

# Learning React Native Basics

---



**Reggie Dawson**

DEVELOPER

@reggiewriteres



# Overview



**View Component**

**Props and State**

**Layout**

**Navigation**



```
headStyle:{  
  ...Platform.select({  
    ios: {  
      height: 300  
    },  
    android: {  
      height: 250  
    }  
  }  
}),  
}
```

◀ Platform Select method

◀ Will return height based on platform



# Return Component Based on Platform

```
const MyComponent = Platform.select({  
  ios: () => require('IOSHome'),  
  android: () => require('AndroidHome'),  
}());
```

```
<MyComponent />;
```



```
this.state = {  
  isLoggedIn: false,  
  theVersion: Platform.Version  
};
```

◀ **Storing the platform version in state**



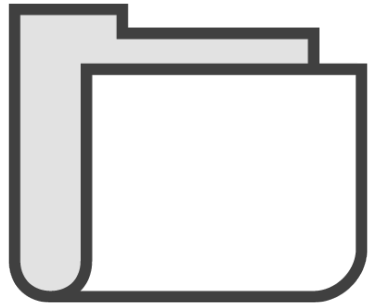
`Home.ios.js`

`Home.android.js`

◀ iOS component

◀ Android component





**Xcode Assets Catalog**

**Android Drawable Folder**

**Android Assets Folder**

**Use these if you eject your app**



```
<Image
```

```
    source={{uri: 'https://my-remote-site.com/myPic.png'}}  
    style={{width: 400, height: 400}}
```

```
/>
```

---

## Network Images

**Images from remote sources**

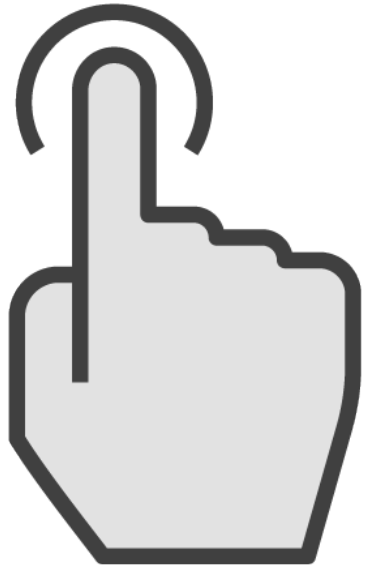




```
<Image  
source={{uri:'data:image/png;base64,iVBORw0KGgoAAAANSUhEUgA  
AADMAAAAzCAYAAAA6oTAqAAAAEXRFWHRTb2Z0d2FyZQBwbmdjcnVzaEB1Sf  
MAAABQSURBVGje7dSxCQBACAR2ab8BEeQNhFi6WSYzYLYudDQYGBgYGBgYG  
BgYGBgYGBgZmcvDqYGBgmhivGQYGBgYGBgYGBgYGBgYGBgbmQw+P/eMrC5U  
TVAAAAABJRU5ErkJgg=='}}  
  
style={{width: 400, height: 400}} />
```

# Data URI

## Encoded images



---

## Responding to Touch

### Gesture Responder system



# Button

```
<Button  
    onPress={someMethod}  
    title="Sample Button"  
    color="#ffffff"  
    accessibilityLabel="For Screen Readers"  
/>
```



```
<TouchableHighlight  
  onPress={this._onPressButton}>  
  <Image source={require( './myButton.png' )} />  
</TouchableHighlight>
```

---

# TouchableHighlight

**Darken on touch**



```
<TouchableNativeFeedback onPress={this._onPressButton}>  
  <View style={{width: 150, height: 100}}>  
    <Text>Button</Text>  
  </View>  
</TouchableNativeFeedback>
```

---

# TouchableNativeFeedback

**Ripples when touched**



```
<TouchableWithoutFeedback onPress={this._onPressButton}>  
  <View>  
    <Text>No Feedback</Text>  
  </View>  
</TouchableWithoutFeedback>
```

---

TouchableWithoutFeedback

**No response to touch**



```
<TouchableOpacity onPress={this._onPressButton}>  
<Image source={require(' ./myButton.png' )} />  
</TouchableOpacity>
```

---

# TouchableOpacity

**Fade on touch**



# Summary



**Props and State**

**Layout**

**Touch Events**

**Navigation**

