

ASSIGNMENT

Page 1: Animation of Balloon being inflated on clicking a button

Route: /balloon-animation

```
import React, { useState } from 'react';
import { View, Text, StyleSheet, TouchableOpacity, Animated } from 'react-native';
```

```
const BalloonAnimation = () => {
  const [animation, setAnimation] = useState(new Animated.Value(0));
  const handlePress = () => {
    Animated.timing(animation, {
      toValue: 1,
      duration: 1000,
      useNativeDriver: false,
    }).start();
  };
};
```

```
const balloonSize = animation.interpolate({
  inputRange: [0, 1],
  outputRange: [150, 250],
});
```

```
return (
  <View style={styles.container}>
    <TouchableOpacity onPress={handlePress}>
      <Animated.View style={[styles.balloon, { height: balloonSize }] />
    </TouchableOpacity>
  </View>
);
```

```
const styles = StyleSheet.create({
  container: {
```

```

    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#fff',
  },
  balloon: {
    width: 100,
    borderRadius: 50,
    backgroundColor: 'red',
    marginBottom: 10,
  },
});

export default BalloonAnimation;

```

Page 2: Form with Email and Name fields, calling a basic API on submit

Route: /form-api-call

```

import React, { useState } from 'react';
import { View, Text, TextInput, TouchableOpacity, StyleSheet } from 'react-native';

const FormApiCall = () => {
  const [name, setName] = useState("");
  const [email, setEmail] = useState("");

  const handleSubmit = () => {
    const data = { name, email };
    fetch('https://example.com/api', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json',
      },
      body: JSON.stringify(data),
    })
  }
}

```

```

        .then((response) => response.json())
        .then((data) => console.log(data))
        .catch((error) => console.error(error));
    };

    return (
      <View style={styles.container}>
        <TextInput
          style={styles.input}
          placeholder="Name"
          value={name}
          onChangeText={(text) => setName(text)}
        />
        <TextInput
          style={styles.input}
          placeholder="Email"
          value={email}
          onChangeText={(text) => setEmail(text)}
        />
        <TouchableOpacity style={styles.button} onPress={handleSubmit}>
          <Text style={styles.buttonText}>Submit</Text>
        </TouchableOpacity>
      </View>
    );
  };

```

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#fff',
  },
  input: {

```

```
width: '80%',  
height: 40,  
borderWidth: 1,  
borderRadius: 5,  
paddingHorizontal: 10,  
marginBottom: 20,  
},  
button: {  
  backgroundColor: 'blue',  
  paddingVertical: 10,  
  paddingHorizontal: 20,  
  borderRadius: 5,  
},  
buttonText: {  
  color: '#fff',  
  fontWeight: 'bold',  
},  
});  
  
export default FormApiCall;
```