Antd

1. 表单

1. 表单数据存储于上层组件

通过使用 onFieldsChange 与 mapPropsToFields，可以把表单的数据存储到上层组件或者 Redux、dva 中，更多可参考 rc-form 示例。

注意：mapPropsToFields 里面返回的表单域数据必须使用 Form.createFormField 包装。

4. Form.createFormField

用于标记 mapPropsToFields 返回的表单域数据

2. 经过 Form.create 包装的组件将会自带 this.props.form 属性，this.props.form 提供的 API 如下：

注意：使用 getFieldsValue getFieldValue setFieldsValue 等时，应确保对应的 field 已经用 getFieldDecorator 注册过了。

3. validateFields 校验并获取一组输入域的值与 Error，若 fieldNames 参数为空，则校验全部组件

5. getFieldDecorator 用于和表单进行双向绑定，详见下方描述



6. Form.Item

一个 Form.Item 建议只放一个被 getFieldDecorator 装饰过的 child，当有多个被装饰过的 child 时，help required validateStatus 无法自动生成。

7. 校验规则

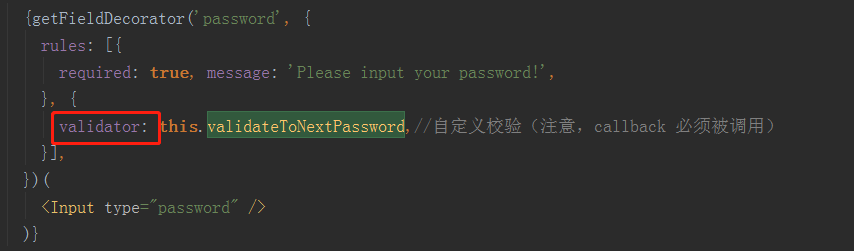
validator 自定义校验（注意，callback 必须被调用）

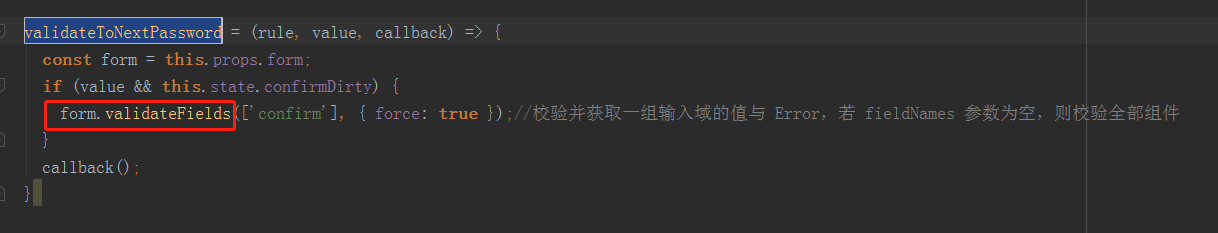
8. resetFields 重置一组输入控件的值（为 initialValue）与状态，如不传入参数，则重置所有组件

Antd 官方文档里的Form的示例代码—注册新用户那段代码

1.校验的这两个的配合用法

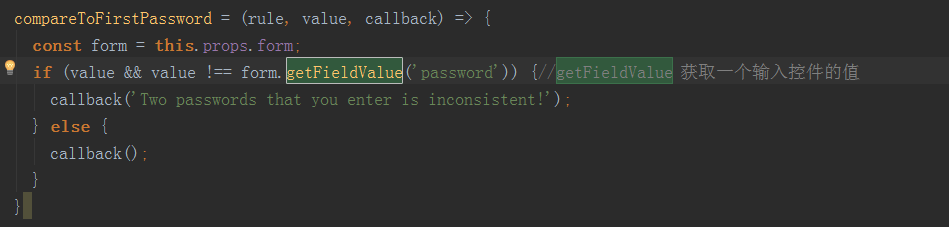
Validator和validateFields

：

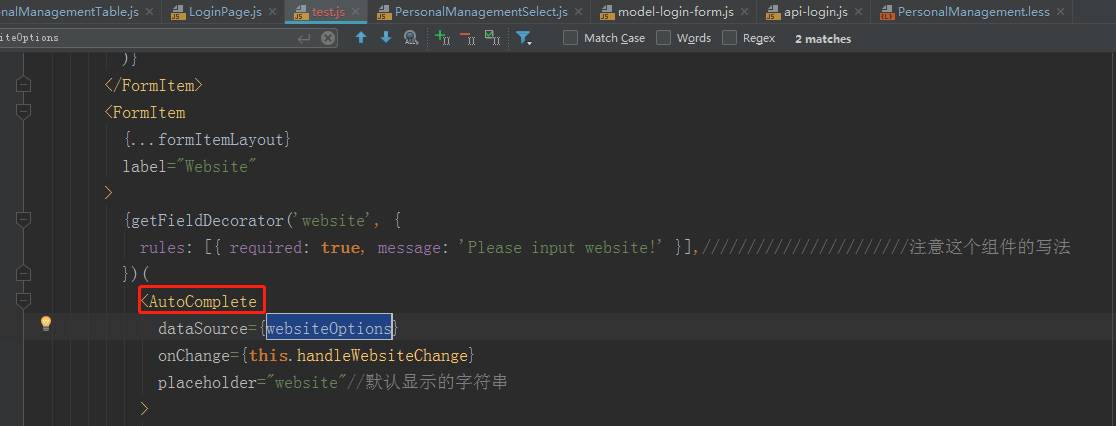


//代码里的callback()是啥？没有找到定义

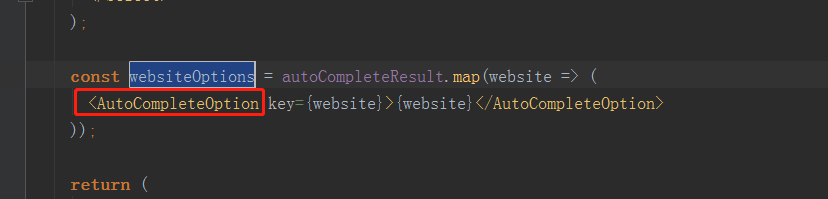
当然，在回调函数里也不是非要校验，可能是在回调里获取一个控件的值：



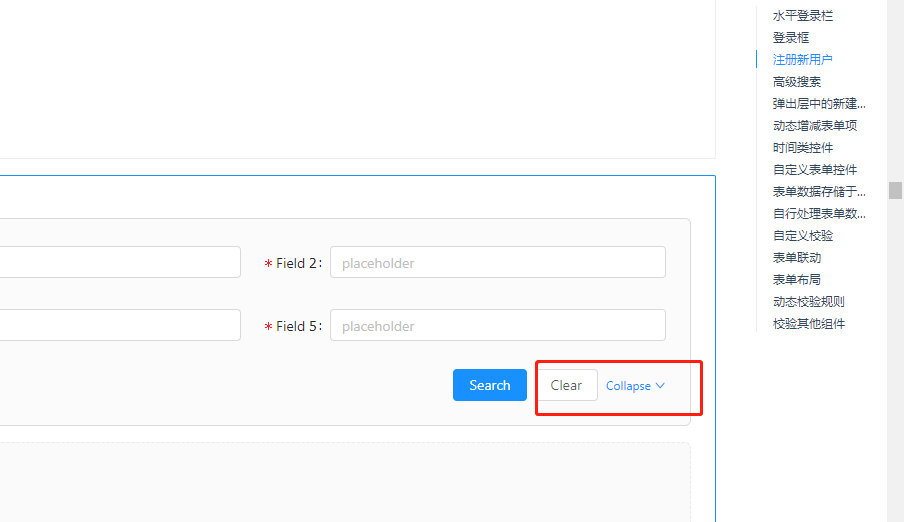
2.这个自动关联的组件叫



它的每一项是这样的，还可以存在数组里：



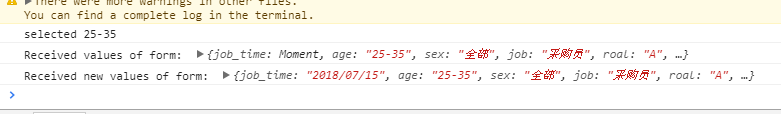
3.清除框



5. 时间类控件

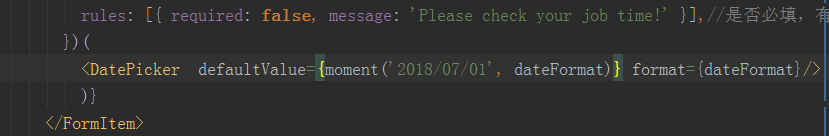
antd@2.0 之后，时间类组件的 value 改为 moment 类型，所以在提交前需要预处理。

这就是处理过时间和没有处理过时间的区别：



6. 你不能用控件的 value defaultValue 等属性来设置表单域的值，默认值可以用 getFieldDecorator 里的 initialValue。

本来这样的是会报错的：



改后不报了：

