

APCSP\App.js

```
import React from 'react';
import { StyleSheet, Text, View, Button, TextInput, SafeAreaView, Keyboard,
TouchableWithoutFeedback, FlatList, ScrollView } from 'react-native';

let nextId = 1;

const Separator = () => <View style={styles.separator} />; //generate seperator

//calculate bmi
const BMI = (height, weight) => {
  height = height / 100
  return Math.round(weight / (height * height));
}

//determine status based on bmi
const STATUS = (b) => {
  if (b < 19) {
    return ("Underweight");
  }
  else if (b < 25) {
    return ("Healthy");
  }
  else if (b < 30) {
    return ("Overweight");
  }
  else if (b < 40) {
    return ("Obese");
  }
  else {
    return ("Extremely obese");
  }
}

export default function App() {
  //variables
  const [bmi, setBMI] = React.useState(NaN);
  const [name, onChangeName] = React.useState("");
  const [age, onChangeAge] = React.useState("");
  const [status, setStatus] = React.useState(NaN);
  const [height, onChangeHeight] = React.useState('');
  const [weight, onChangeWeight] = React.useState('');
  const [users, setUser] = React.useState([]);

  //change variables
  function handleBMI(b) {
    setBMI(b);
  }

  function handleStatus(s) {
    setStatus(s);
  }
}
```

```

function handleResult(h, w, n, a) {
  b = BMI(h, w);
  s = STATUS(b)
  handleBMI(b);
  handleStatus(s);
  setUser([...users, { id: nextId++, bmi: b, status: s, name: n, age: a, height: h, weight: w
}]);
}
// const user = [{
//   Id: 100,
//   Bmi: 30,
//   status: 'ok',
//   Name: 'hello',
//   Age: 14,
//   Height: 160,
//   Weight: 54,
// }]

//design list view
const Item = ({ user }) => (
  <Text style={styles.baseText}> {user.id}. {user.name} ({user.age})
    | W: {user.weight}, H: {user.height} BMI: {user.bmi}| {user.status} </Text>
);

//show results
function showResult(b, s) {
  if (b > 0) {
    return (
      <View>
        <Text style={styles.titleText}>Your BMI is: {b}</Text>
        <Text style={styles.titleText}>Body status: {s}</Text>
      </View>
    );
  }
  else {
    return (<Text>Input the measures and click "Show results" to see your index.</Text>);
  }
}

//design the UI
return (
  <SafeAreaView style={styles.container}>
    <View style={styles.container}>
      <Text style = {styles.title}>BMI Calculator</Text>
      <Text style={styles.titleText}>Calculate your BMI here!</Text>
      <Text style={styles.baseText}>
        Utilize this app to calculate your body mass index in a split second.
      </Text>
      <Separator />
      <Text style={styles.baseText}>Basic information:</Text>
      <Text></Text>
      <TouchableWithoutFeedback onPress={Keyboard.dismiss} accessible={false}>
        <View>
          <View style={styles.container}>
            <Text>Name:</Text>
            <TextInput
              style={styles.input}

```

```

        onChangeText={onChangeName}
        value={name}
        placeholder="name"
        keyboardType="text"
        onSubmitEditing={Keyboard.dismiss}
    />
    <Text>Age:</Text>
    <TextInput
        style={styles.input}
        onChangeText={onChangeAge}
        value={age}
        placeholder="age"
        keyboardType="numeric"
    />
</View>
<Text style={styles.baseText}>Please input your measures:</Text>
<Text></Text>
<View style={styles.container}>
    <Text>Weight:</Text>
    <TextInput
        style={styles.input}
        onChangeText={onChangeWeight}
        value={weight}
        placeholder="weight (kg)"
        keyboardType="numeric"
    />
    <Text>Height:</Text>
    <TextInput
        style={styles.input}
        onChangeText={onChangeHeight}
        value={height}
        placeholder="height (cm)"
        keyboardType="numeric"
    />
</View>
</View>
</TouchableWithoutFeedback>
<View style={styles.centered}>
    <Button
        title="Show results"
        onPress={() => handleResult(height, weight, name, age)}
    />
</View>
<Separator />
<View>
    <Text>Results: </Text>
    <Text></Text>
    <Text>{showResult(bmi, status)}</Text>
</View>
<Separator />
<View>
    <Text>History: </Text>
    <ScrollView style={styles.scrollView}>
        <View>
            <FlatList
                data={users}

```

```

        renderItem={({ item }) => <Item user={item} />}
        keyExtractor={item => item.Id} />
      </View>
    </ScrollView>
  </View>
</View>
</SafeAreaView>
);
}
//styles
const styles = StyleSheet.create({
  container: {
    marginHorizontal: 16,
  },
  title: {
    textAlign: 'center',
    marginVertical: 8,
    fontSize: 50,
  },
  fixToText: {
    flexDirection: 'row',
    justifyContent: 'space-between',
  },
  separator: {
    marginVertical: 8,
    borderBottomColor: '#737373',
    borderBottomWidth: StyleSheet.hairlineWidth,
  },
  baseText: {
    fontFamily: 'Cochin',
  },
  titleText: {
    fontSize: 20,
    fontWeight: 'bold',
  },
  input: {
    height: 40,
    margin: 12,
    borderWidth: 1,
    padding: 10,
  },
  centered: {
    alignItems: "center",
  },
});

```