}>
    <DatePicker />
  </Form.Item>
</Form.Item>

<Form.Item label="Success" hasFeedback validateStatus="success">
  <InputNumber style={{ width: '100%' }} />
</Form.Item>

<Form.Item label="Success" hasFeedback validateStatus="success">
  <Input allowClear placeholder="with allowClear" />
</Form.Item>

<Form.Item label="Warning" hasFeedback validateStatus="warning">
  <Input.Password placeholder="with input password" />
</Form.Item>

<Form.Item label="Error" hasFeedback validateStatus="error">

```

```
    <Input.Password allowClear placeholder="with input password and allowClear" />
  </Form.Item>

  <Form.Item label="Fail" validateStatus="error" hasFeedback>
    <Mentions />
  </Form.Item>

  <Form.Item label="Fail" validateStatus="error" hasFeedback help="Should have
something">
    <Input.TextArea allowClear showCount />
  </Form.Item>
</Form>
);
```