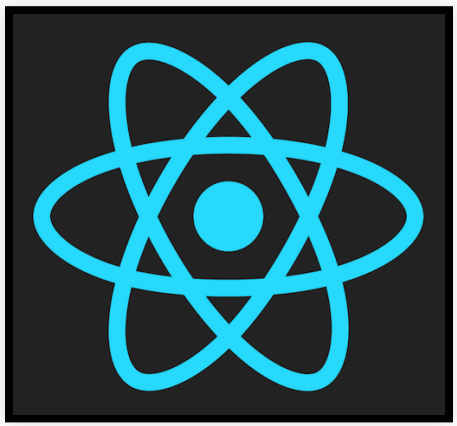
React Native Training



Rahman Usta
RUsta@luxoft.com

React Native

Mobile application development framework based on React.js. It uses JavaScriptCore for a JS engine.

Advantages of React Native

- Same code for 2 platform (IOS ve Android)
- Mobile development with JavaScript
- Fast development and debugging
- Native close performance
- Advantages of React.js
- Styling with CSS

Who Uses?

- Facebook
- Instagram
- Walmart
- Airbnb
- Tesla
- SoundCloud Pulse

Requirements

- Node.js
- Expo, emulator or real device

Create a React Native Project

```
npm install -g expo-cli

expo init MyMobileProject
cd MyMobileProject
npm start
```

Install Expo Client

Expo client is a mobile application to run React Native applications on mobile devices during development.

https://expo.io/tools#client

React Native Components

- View
- Text
- Button
- Picker
- TextInput
- Alert
- Modal
- WebView
- FlatList
- Switch
- and more..

React Native Debugging

Shake your device, and Click `Start Remote JS Debugging` in expo client

React Native APIs

- Dimensions
- Platform
- Clipboard
- AsyncStorage
- Vibration
- StyleSheet
- more..

React Native Styling

CSS is supported to style React Native components.

```
(<View style={styles.container}>
    <Text>Hello World</Text>
</View>)

let styles= StyleSheet.create({
    container: {
        flex: 1,
        backgroundColor: "red"
    }
});
```

React Native Layout

React Native uses a responsive grid model called Flexbox.

Flex Direction - Column #1

`flex` places elements on the screen with a proportional coefficient. Direction, axis can be top to down (`column`), or left to right (`row`). Default flexDirection is : `column`

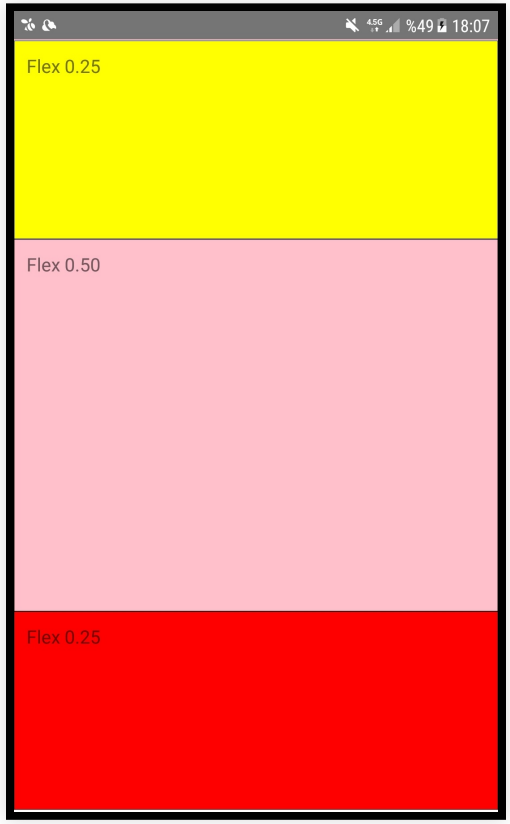


Figure 1. src/Layout000.js

Flex Direction - Row #2

To change the direction, flexDirection CSS attribute should be used.

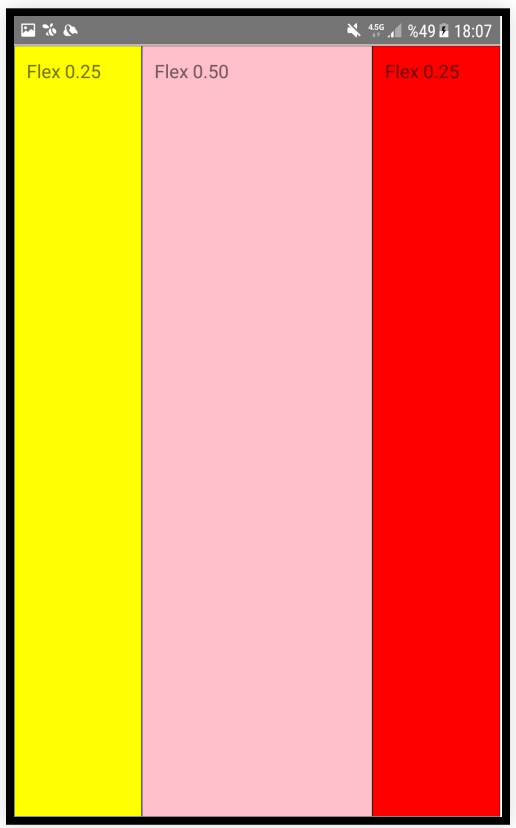


Figure 2. src/Layout000.js

Positioning of elements #1

justifyContent

Determines to place of elements on the main axis (`flex-start`, `flex-end`, `center`, `space-around`, `space-between`)

Elemanların yerleşimi #2

alignItems

Determines to place of elements on the secondary axis (`flex-start`, `flex-end`, `center`, `stretch`)

Example #1

Figure 3. flexDirection=column, justifyContent=flex-start

Example #2

Figure 4. flexDirection=column, justifyContent=flex-end

Example #3

Figure 5. flexDirection=column, justifyContent=center

Example #4

Figure 6. flexDirection=column, justifyContent=space-around

Example #5

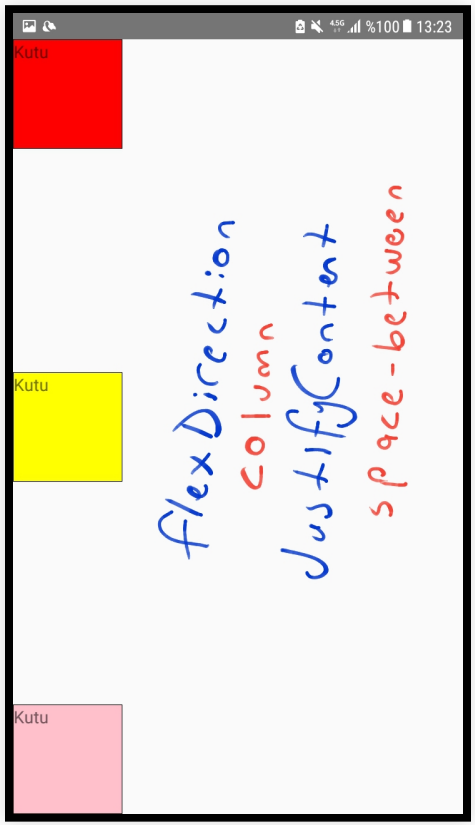


Figure 7. flexDirection=column, justifyContent=space-between

Example #6

Figure 8. flexDirection=row, justifyContent=flex-start

Example #7

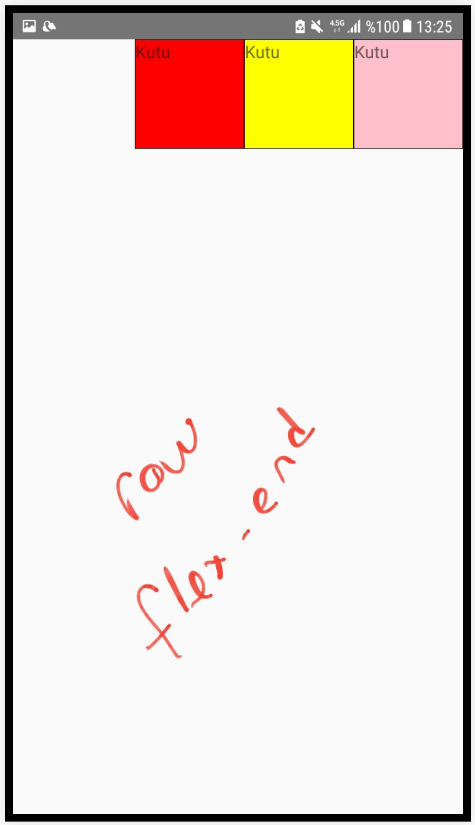


Figure 9. flexDirection=row, justifyContent=flex-end

Example #8

Figure 10. flexDirection=row, justifyContent=center

Example #9

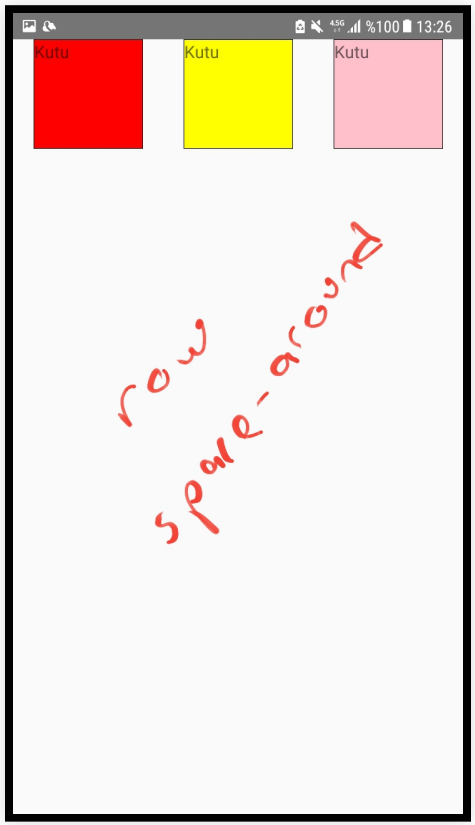


Figure 11. flexDirection=row, justifyContent=space-around

Example #10

Figure 12. flexDirection=row, justifyContent=space-between

Example #11

Figure 13. flexDirection=column, justifyContent=center, alignItems=flex-start

Example #12

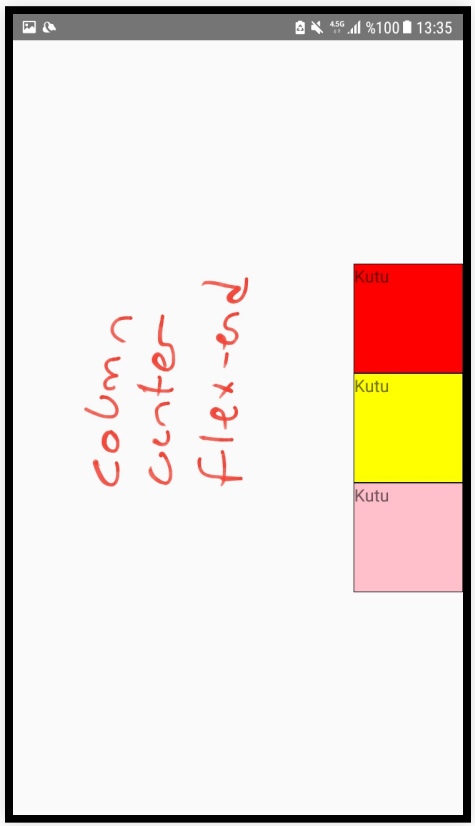


Figure 14. flexDirection=column, justifyContent=center, alignItems=flex-end

Example #13

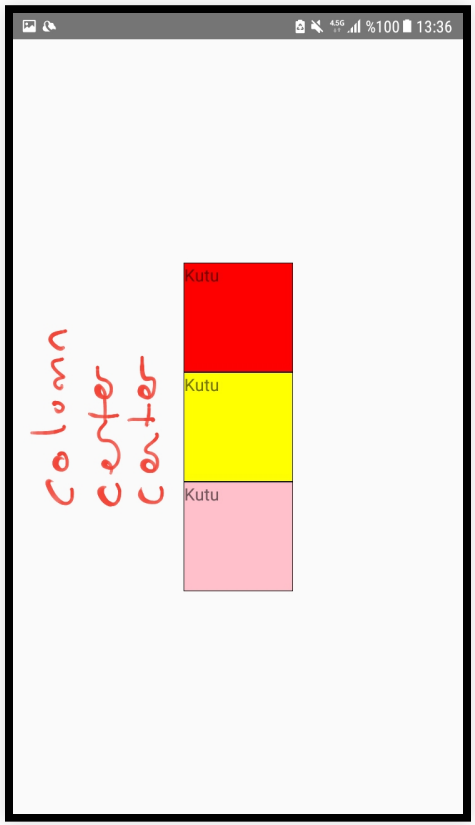


Figure 15. flexDirection=column, justifyContent=center, alignItems=center

Example #14

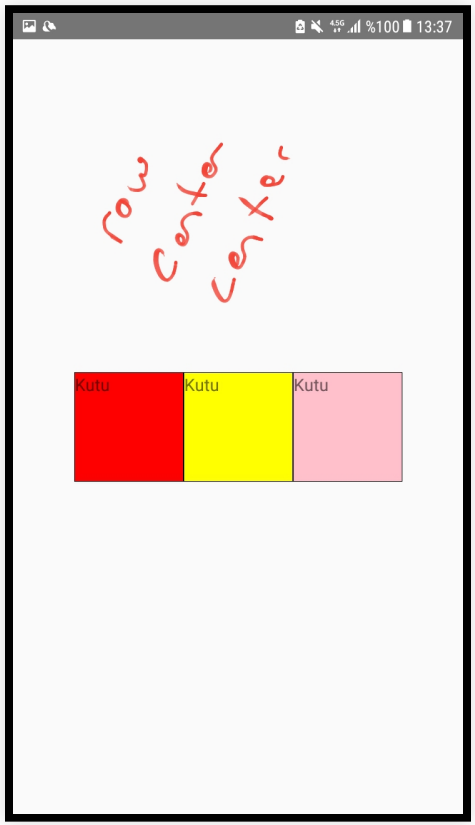


Figure 16. flexDirection=row, justifyContent=center, alignItems=center

Button

```
<Button
      title="Press me"
      color="#f194ff"
      onPress={() => Alert.alert('Pressed')}
/>
```

TextInput

```
 <TextInput
      multiline={false}
      numberOfLines={1}
      onChangeText={text => onChangeText(text)}
      value={value}
/>
```

Text

```
<Text>{someText}</Text>

<Text numberOfLines={5}>
  {someText}
</Text>
```

FlatList

```
<FlatList
    data={DATA}
    renderItem={({ item }) => <Item title={item.title} />}
    keyExtractor={item => item.id}
/>
```

Other components

https://facebook.github.io/react-native/docs/button

React Native Elements

https://react-native-elements.github.io/react-native-elements/docs/overview.html

Demo application

Demo

Building Native Packages

```
expo build:android
expo build:ios
```

Deploying App

https://docs.expo.io/versions/latest/workflow/publishing/#to-the-app-store-and-play

The End

Thank you