

React Native - Basics - Lists

- Dr Nick Hayward

A brief intro to the basics of ListView in React Native app development.

Contents

- intro
- FlatList
- SectionList

Intro

As with other UI elements and views, React Native provides suggested view components for lists. The two primary examples include `FlatList` and `SectionList`.

FlatList

`FlatList` is meant to be used for long lists of data, in particular where data items may change during the lifecycle of an app.

A `FlatList` will only render elements currently shown on screen, and not all of the available elements at the same time.

A standard `FlatList` component will use the following pattern,

```
<FlatList
  data={[
    {key: 'Amelia'},
    {key: 'Beatrice'},
    {key: 'Daisy'},
  ]}
  renderItem={({item}) => <Text style={styles.listItem}>{item.key}</Text>
/>
```

So, this component expects two *props*, `data` for the list itself and `renderItem` to define the output structure for each list item.

For example,

```
renderItem={() => <Text></Text>}
```

SectionList

We may also create section breaks in a list of data. e.g.

```
<SectionList
  sections={[
    {title: 'a', data:[{key: 1, name: 'Amelia'}]},
    {title: 'b', data:[{key: 1, name: 'Beatrice'}]},
    {title: 'c', data: [{key: 1, name: 'Daisy'}, {key: 2, name: 'Devon'}]},
    {title: 'e', data: [{key: 1, name: 'Elizabeth'}, {key: 2, name:
'Emma'}, {key: 3, name: 'Evelyn'}]},
    {title: 'g', data:[{key: 1, name: 'Georgiana'}]},
    {title: 'j', data:[{key: 1, name: 'Jane'}]},
    {title: 'r', data:[{key: 1, name: 'Rose'}]},
    {title: 'v', data: [{key: 1, name: 'Victoria'}, {key: 2, name:
'Violet'}]},
    {title: 'y', data:[{key: 1, name: 'Yvaine'}]},
  ]}
  //keyExtractor={item => item}
  renderItem={({item}) => <Text style={styles.listItem}>{item.key} - {item.name}
</Text>}
  renderSectionHeader={({section}) => <Text style={styles.heading4}>{section.title}
</Text>}
/>
```