

Working With Core Components



“Core” Components (Built Into React Native)

“Translation” to native UI widgets is provided by
React Native

```
<View />
```

```
<Text />
```

```
<Button />
```

```
<TextInput />
```

```
<Image />
```

```
<... />
```



Your UI & Custom Components

Combination of “Core” components & other
built-in components

```
const MyTitle = props => {  
  return (  
    <View>  
      <Text>{props.title}</Text>  
    </View>  
  );  
};
```

Styling React Native Apps



There Is No CSS!

Inline Styles



StyleSheet Objects

Written in JavaScript

(i.e. in the JavaScript code files, next to the component code)

Based on CSS syntax, but only a **subset** of properties & features is supported!

Layouts & Flexbox

Layouts are (typically) created with Flexbox

Very similar to browser CSS flexbox!

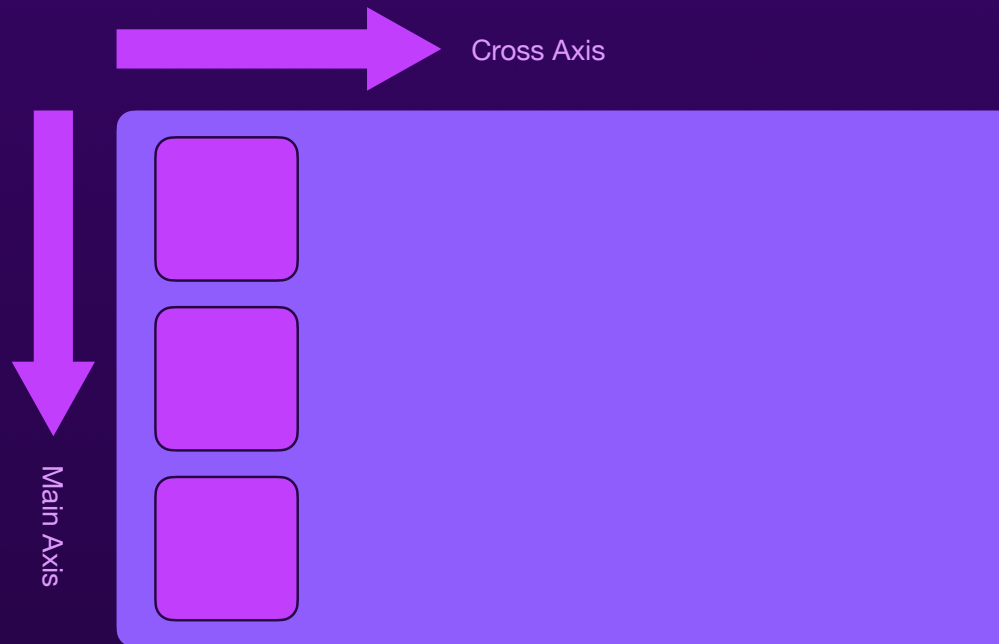


Elements are positioned inside of containers

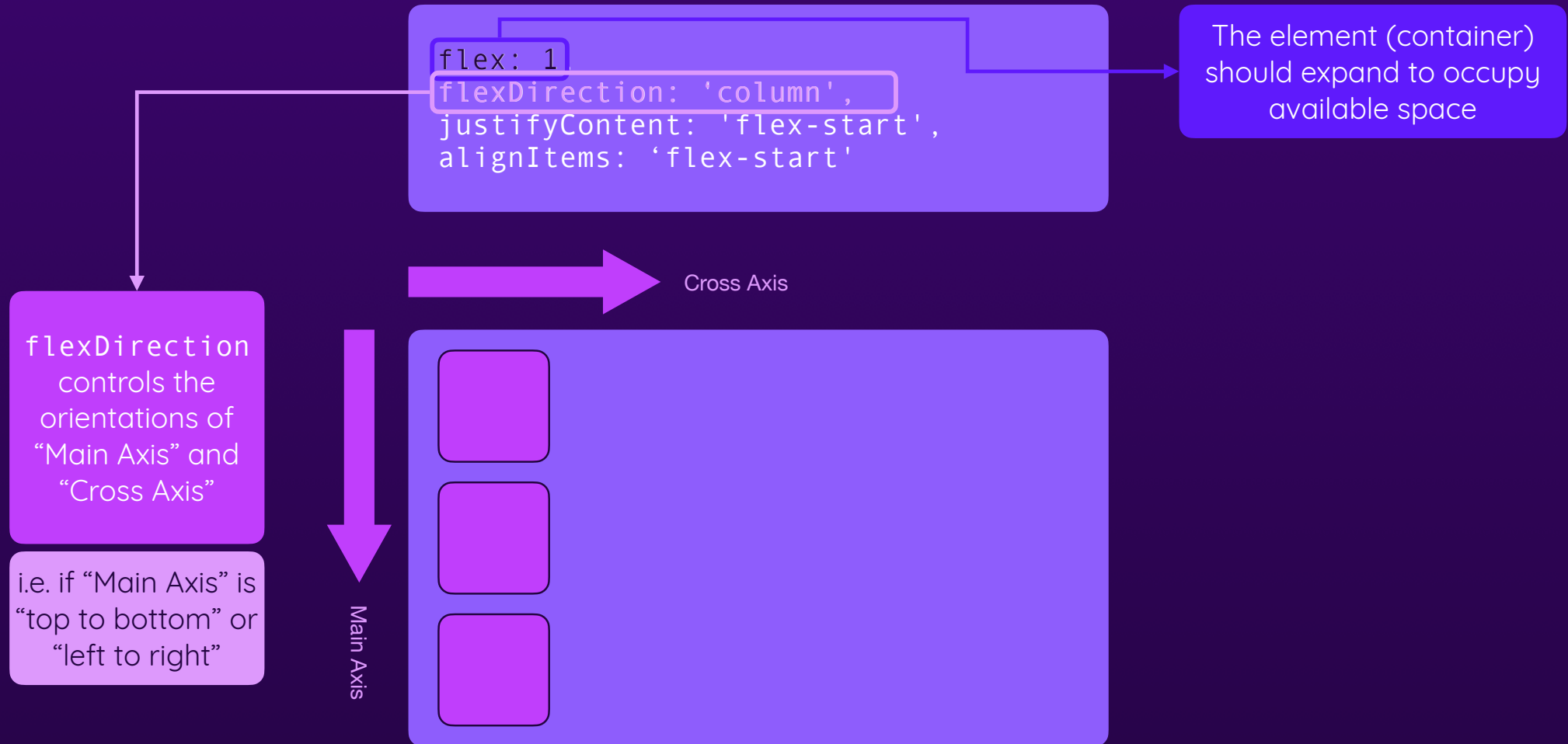
Positioning is controlled via style settings applied to the element container

Layouts & Flexbox

```
flex: 1,  
flexDirection: 'column',  
justifyContent: 'flex-start',  
alignItems: 'flex-start'
```

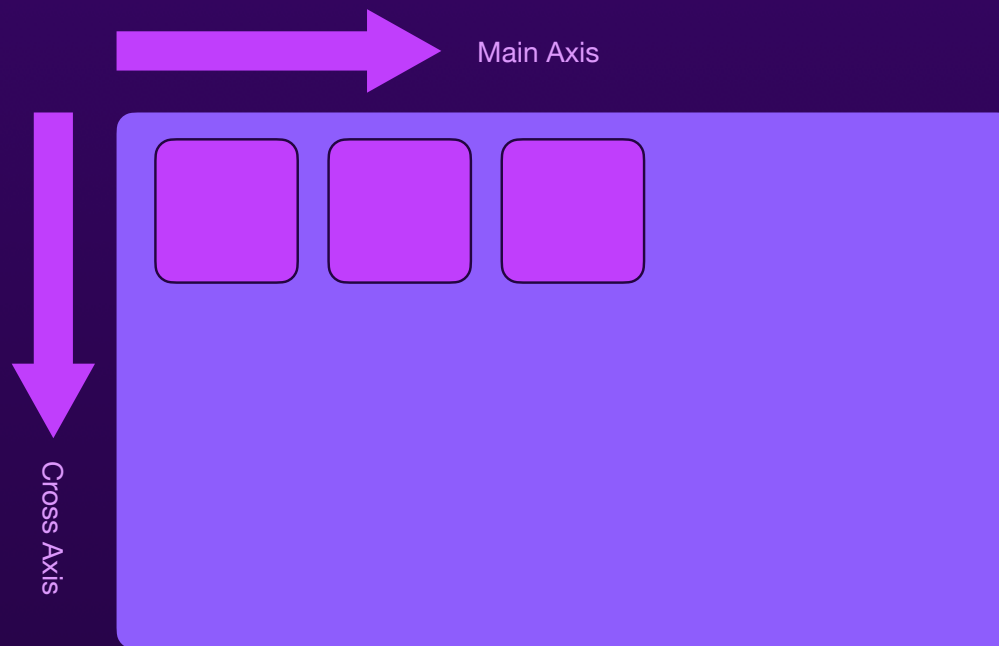


Layouts & Flexbox



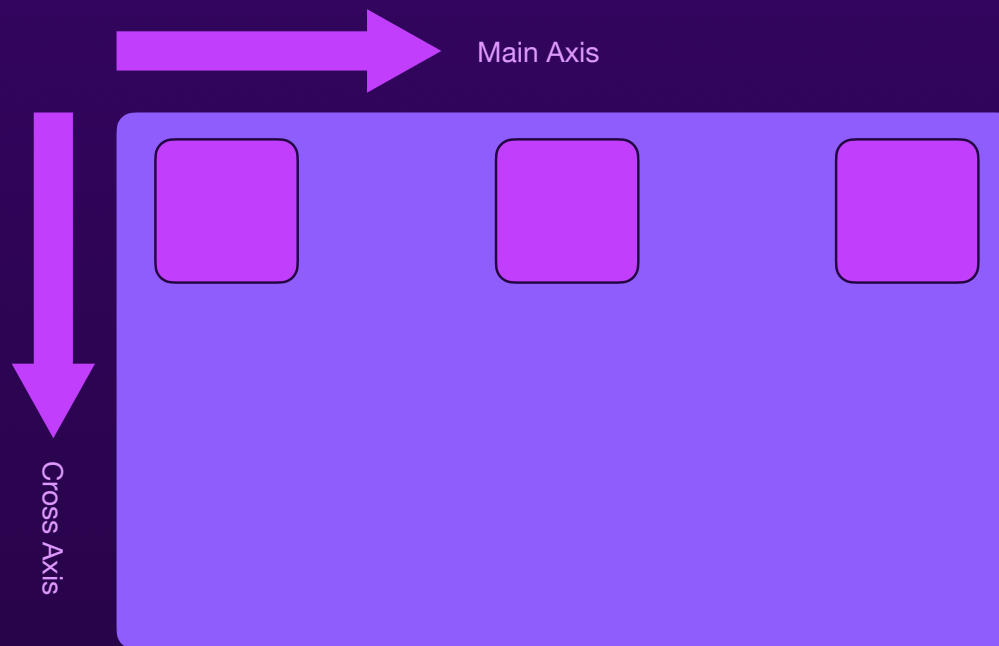
Layouts & Flexbox

```
flex: 1,  
flexDirection: 'row',  
justifyContent: 'flex-start',  
alignItems: 'flex-start'
```



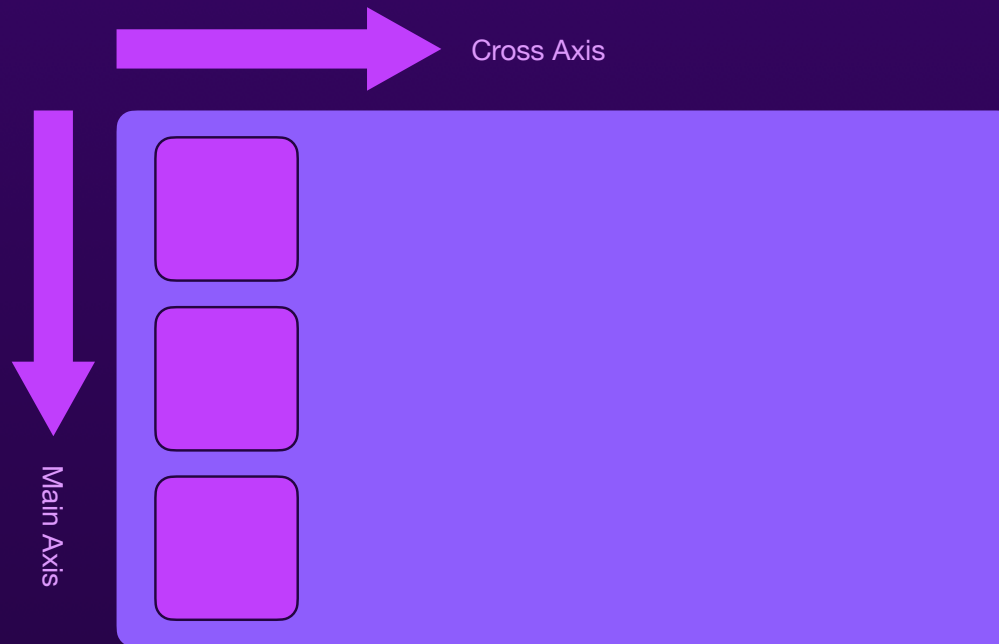
Layouts & Flexbox

```
flex: 1,  
flexDirection: 'row',  
justifyContent: 'space-between',  
alignItems: 'flex-start'
```



Layouts & Flexbox

```
flex: 1,  
flexDirection: 'column',  
justifyContent: 'space-between',  
alignItems: 'flex-start'
```



Multiple Flexbox Containers

Multiple containers split
the available space

flex values are set into
relation to determine
each container's space

flex: 2

flex: 1