

# День 2

Стили и Flexbox

# Стили

**В React Native не используется специальный язык или синтаксис для определения стилей.**

**Для применения стилей используется JavaScript.**

# style

**Все встроенные компоненты в React Native для применения стилей используют свойство `style`.**

**Названия стилей и значений похожи на CSS.  
Различие в правописаний.**

**backgroundColor, а не background-color.**

**fontSize, а не font-size.**

# Встроенные компоненты

**View, Text, Image, Button, TextInput, FlatList ...**

**<Text>НИШ</Text>**

**<Text style={} >НИШ</Text>**

```
<Text style={ { color: 'blue', fontSize: 20 } } >HIШ</Text>
```





```
<View style={{ flex: 1, backgroundColor: 'pink' }} />
```

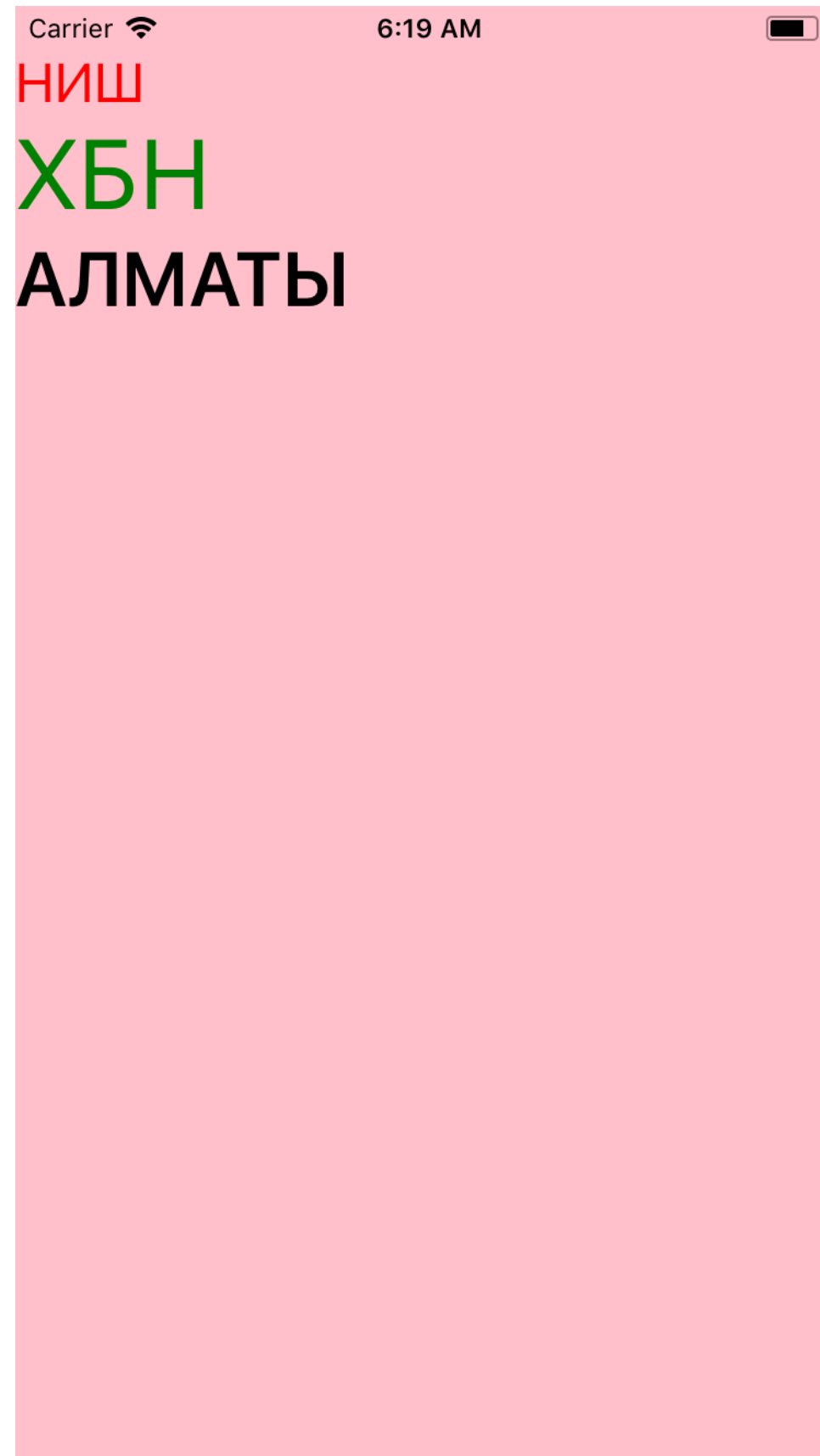
```
<View style={styles.container} />
```

```
const styles = StyleSheet.create({  
  container: {  
    flex: 1,  
    backgroundColor: 'pink'  
  }  
});
```



```
<View style={styles.container}>
  <Text style={styles.text1}>НИШ</Text>
  <Text style={styles.text2}>ХБН</Text>
  <Text style={styles.text3}>АЛМАТЫ</Text>
</View>
```

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    paddingTop: 20,
    backgroundColor: 'pink'
  },
  text1: {
    fontSize: 25,
    color: 'red'
  },
  text2: {
    fontSize: 45,
    color: 'green'
  },
  text3: {
    fontSize: 35,
    color: 'black',
    fontWeight: 'bold'
  }
});
```



```
<View style={styles.container}>
  <Text style={styles.text1}>НИШ</Text>
  <Text style={styles.text2}>ХБН</Text>
  <Text style={styles.text3}>АЛМАТЫ</Text>
</View>
```

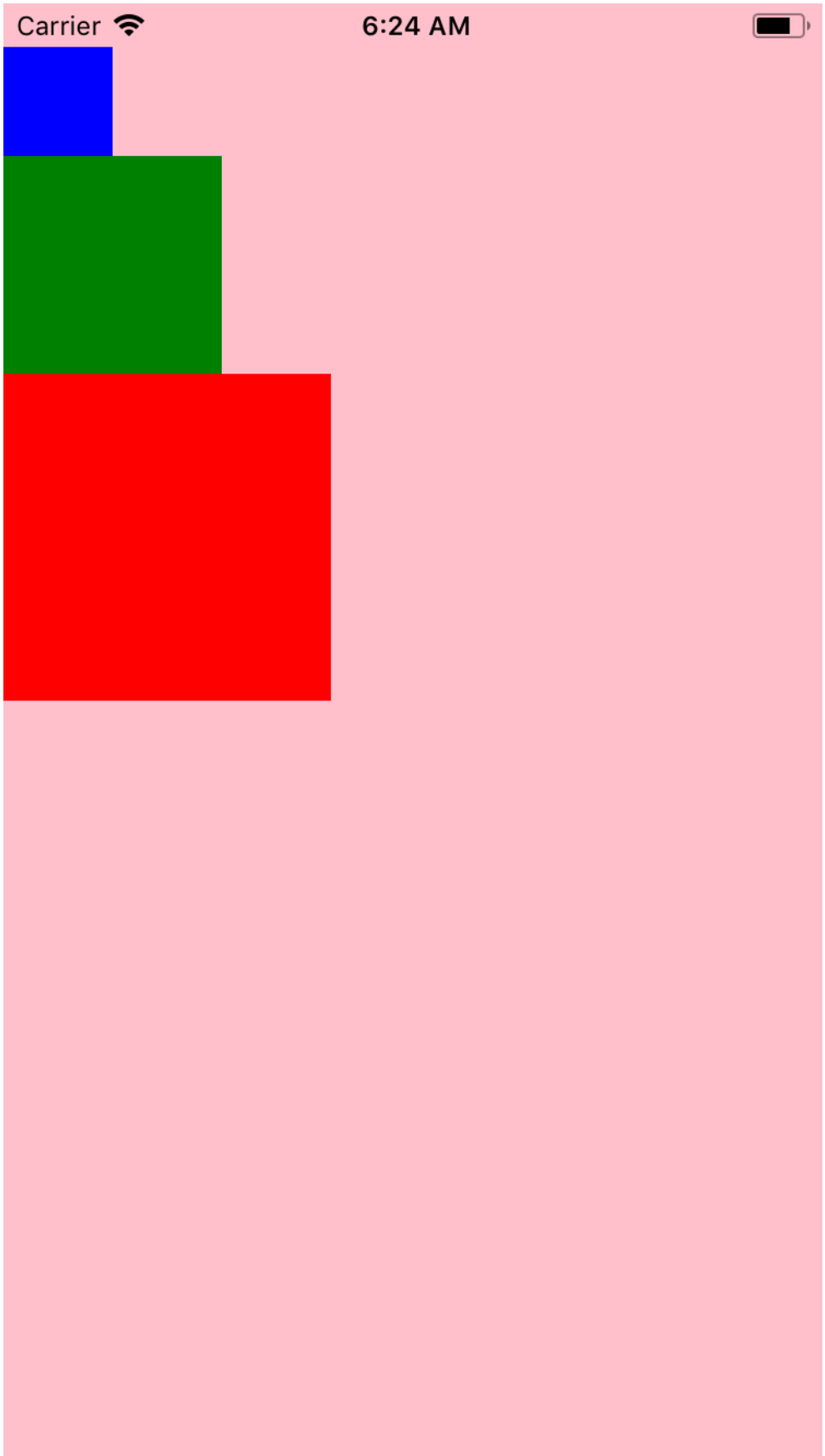
```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    paddingTop: 20,
    backgroundColor: 'pink'
  },
  text1: {
    fontSize: 25,
    color: 'red',
    textAlign: 'right'
  },
  text2: {
    fontSize: 45,
    color: 'green'
  },
  text3: {
    fontSize: 35,
    color: 'black',
    fontWeight: 'bold',
    textAlign: 'right'
  }
});
```





```
<View style={styles.container}>
  <View style={styles.box1} />
  <View style={styles.box2} />
  <View style={styles.box3} />
</View>
```

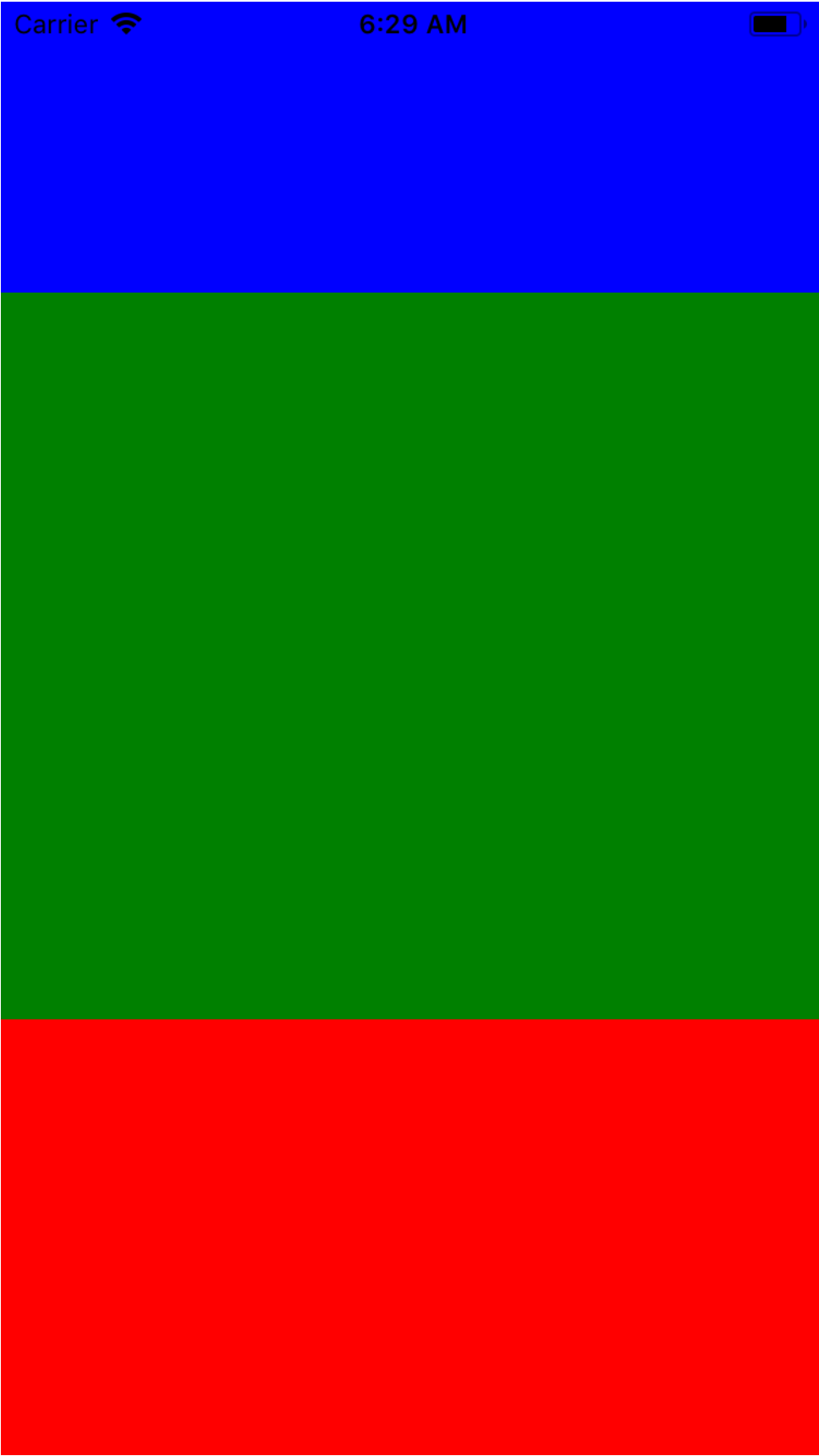
```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    paddingTop: 20,
    backgroundColor: 'pink'
  },
  box1: {
    width: 50,
    height: 50,
    backgroundColor: 'blue'
  },
  box2: {
    width: 100,
    height: 100,
    backgroundColor: 'green'
  },
  box3: {
    width: 150,
    height: 150,
    backgroundColor: 'red'
  }
});
```



**Flex**

**С помощью flex, регулируется размер контейнера, относительно пропорции экрана устройства.**

```
<View style={flex: 2} /> // 20%  
<View style={flex: 4} /> // 40%  
<View style={flex: 3} /> // 30%
```



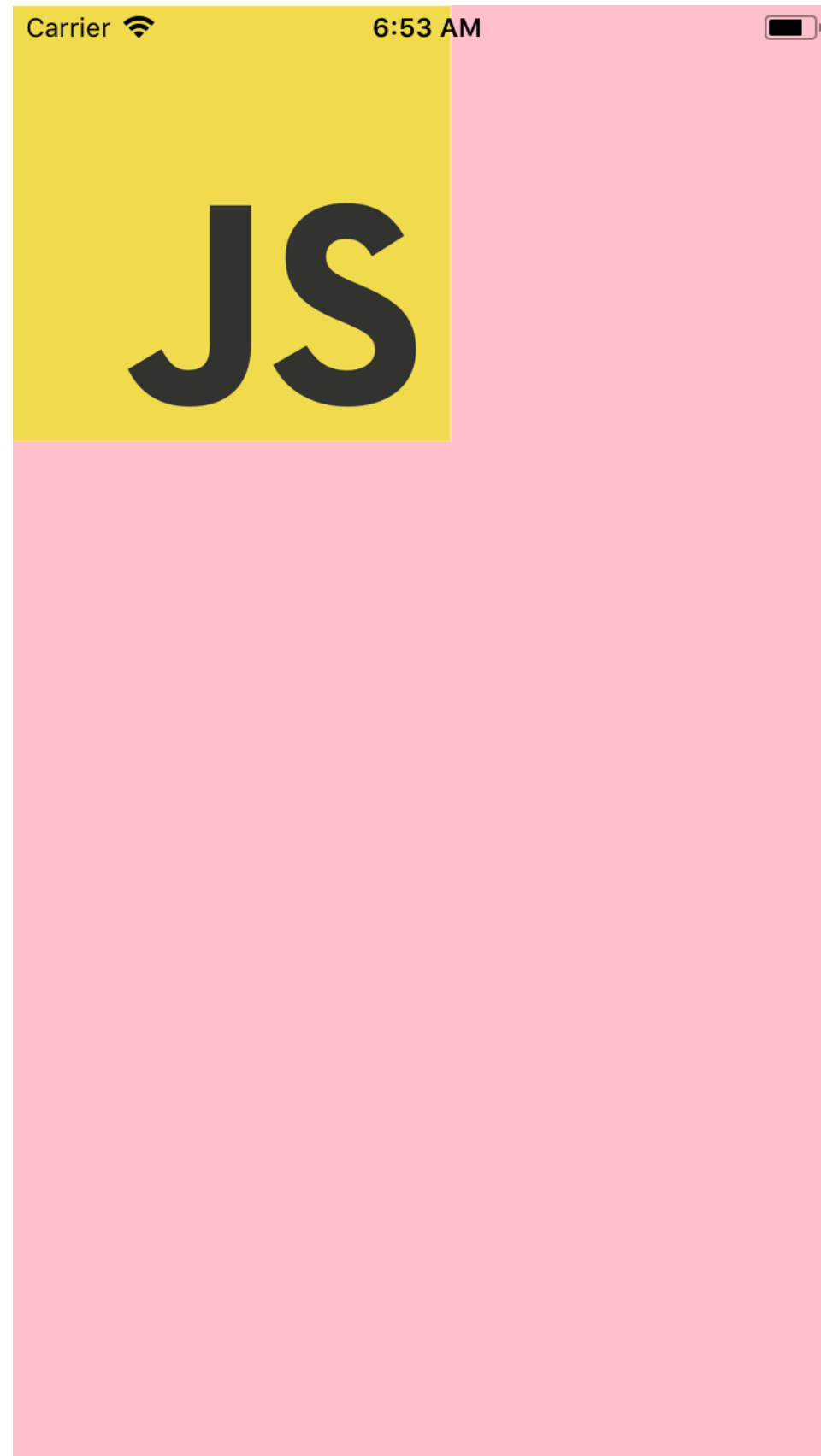
# Практика 1

Image



```
<Image source={require('./assets/jsLogo.png')} style={styles.box2} />
```

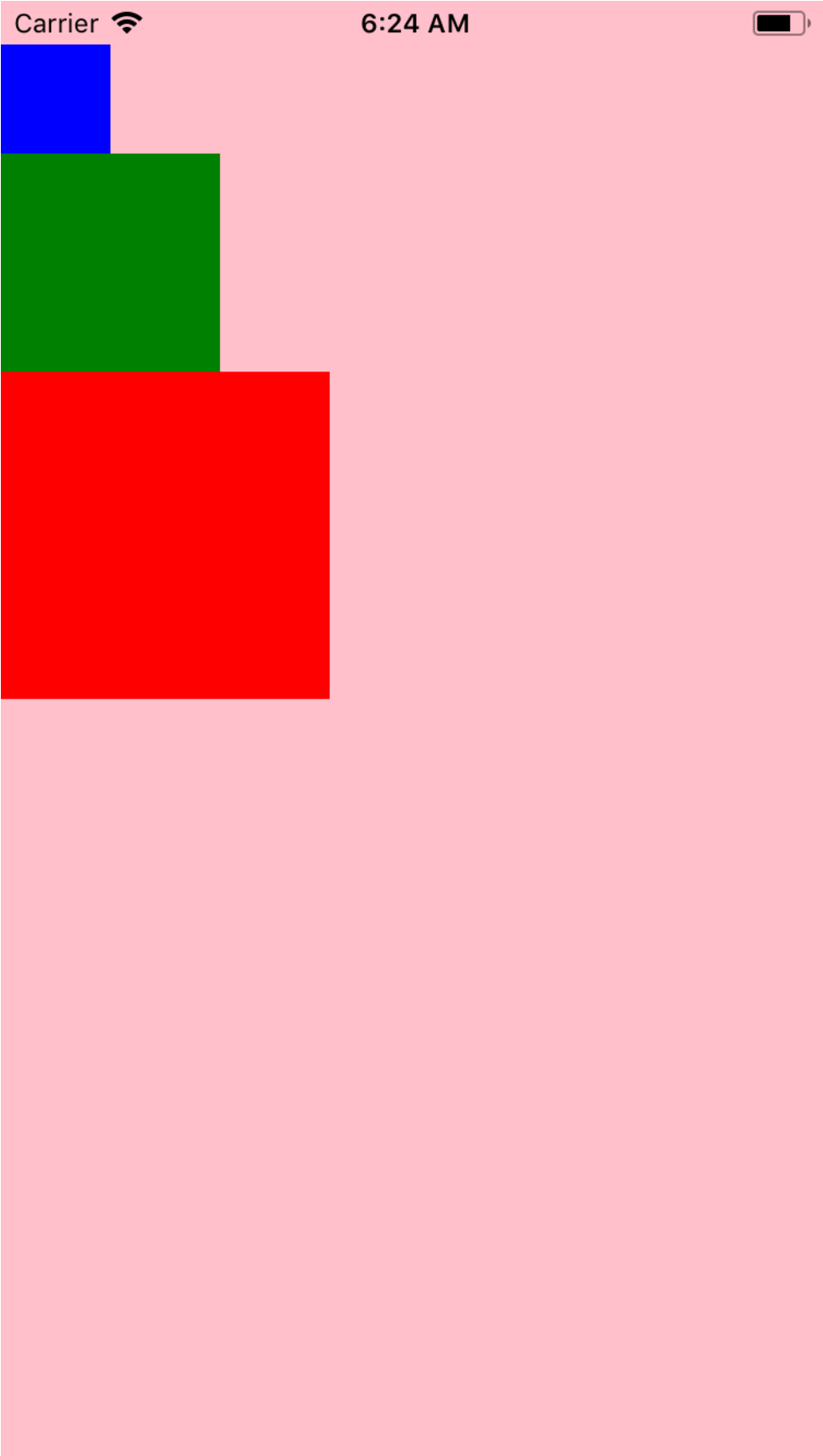
```
box2: {  
  width: 200,  
  height: 200,  
  backgroundColor: 'green'  
}
```



# Практика 2

# Flex Direction

**По умолчанию в React Native главной является вертикальная ось.**

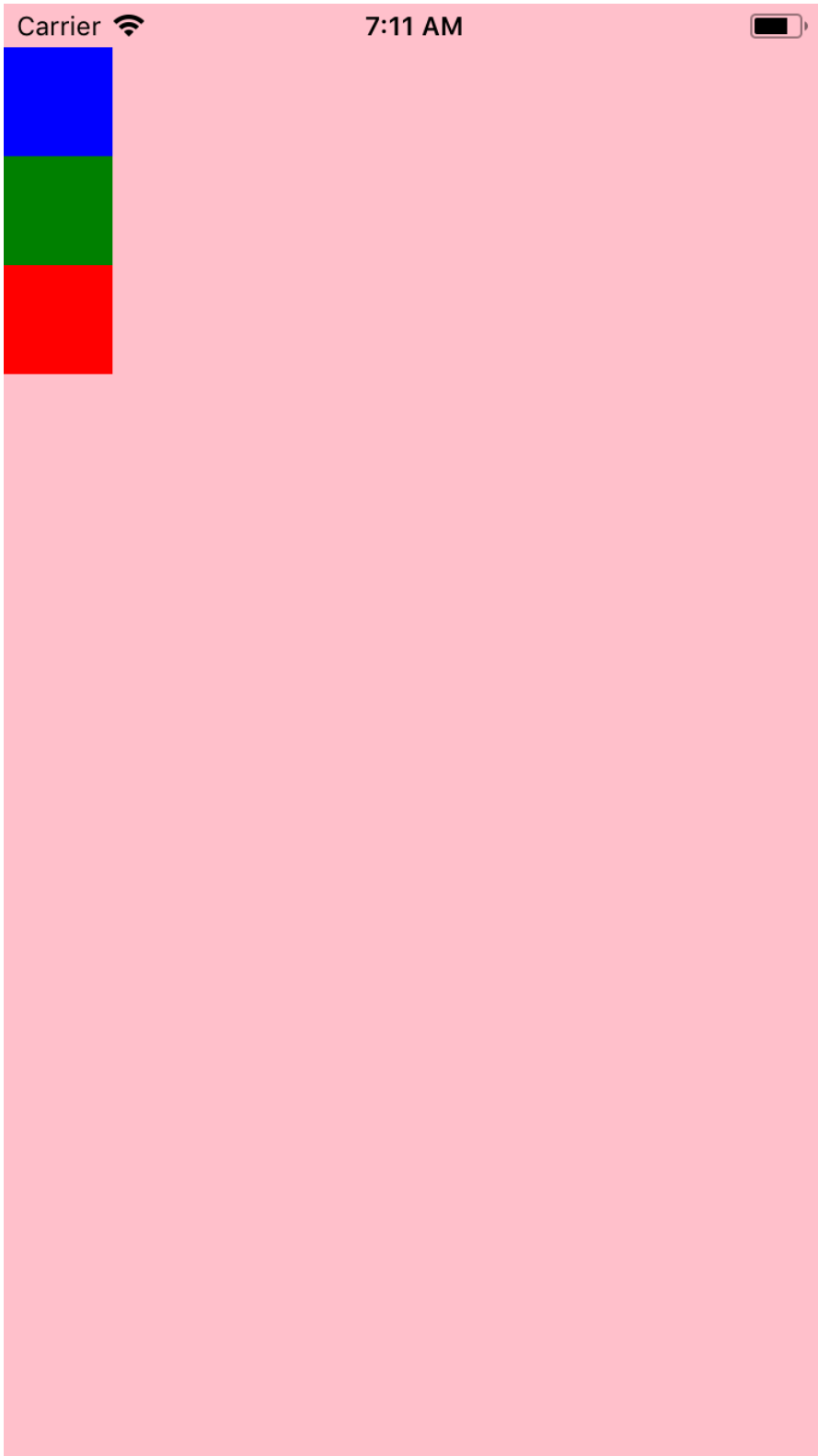


Главную ось можно задать с помощью **flexDirection** через значения **row** и **column**

```
<View style={styles.container}>  
  <View style={styles.box1} />  
  <View style={styles.box2} />  
  <View style={styles.box3} />  
</View>
```

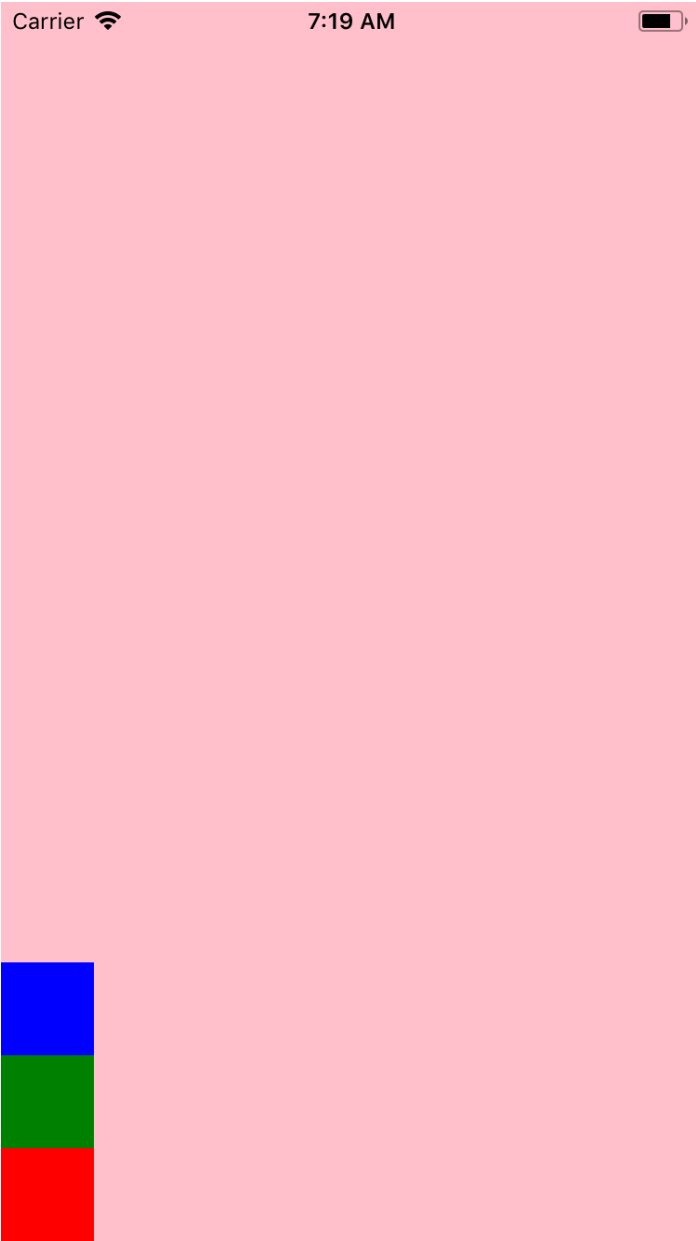
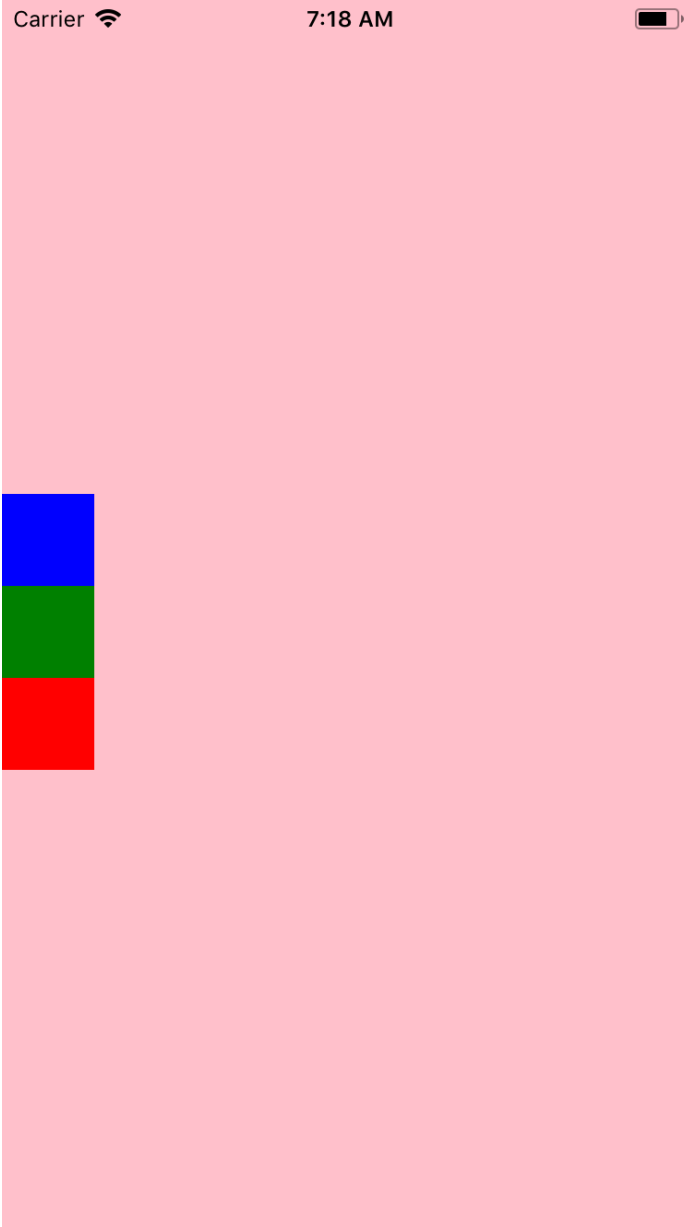
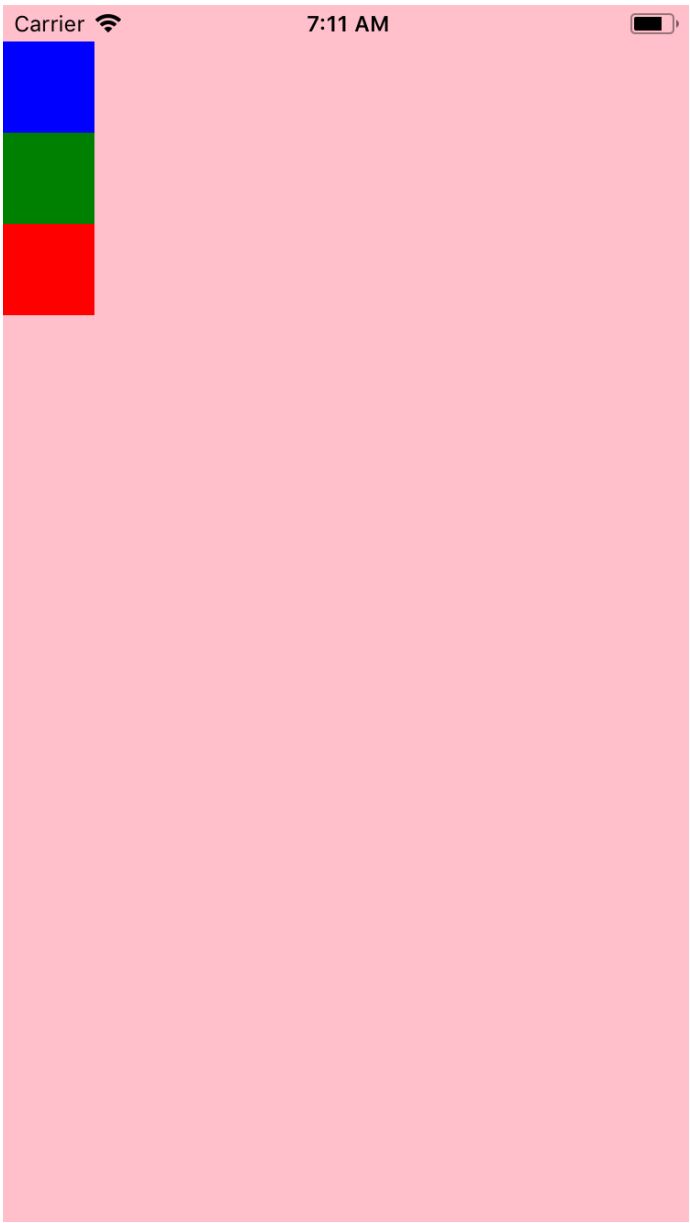
```
container: {  
  flex: 1,  
  paddingTop: 20,  
  flexDirection: 'row',  
  backgroundColor: 'pink'  
},
```





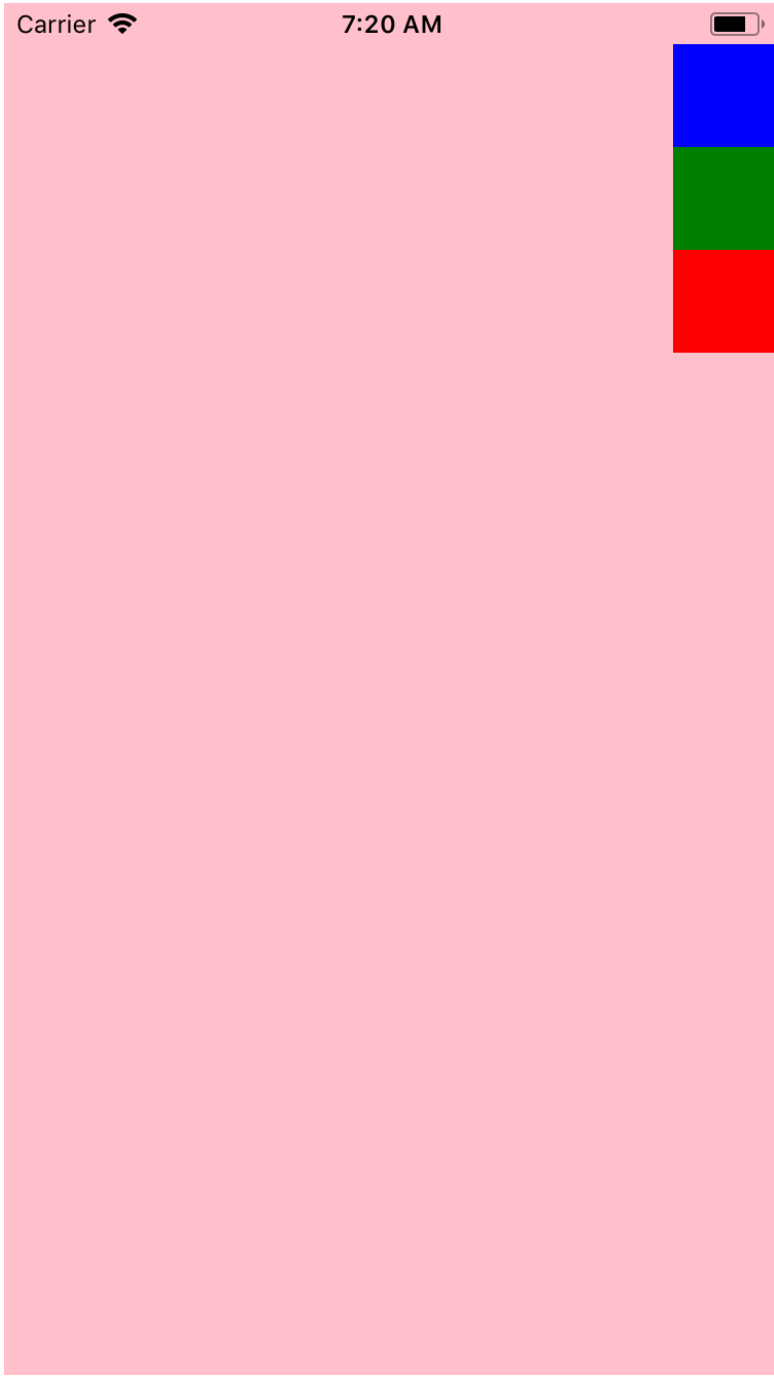
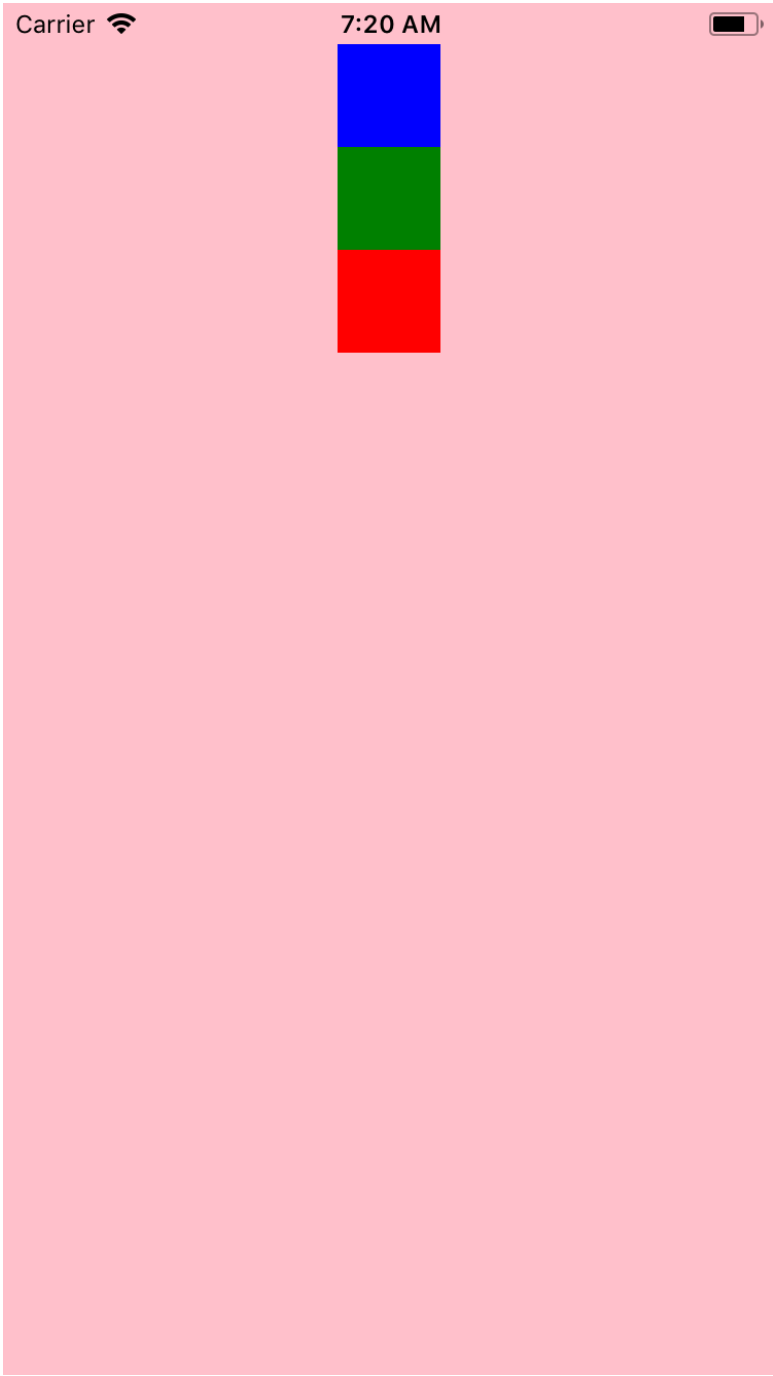
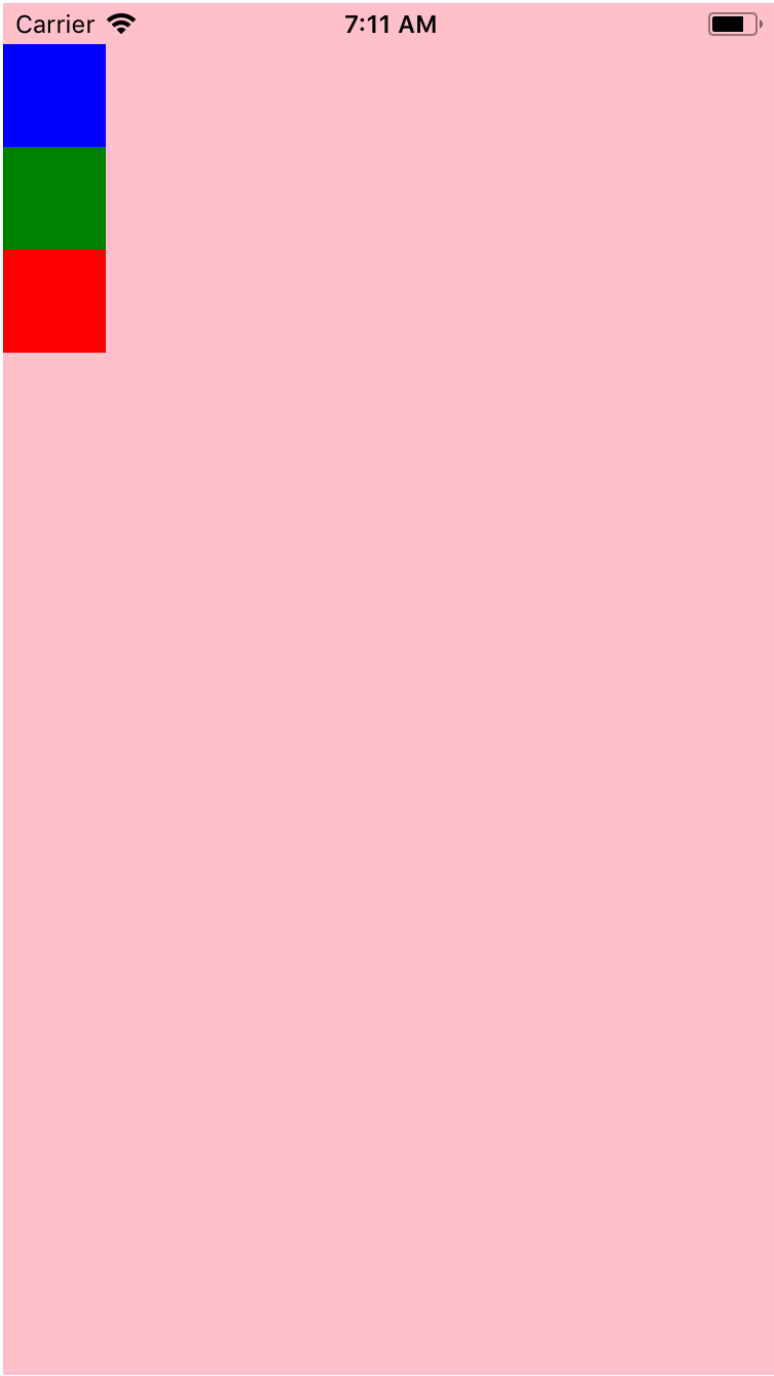
**justifyContent** - выравнивание по главной оси.

принимаемые значения: **'center', 'flex-start', 'flex-end', 'space-around', 'space-between'**



**alignItems** - выравнивание по второстепенной оси.

принимаемые значения: 'center', 'flex-start', 'flex-end', 'space-around', 'space-between'



# Практика 3