1、基本结构



2、块渲染用于定义支持的块类型及其各自的渲染器，以及将粘贴的内容转换为已知的块类型。

当粘贴内容或使用convertFromHTML草稿时，将粘贴的内容转换为相应的块渲染类型，方法是将“草稿”块渲染图与匹配的标签进行匹配。

<h1/> header-one

<h2/> eader-two

<h3/> header-three

<h4/> header-four

<h5/> header-five

<h6/> header-six

<blockquote/> blockquote

<pre/> code-block

<figure/> atomic

<li/> unordered-list-item,ordered-list-item\*\*

<div/> unstyled\*

（1）通过将 Immutable.Map传递给编辑器blockRenderMap，可以覆盖Draft默认块渲染。

const **blockRenderMap** = **Immutable.Map**({

'header-two': {

element: 'h2'

},

'unstyled': {

element: 'h2'

}});

class RichEditor extends React.Component {

render() {

return (

<Editor

...

blockRenderMap={**blockRenderMap**}

/>

);

}}

1. 有些情况下，而不是覆盖默认值，我们只是想添加新的块类型; 这可以通过使用DefaultDraftBlockRenderMap引用来创建一个新的blockRenderMap来完成。

const blockRenderMap = Immutable.Map({

'section': {

element: 'section'

}});

Const **extendedBlockRenderMap** = **Draft.DefaultDraftBlockRenderMap.merge**(blockRenderMap);

class RichEditor extends React.Component {

render() {

return (

<Editor

...

blockRenderMap={**extendedBlockRenderMap**}

/>

);

}}

1. 当Draft解析粘贴HTML时，它将HTML元素映射回draft块类型。

如果要指定映射到特定块类型的其他HTML元素，则可以将数组**aliasedElements**添加到块配置。

'unstyled':

{

element: 'div',

aliasedElements: ['p'],

}