REACT NATIVE

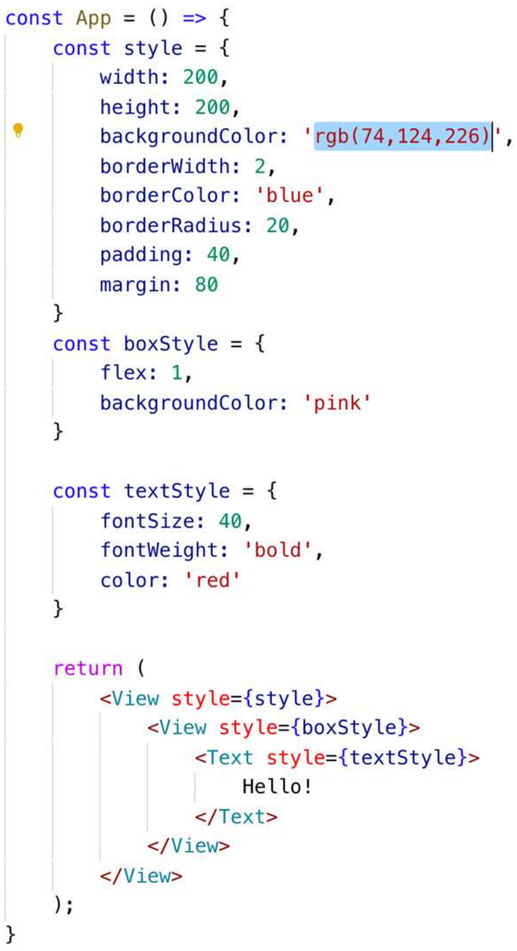
React Native Styling:

CSS, Inline styles, FlexBox

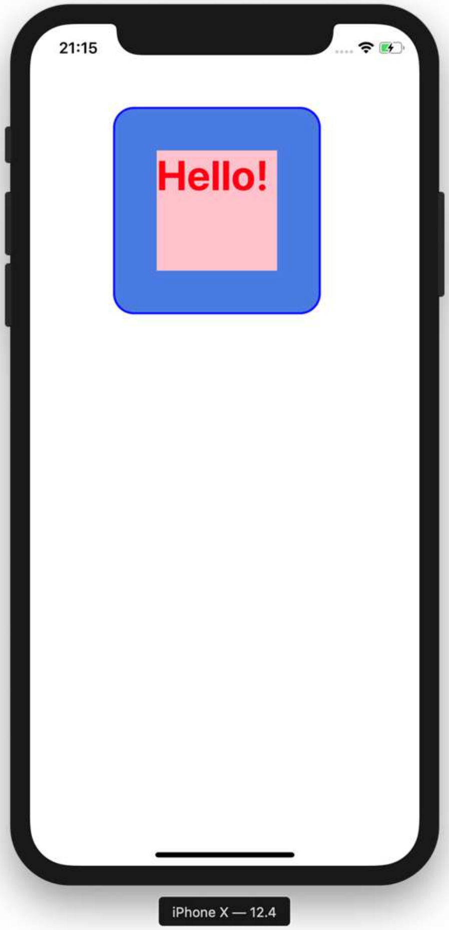
Style

HIENLTH

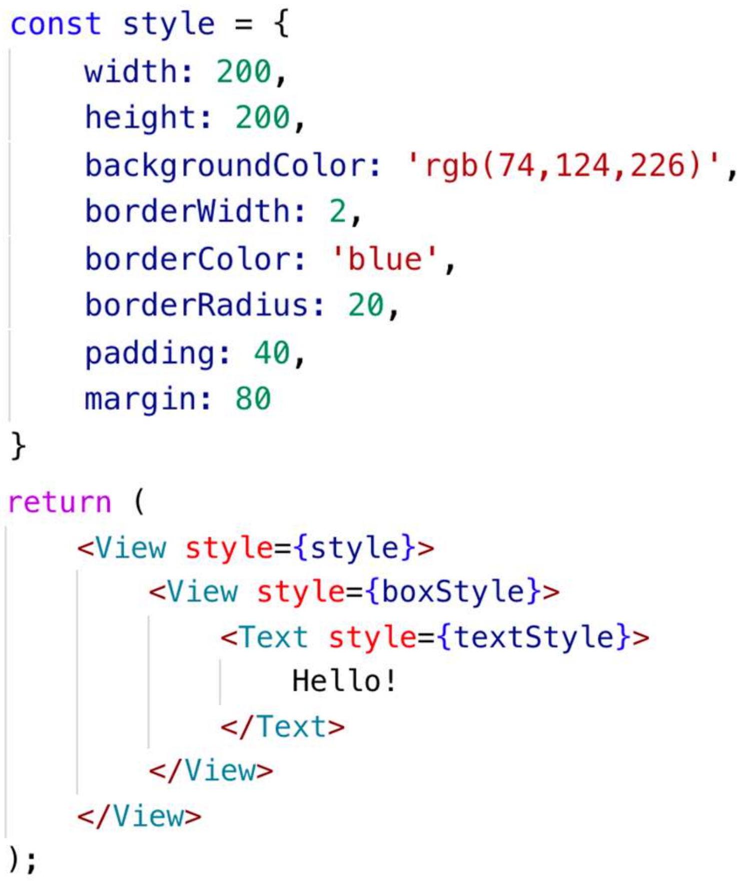
Basic CSS



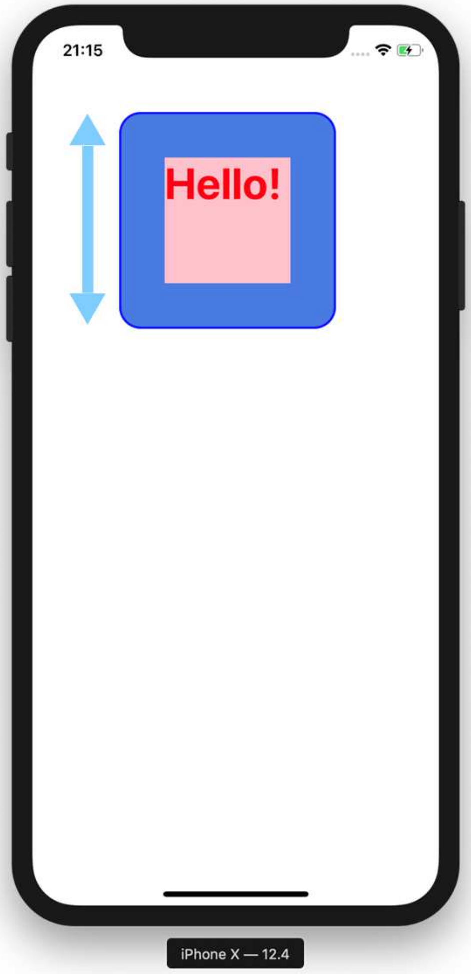
```
const App = () => {
    const style = {
        width: 200,
        height: 200,
        backgroundColor: 'rgb(74,124,226)',
        borderWidth: 2,
        borderColor: 'blue',
        borderRadius: 20,
        padding: 40,
        margin: 80
    }
    const boxStyle = {
        flex: 1,
        backgroundColor: 'pink'
    }

    const textStyle = {
        fontSize: 40,
        fontWeight: 'bold',
        color: 'red'
    }

    return (
        <View style={style}>
            <View style={boxStyle}>
                <Text style={textStyle}>
                    Hello!
                </Text>
            </View>
        </View>
    );
}
```



View: Blue Box

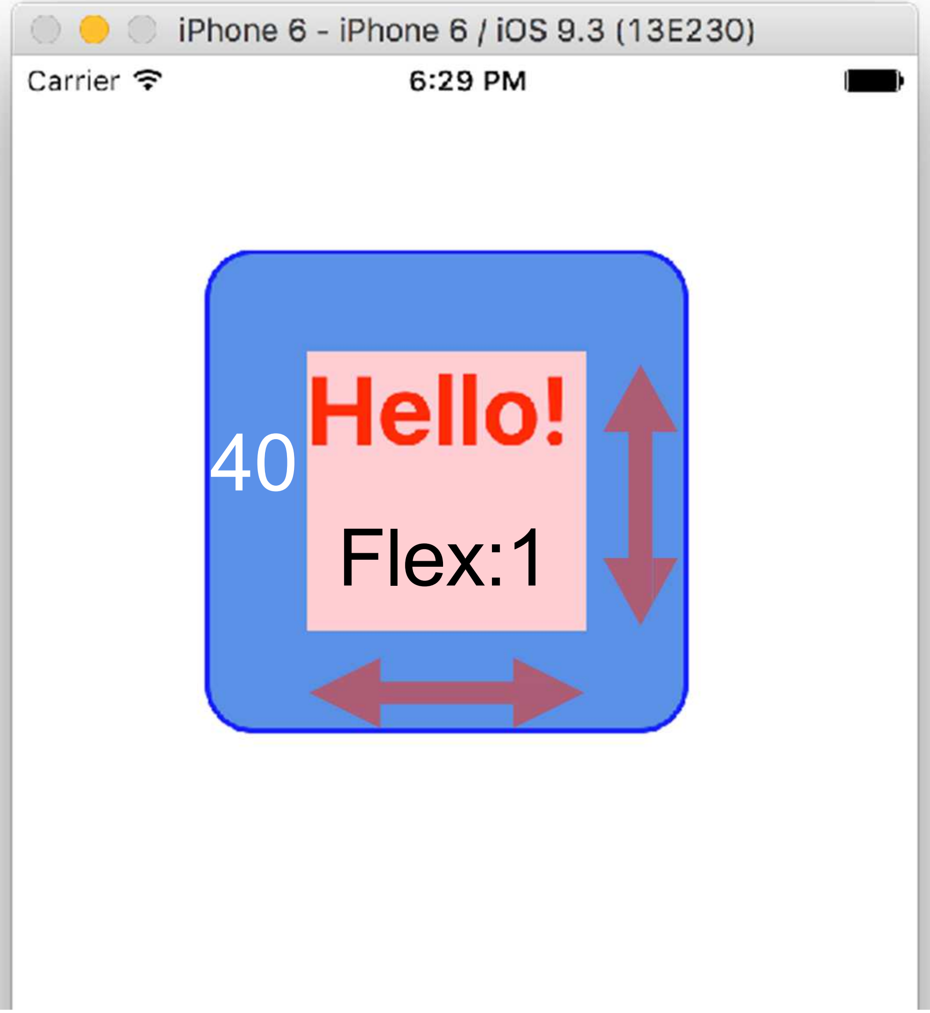


```
const style = {
    width: 200,
    height: 200,
    backgroundColor: 'rgb(74,124,226)',
    borderWidth: 2,
    borderColor: 'blue',
    borderRadius: 20,
    padding: 40,
    margin: 80
}
return (
    <View style={style}>
        <View style={boxStyle}>
            <Text style={textStyle}>
                Hello!
            </Text>
        </View>
    </View>
);
```

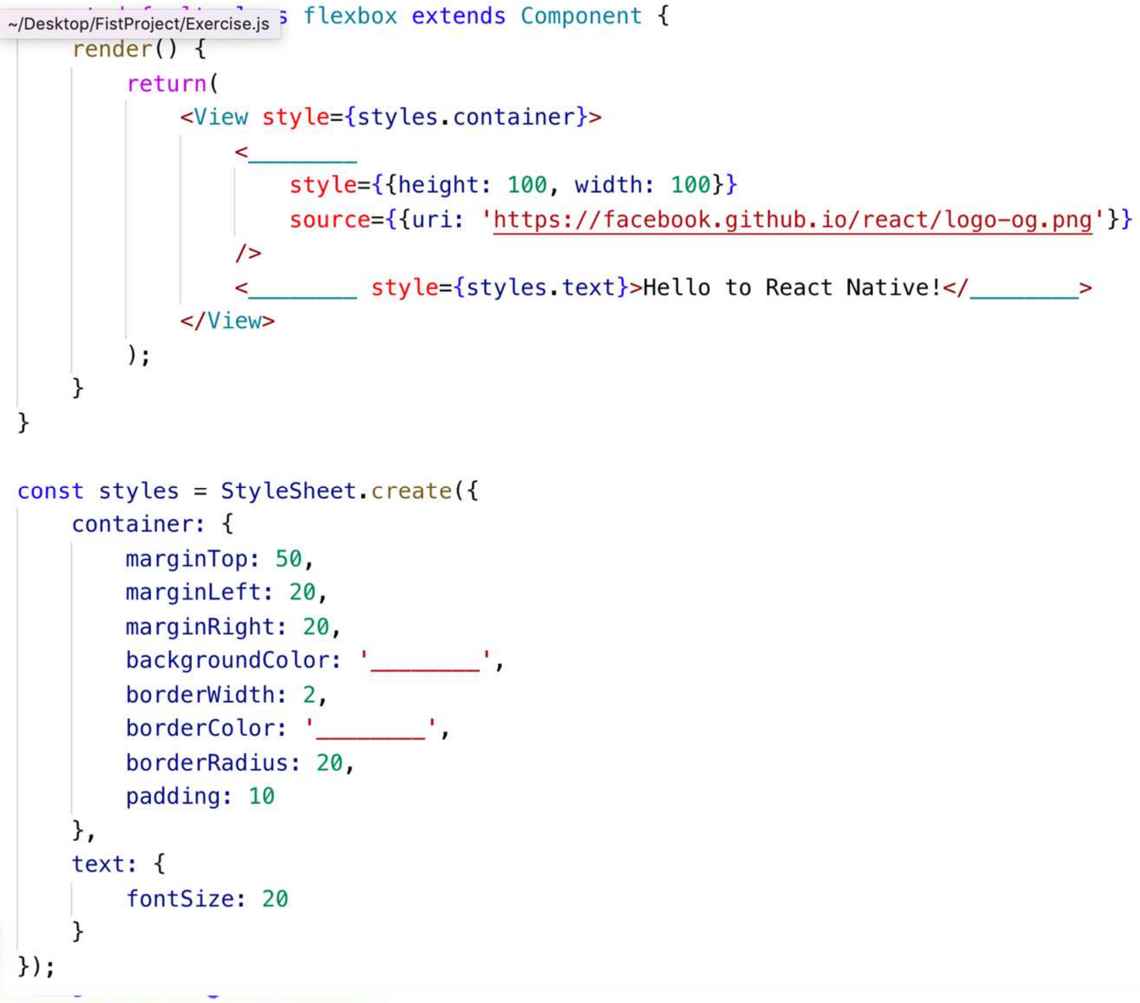


View: Red Box + Text

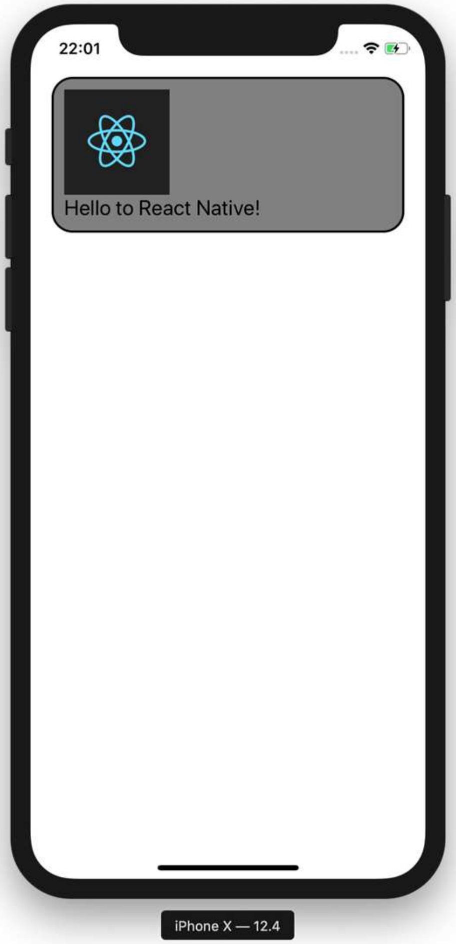
```
const boxStyle = {
    flex: 1,
    backgroundColor: 'pink'
}

const textStyle = {
    fontSize: 40,
    fontWeight: 'bold',
    color: 'red'
}

return (
    <View style={style}>
        <View style={boxStyle}>
            <Text style={textStyle}>
                Hello!
            </Text>
        </View>
    </View>
);
```



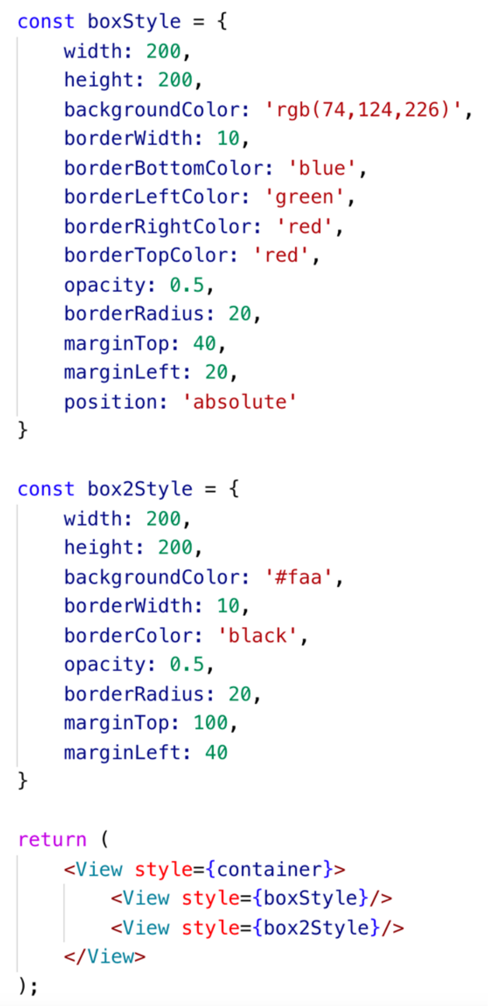
Exercise I (5 min)



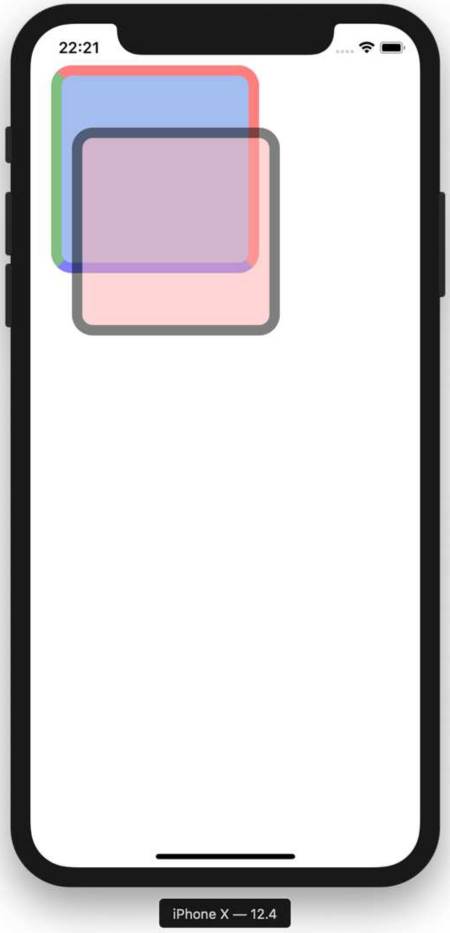
```
~/Desktop/FistProject/Exercise.js
s flexbox extends Component {
    render() {
        return(
            <View style={styles.container}>
                <________
                    style={{height: 100, width: 100}}
                    source={{uri: 'https://facebook.github.io/react/logo-og.png'}}
                />
                <________ style={styles.text}>Hello to React Native!</________>
            </View>
        );
    }
}

const styles = StyleSheet.create({
    container: {
        marginTop: 50,
        marginLeft: 20,
        marginRight: 20,
        backgroundColor: '________',
        borderWidth: 2,
        borderColor: '________',
        borderRadius: 20,
        padding: 10
    },
    text: {
        fontSize: 20
    }
});
```



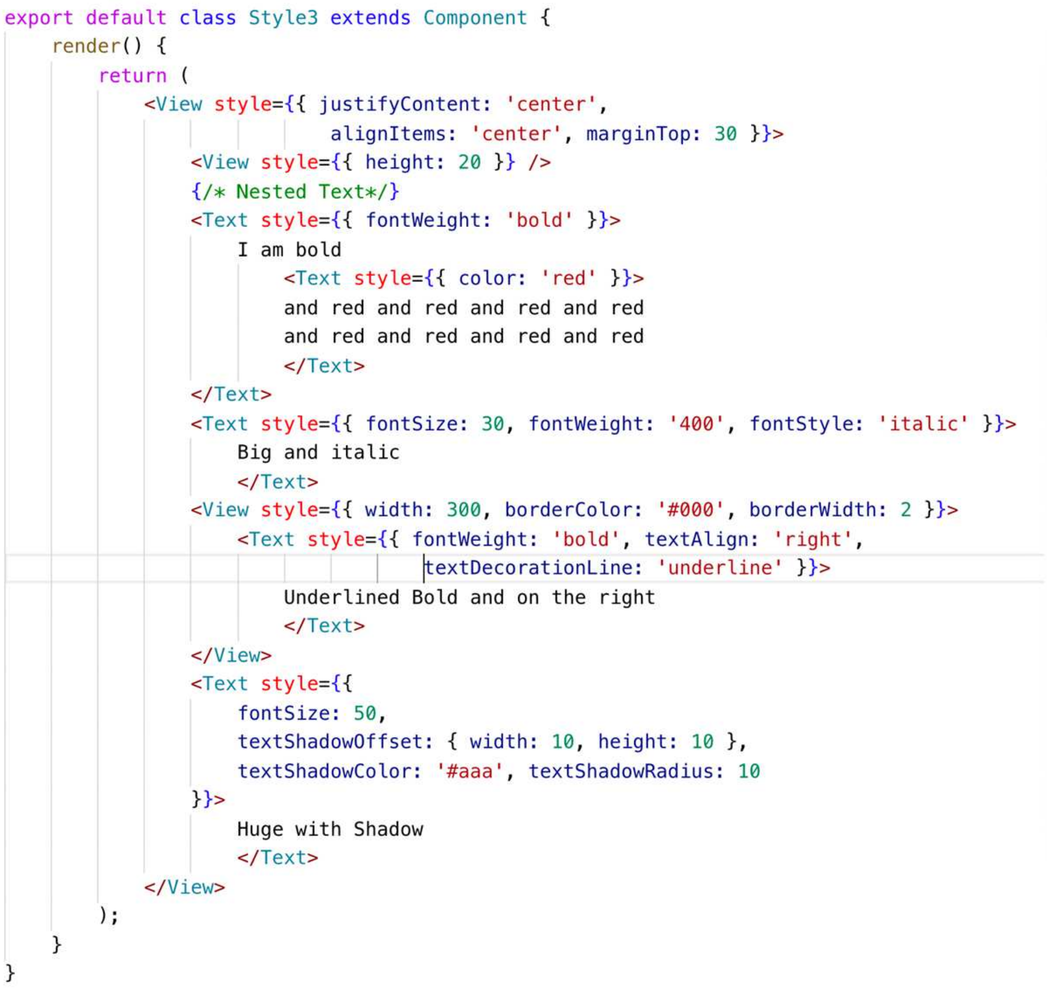
Position



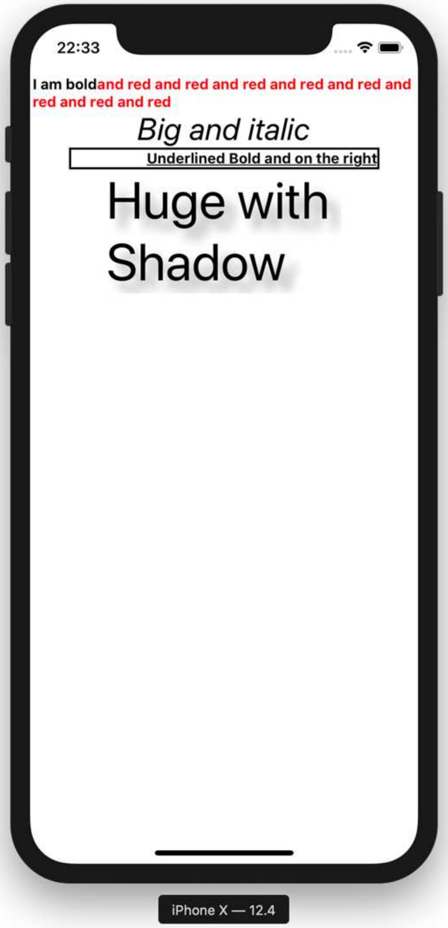
```
const boxStyle = {
    width: 200,
    height: 200,
    backgroundColor: 'rgb(74,124,226)',
    borderWidth: 10,
    borderBottomColor: 'blue',
    borderLeftColor: 'green',
    borderRightColor: 'red',
    borderTopColor: 'red',
    opacity: 0.5,
    borderRadius: 20,
    marginTop: 40,
    marginLeft: 20,
    position: 'absolute'
}

const box2Style = {
    width: 200,
    height: 200,
    backgroundColor: '#faa',
    borderWidth: 10,
    borderColor: 'black',
    opacity: 0.5,
    borderRadius: 20,
    marginTop: 100,
    marginLeft: 40
}

return (
    <View style={container}>
        <View style={boxStyle}/>
        <View style={box2Style}/>
    </View>
);
```



Nest Text



```
export default class Style3 extends Component {
    render() {
        return (
            <View style={{ justifyContent: 'center',
                            alignItems: 'center', marginTop: 30 }}>
                <View style={{ height: 20 }} />
                {/* Nested Text*/}
                <Text style={{ fontWeight: 'bold' }}>
                    I am bold
                        <Text style={{ color: 'red' }}>
                        and red and red and red and red
                        and red and red and red and red
                        </Text>
                </Text>
                <Text style={{ fontSize: 30, fontWeight: '400', fontStyle: 'italic' }}>
                    Big and italic
                    </Text>
                <View style={{ width: 300, borderColor: '#000', borderWidth: 2 }}>
                    <Text style={{ fontWeight: 'bold', textAlign: 'right',
                                    textDecorationLine: 'underline' }}>
                        Underlined Bold and on the right
                        </Text>
                </View>
                <Text style={{
                    fontSize: 50,
                    textShadowOffset: { width: 10, height: 10 },
                    textShadowColor: '#aaa', textShadowRadius: 10
                }}>
                    Huge with Shadow
                    </Text>
            </View>
        );
    }
}
```



I am boldand red and red and red and red and red and red and red and red

Big and italic

Underlined Bold and on the right

Huge with Shadow

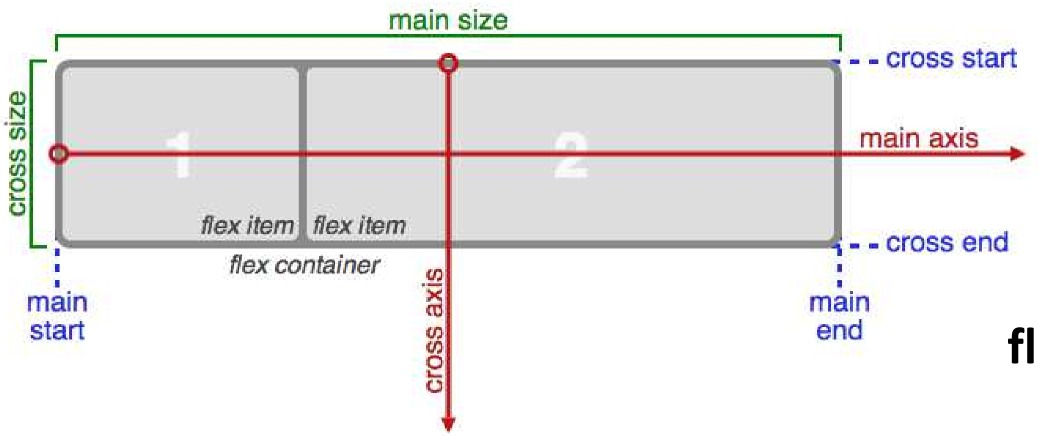
Flexbox Layout

- Flexbox => **CSS Flexible Box Layout** (in W3C Last Call Working Draft)
- Providing efficient way to layout, align and distribute space among items in a container, even when their size is unknown and/or dynamic (flex)
- **Containers** can alter its **items** width/height and order to best fill the available space.
- Flexbox is a direction-agnostic, which support complex applications (especially when it comes to orientation changing, resizing, stretching, shrinking, etc.)

https://reactnative.dev/docs/flexbox

Direction

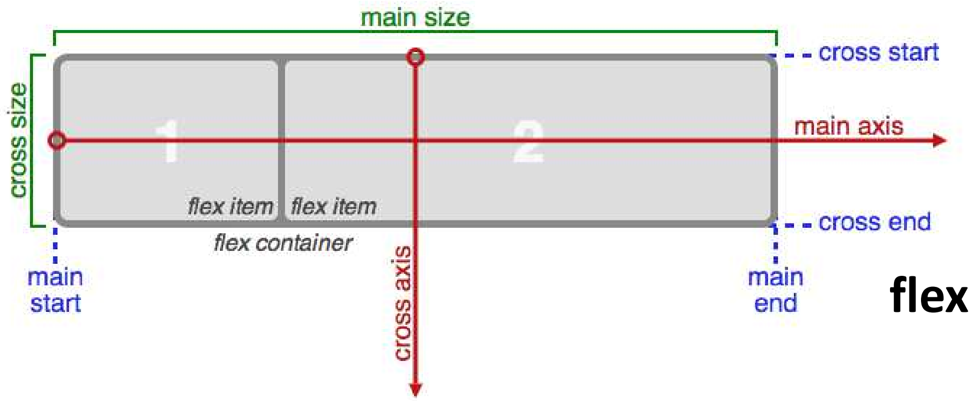
- **main axis** - Primary axis of a flex container, defined by flexDirection
- **main-start | main-end** — Flex items placed within container starting from main-start and going to main-end
- **main-size** - Flex item’s width or height, whichever is in the primary dimension.



flexDirection = row (horizontal)

Direction

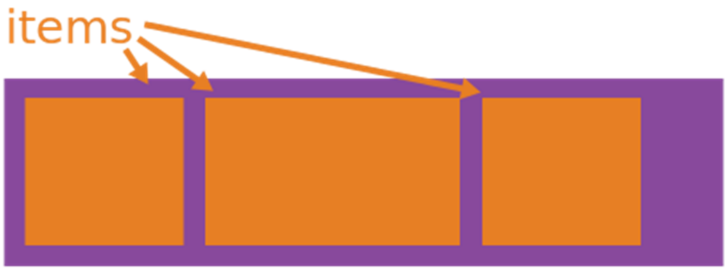
- **cross axis** - Secondary axis that perpendicular to the primary axis (**opposed** with the **flexDirection**)
- **cross-start | cross-end** - Flex lines are filled with items and placed into the container starting on the cross-start side or on the cross-end side
- **cross-size** - the flex item’s width or height, whichever is in the cross dimension.

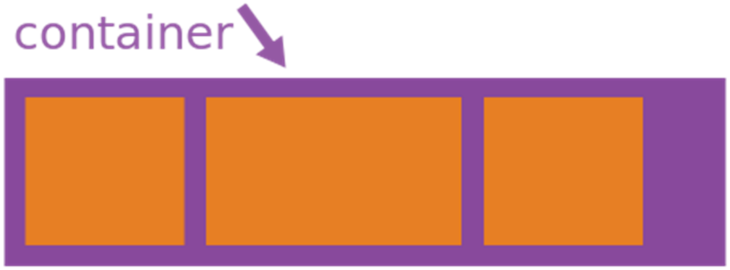


flexDirection = row (horizontal)

Two types of Flex properties

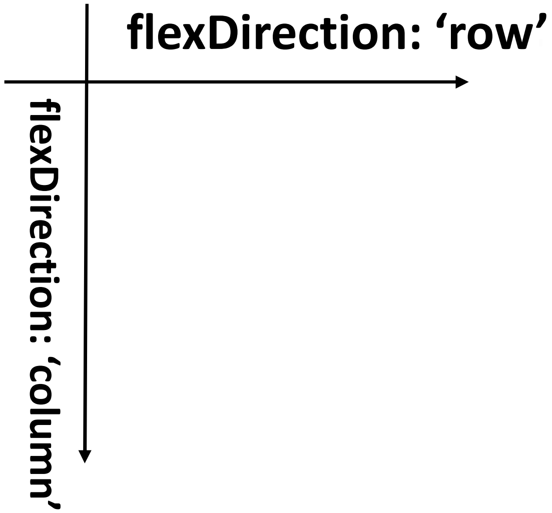
- flexDirection
- justifyContent
- alignItems
- flexWrap





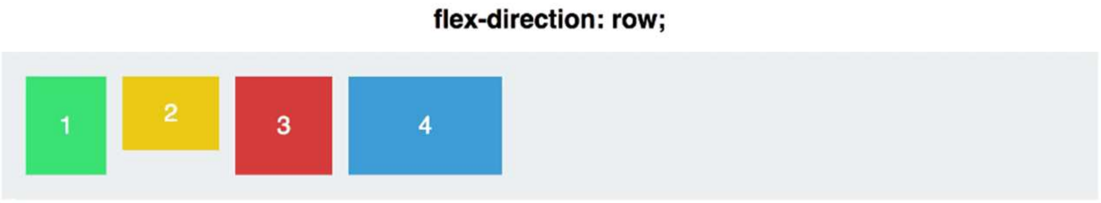
- flex
- alignSelf

flexDirection



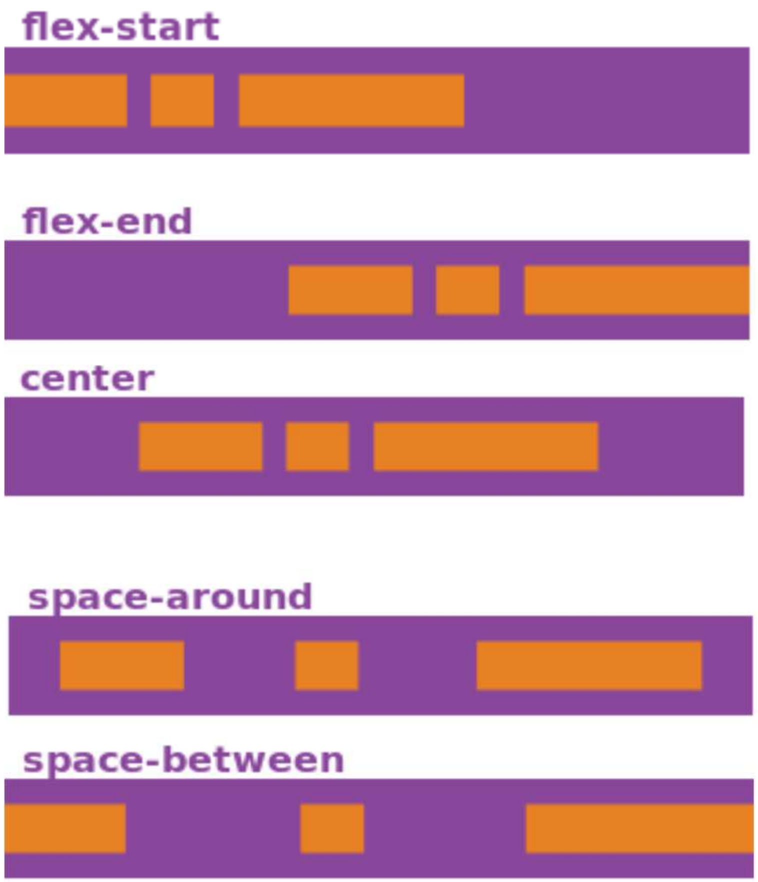
flexDirection: ‘row’

flexDirection: ‘column’



flex-direction: row;

justifyContent



justifyContent: ‘flex-start’

justifyContent: ‘flex-end’

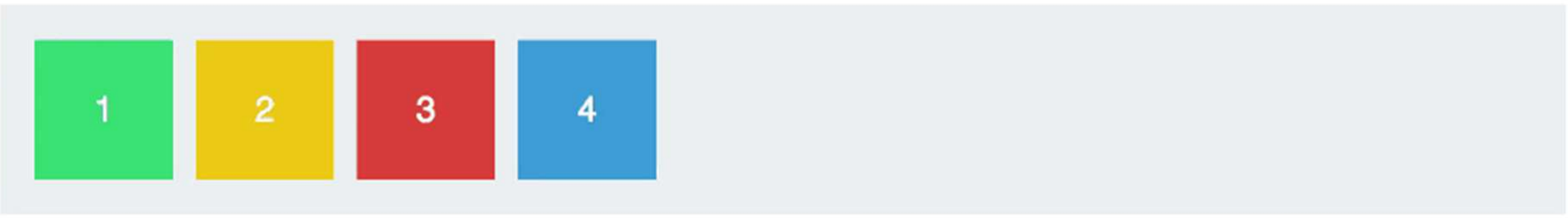
justifyContent: ‘center’

justifyContent: ‘space-around’

justifyContent: ‘space-between’

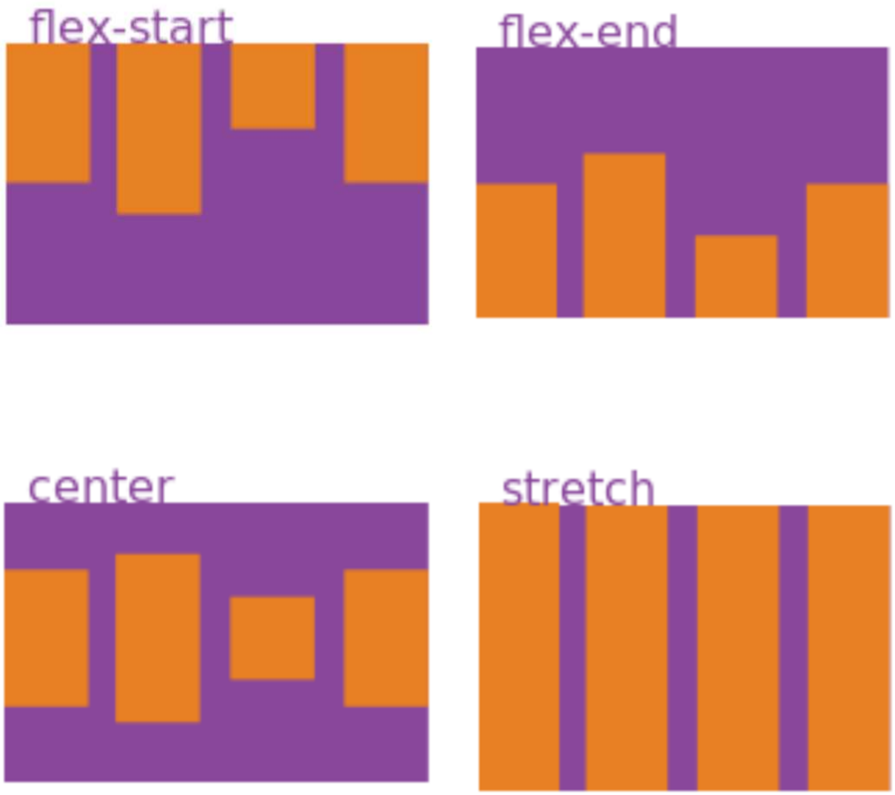
justifyContent

- Adding **justifyContent** to a component's style determines the distribution of children along the primary axis
- Should children be distributed at the start, the center, the end, or spaced evenly?
- Available options are `flex- start`, `center`, `flex- end`, `space-around`, and `space-between`
- Default is `flex-start`


justify-content: flex-start;

alignItems

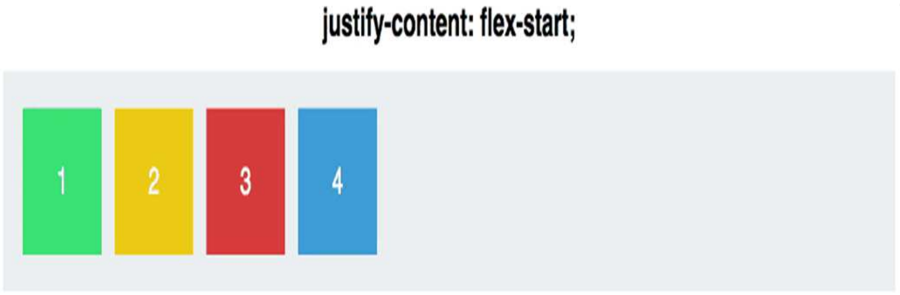
- Adding **alignItems** to a component's style determines the alignment of children along the secondary axis (if the primary axis is row, then the secondary is column, and vice versa)
- Should children be aligned at the start, the center, the end, or stretched to fill?
- Available options are `flex-start`, `center`, `flex-end`, and `stretch`
- Default is `flex-start`

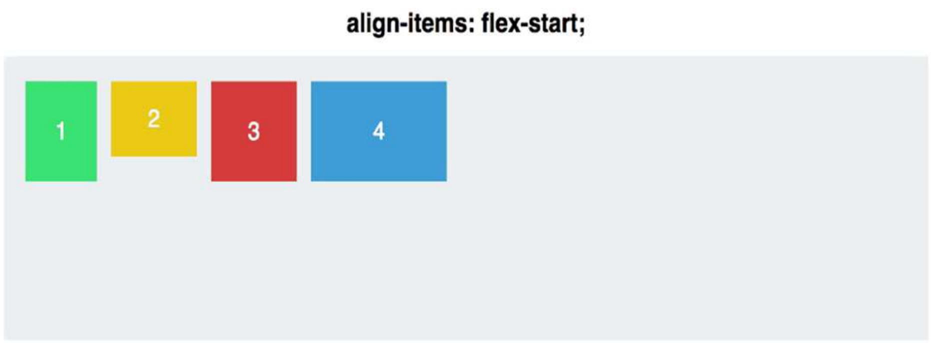


alignItems and justifyContent

- **justifyContent**

- **alignItems**





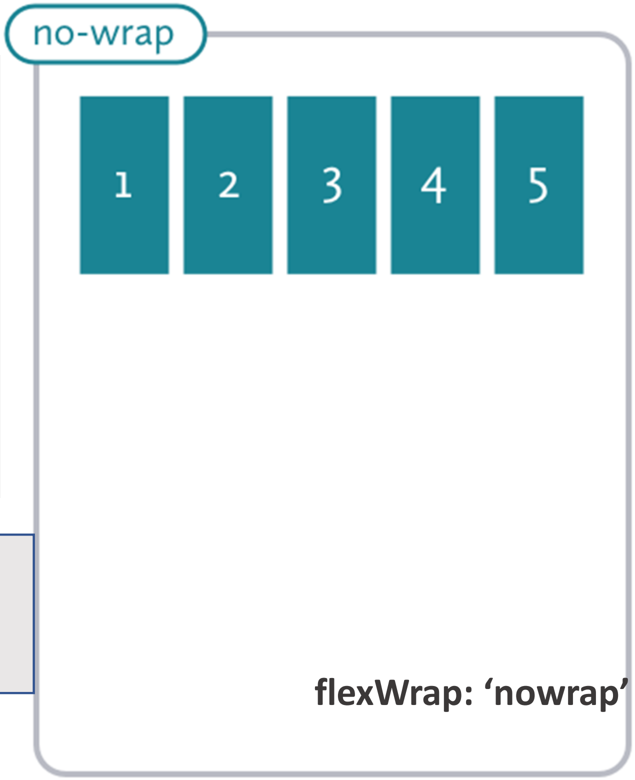
Main axis

Cross axis

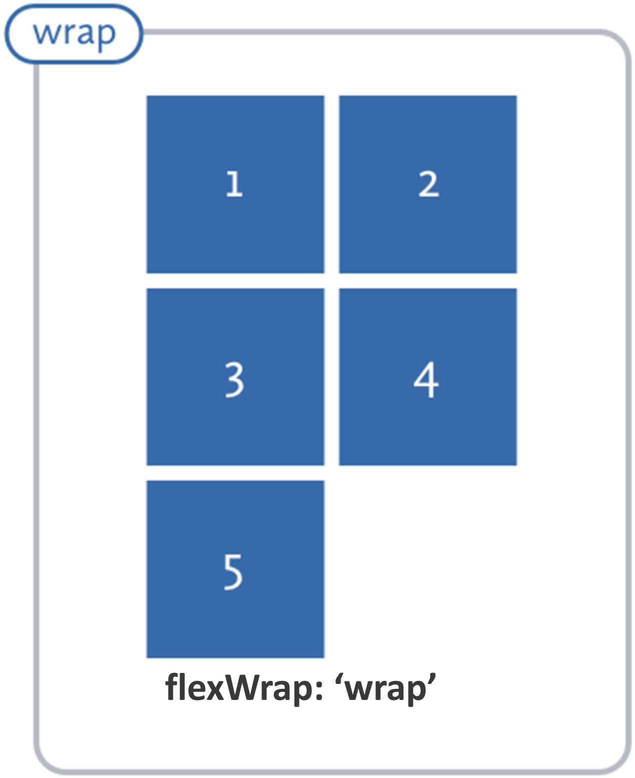
flexWrap

The initial flexbox concept is the container to set its items in one single line.
The flex-wrap property controls if the flex container lay out its items in single or multiple lines, and the direction the new lines are stacked in.

Default: nowrap



flexWrap: ‘nowrap’

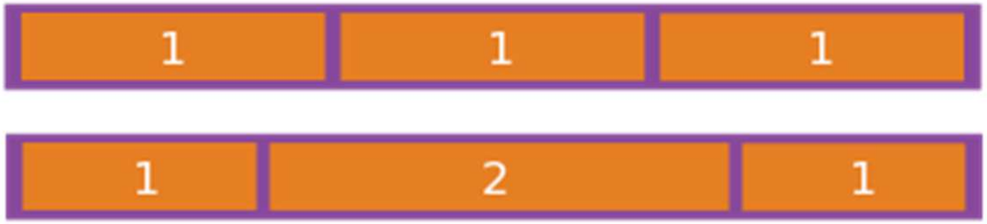


flexWrap: ‘wrap’

flex

The flex property sets the flexible length on flexible items.

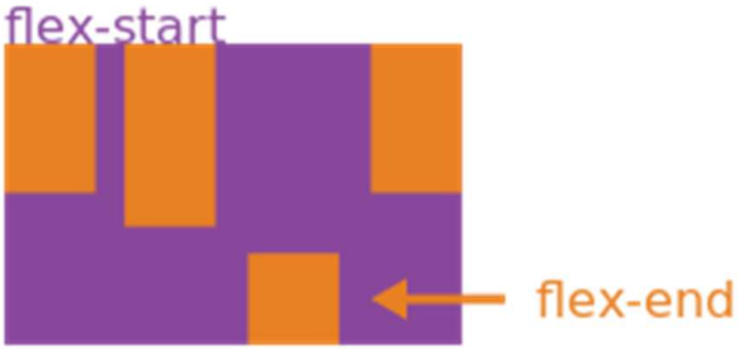
Note: If the element is not a flexible item, the flex property has no effect.



```
<View>
    <View style={[style.item1, {flex:1}]}></View>
    <View style={[style.item1, {flex:2}]}></View>
    <View style={[style.item1, {flex:1}]}></View>
</View>
```

alignSelf

Note: The alignSelf property overrides the flexible container's alignItems property.



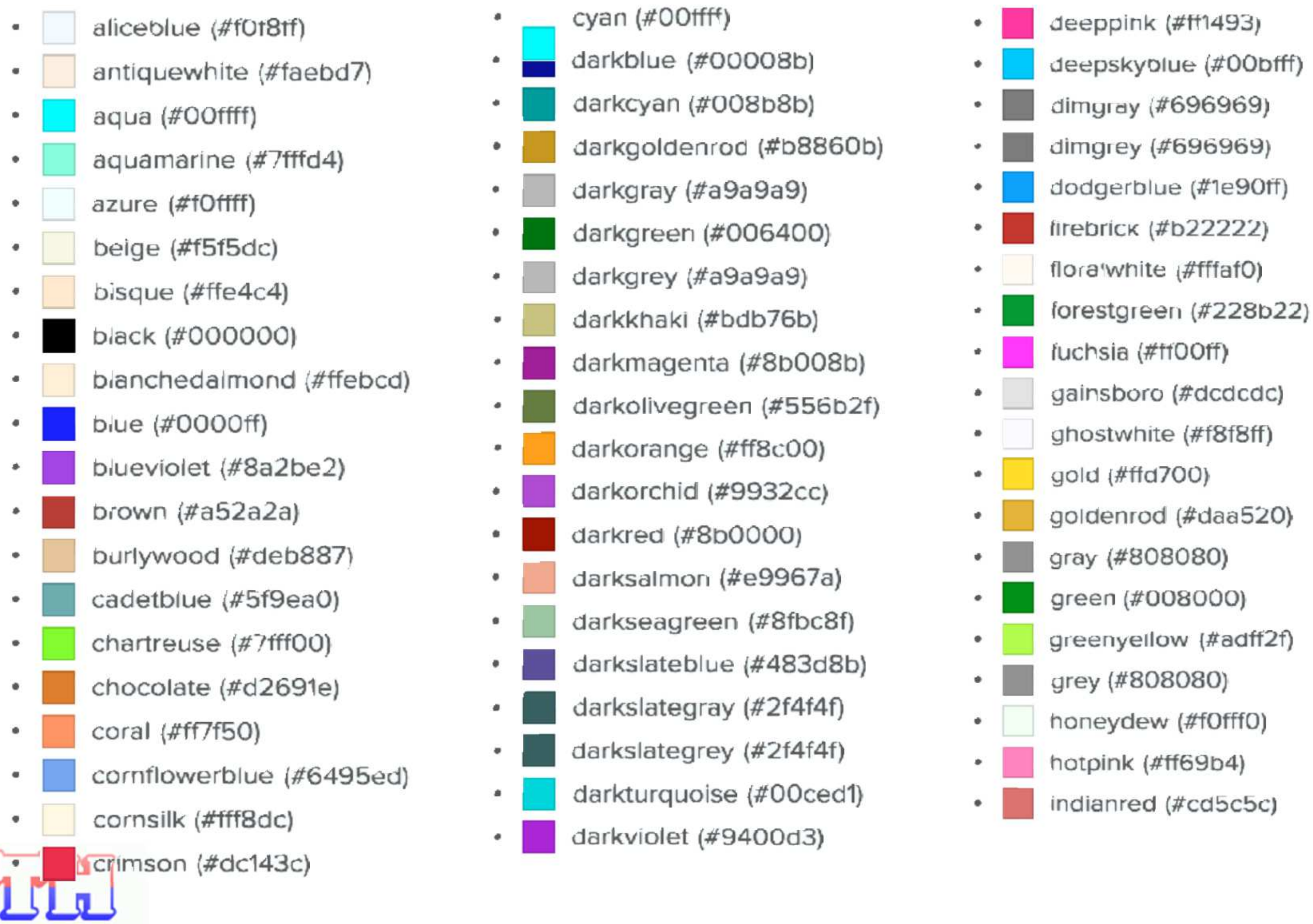
```
<View>
  <View style={[styles.container, {alignItems: flex-start}]}/>
  <View style={[styles.item3, {alignSelf: flex-end}]}/>
</View>
```

Colors

- '#f0f' (#rgb)
- '#f0fc' (#rgba)
- '#ff00ff' (#rrggbb)
- '#ff00ff00' (#rrggbbaa)
- 'rgb(255, 255, 255)'
- 'rgba(255, 255, 255, 1.0)'

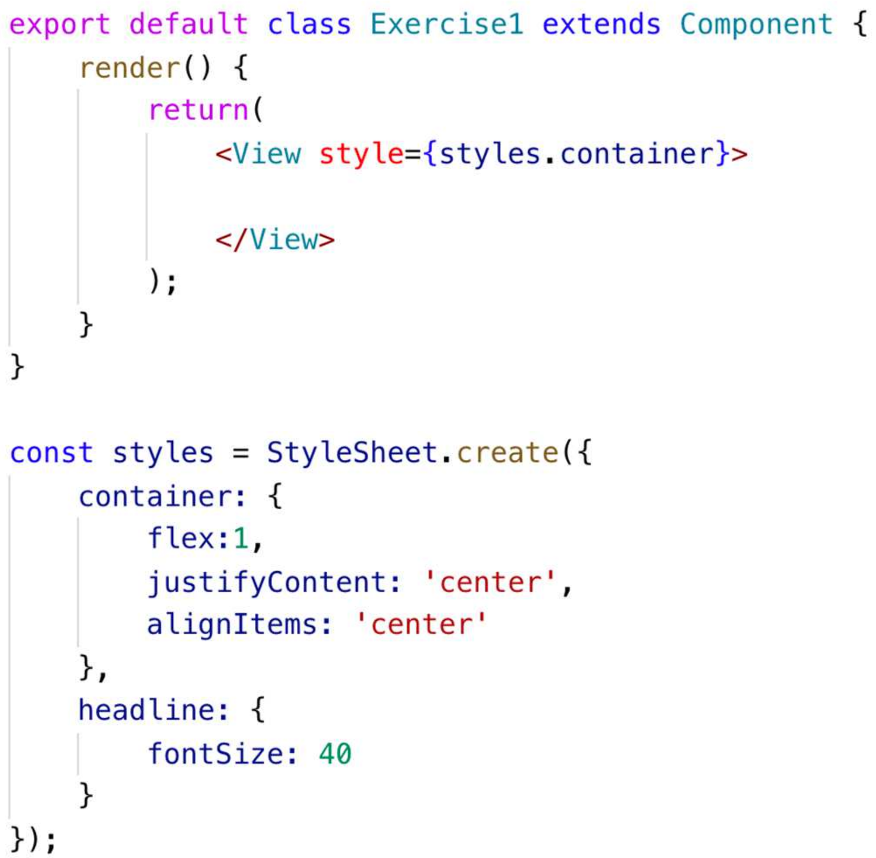
- hsl (360, 100%, 100%)
- 'hsla(360, 100%, 100%, 1)’
- 'Transparent’
- 'red’
- 0xff00ff00 (0xrrggbbaa)

More Colors…

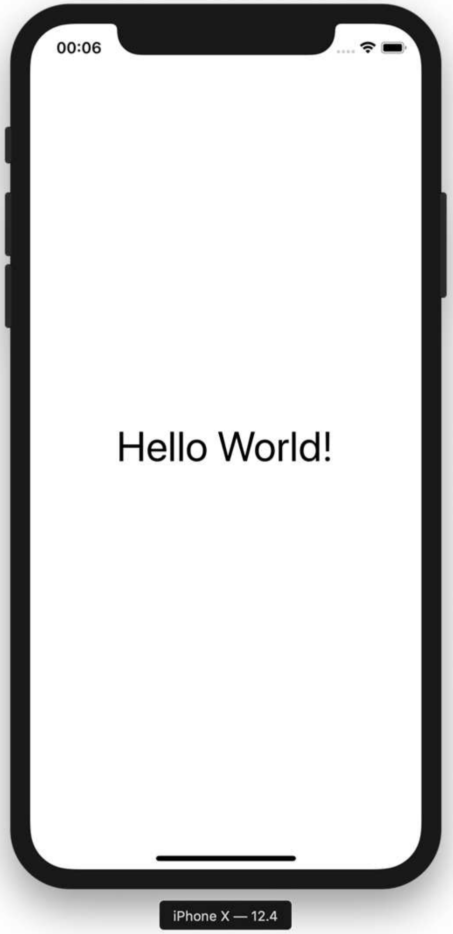


- aliceblue (#f0f8ff)
- antiquewhite (#faebd7)
- aqua (#00ffff)
- aquamarine (#7fffd4)
- azure (#f0ffff)
- beige (#f5f5dc)
- bisque (#ffe4c4)
- black (#000000)
- blanchedalmond (#ffebcd)
- blue (#0000ff)
- blueviolet (#8a2be2)
- brown (#a52a2a)
- burlywood (#deb887)
- cadetblue (#5f9ea0)
- chartreuse (#7fff00)
- chocolate (#d2691e)
- coral (#ff7f50)
- cornflowerblue (#6495ed)
- cornsilk (#fff8dc)
- crimson (#dc143c)
- cyan (#00ffff)
- darkblue (#00008b)
- darkcyan (#008b8b)
- darkgoldenrod (#b8860b)
- darkgray (#a9a9a9)
- darkgreen (#006400)
- darkgrey (#a9a9a9)
- darkkhaki (#bdb76b)
- darkmagenta (#8b008b)
- darkolivegreen (#556b2f)
- darkorange (#ff8c00)
- darkorchid (#9932cc)
- darkred (#8b0000)
- darksalmon (#e9967a)
- darkseagreen (#8fbc8f)
- darkslateblue (#483d8b)
- darkslategray (#2f4f4f)
- darkslategrey (#2f4f4f)
- darkturquoise (#00ced1)
- darkviolet (#9400d3)
- deeppink (#ff1493)
- deepskyblue (#00bfff)
- dimgray (#696969)
- dimgrey (#696969)
- dodgerblue (#1e90ff)
- firebrick (#b22222)
- floralwhite (#fffaf0)
- forestgreen (#228b22)
- fuchsia (#ff00ff)
- gainsboro (#dcdcdc)
- ghostwhite (#f8f8ff)
- gold (#ffd700)
- goldenrod (#daa520)
- gray (#808080)
- green (#008000)
- greenyellow (#adff2f)
- grey (#808080)
- honeydew (#f0fff0)
- hotpink (#ff69b4)
- indianred (#cd5c5c)

Exercise II (10 min)

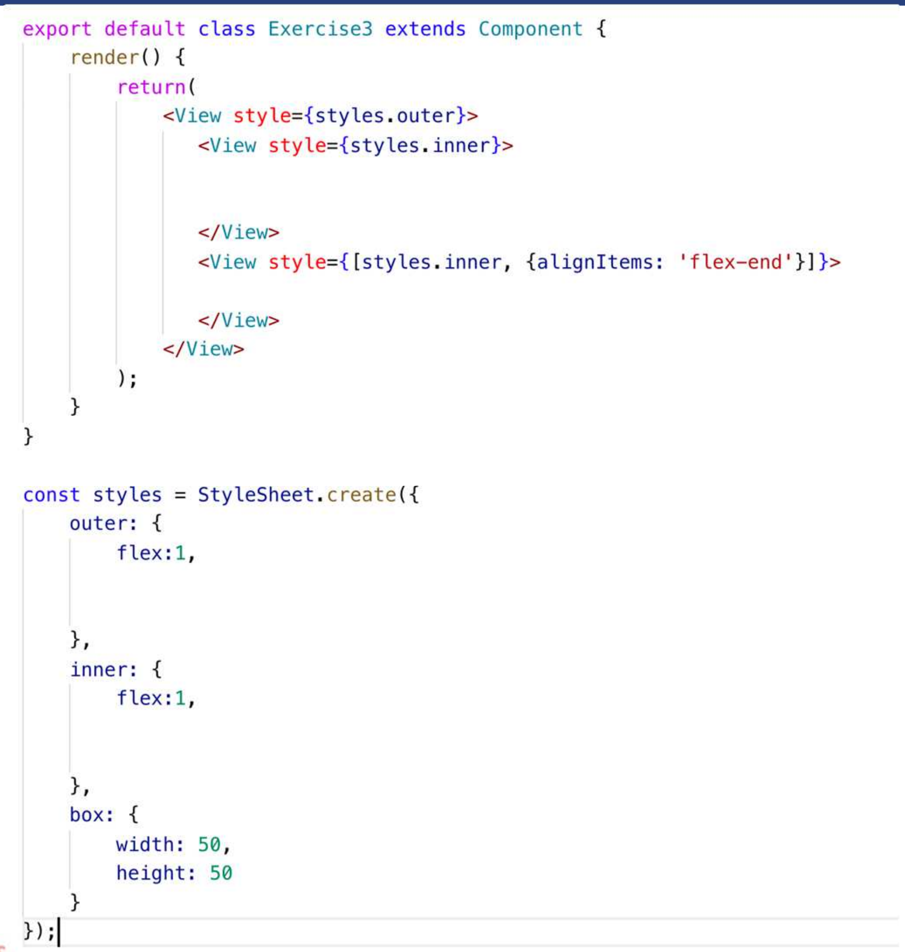


```
export default class Exercise1 extends Component {
    render() {
        return(
            <View style={styles.container}>

            </View>
        );
    }
}

const styles = StyleSheet.create({
    container: {
        flex:1,
        justifyContent: 'center',
        alignItems: 'center'
    },
    headline: {
        fontSize: 40
    }
});
```

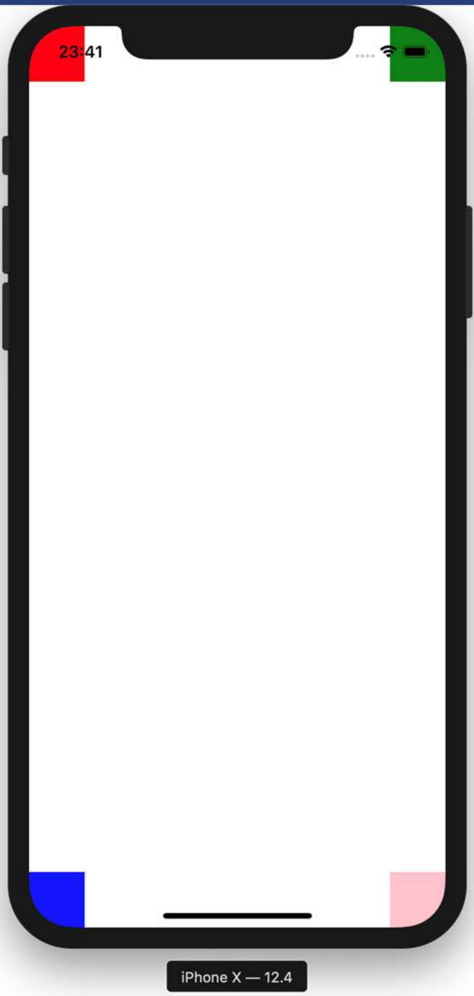


Hello World!

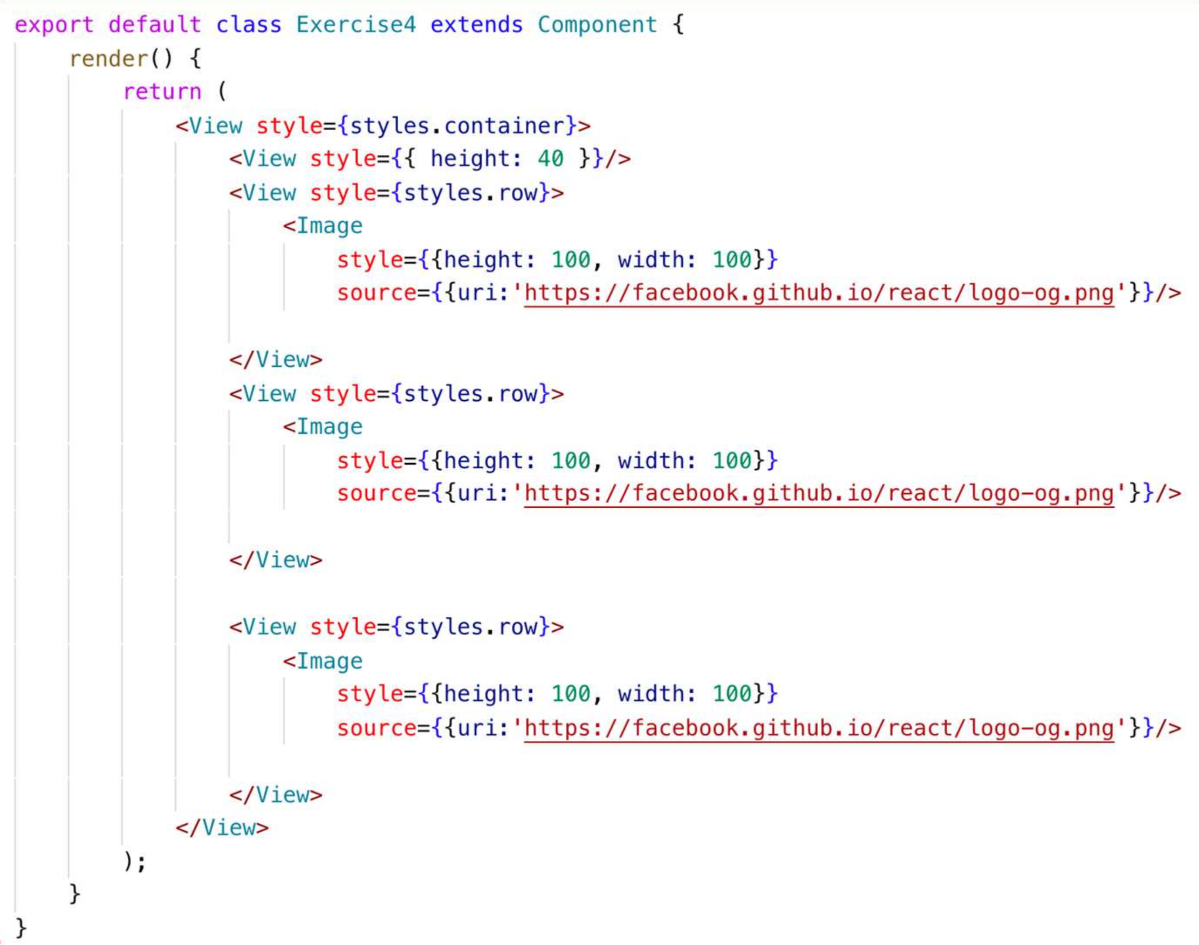
Exercise III (15 min)



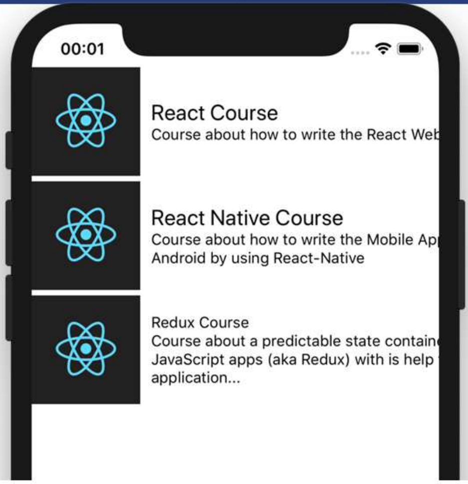
```
export default class Exercise3 extends Component {
    render() {
        return(
            <View style={styles.outer}>
               <View style={styles.inner}>


               </View>
               <View style={[styles.inner, {alignItems: 'flex-end'}]}>

               </View>
            </View>
        );
    }
}

const styles = StyleSheet.create({
    outer: {
        flex:1,


    },
    inner: {
        flex:1,


    },
    box: {
        width: 50,
        height: 50
    }
});
```



Exercise IV (15 min)



```
export default class Exercise4 extends Component {
    render() {
        return (
            <View style={styles.container}>
                <View style={{ height: 40 }}/>
                <View style={styles.row}>
                    <Image
                        style={{height: 100, width: 100}}
                        source={{uri:'https://facebook.github.io/react/logo-og.png'}}/>

                </View>
                <View style={styles.row}>
                    <Image
                        style={{height: 100, width: 100}}
                        source={{uri:'https://facebook.github.io/react/logo-og.png'}}/>

                </View>

                <View style={styles.row}>
                    <Image
                        style={{height: 100, width: 100}}
                        source={{uri:'https://facebook.github.io/react/logo-og.png'}}/>

                </View>
            </View>
        );
    }
}
```

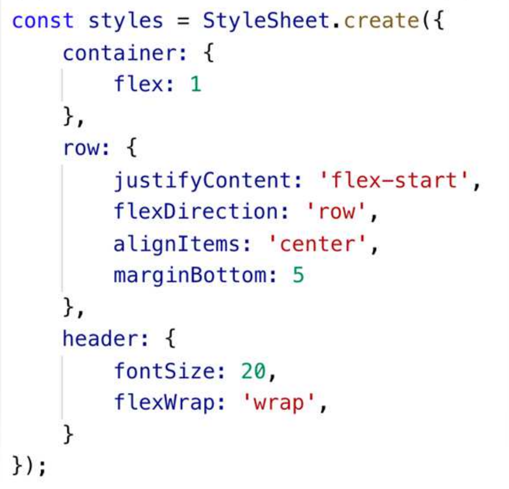


00:01

React Course
Course about how to write the React Web

React Native Course
Course about how to write the Mobile Ap
Android by using React-Native

Redux Course
Course about a predictable state contain
JavaScript apps (aka Redux) with is help
application...



```
const styles = StyleSheet.create({
    container: {
        flex: 1
    },
    row: {
        justifyContent: 'flex-start',
        flexDirection: 'row',
        alignItems: 'center',
        marginBottom: 5
    },
    header: {
        fontSize: 20,
        flexWrap: 'wrap',
    }
});
```

References

- https://reactnative.dev/docs/style

- https://docs.expo.dev/versions/latest/react-native/stylesheet

- https://www.bigbinary.com/learn-react-native

Thank you!