链滴

React Visual - 星级比率组件

作者：Vanessa
原文链接：https://ld246.com/article/1570941699454
来源网站：链滴

2019-10-13

描述

渲染一个星级比率组件。

- 定义一个组件，每颗星星都根据父组件的状态单独调用 Star 进行渲染显示
- 在 StarRating 组件中使用 React.useState() hook 定义 rating 和 selection 状态变量，并分别始化他们的值为 props.rating（如果无效或不支持时为 0）和 0
- 创建一个 hoverOver 方法，通过提供的 event 来更新 selected 和 rating
- 使用 Array.from 创建一个长度为 5 的数组，其中每一项都是一个 <Star> 组件，将这些组件包在一个 元素中。当触发 onMouseLeave 事件时设置 selection 为 0，当触发 onClick 事件时置 rating，当触发 onMouseOver 事件时设置 selection 为当前 event.target 中的 star-id 属性值
- 最后，将正确的值专递给每一个 <Star> 组件（starId and marked）

实现

```
function Star({ marked, starId }) {
  return (
    <span star-id={starId} style={{ color: '#ff9933' }} role="button">
      {marked ? '\u2605' : '\u2606'}
    </span>
  );
}

function StarRating(props) {
  const [rating, setRating] = React.useState(typeof props.rating == 'number' ? props.rating : 0)

  const [selection, setSelection] = React.useState(0);
  const hoverOver = event => {
    let val = 0;
    if (event && event.target && event.target.getAttribute('star-id'))
      val = event.target.getAttribute('star-id');
    setSelection(val);
  };
  return (
    <div
      onMouseOut={() => hoverOver(null)}
      onClick={event => setRating(event.target.getAttribute('star-id') || rating)}
      onMouseOver={hoverOver}
    >
      {Array.from({ length: 5 }, (v, i) => (
        <Star
          starId={i + 1}
          key={`star_${i + 1} `}
          marked={selection ? selection >= i + 1 : rating >= i + 1}
        />
      ))}
    </div>
  );
```

```
}
```

使用

```
ReactDOM.render(<StarRating />, document.getElementById('root'));
ReactDOM.render(<StarRating rating={2} />, document.getElementById('root'));
```

返回总目录

每天 30 秒系列之 React