INTRODUCTION TO MOBILE APPLICATIONS DEVELOPMENT

Introduction to React

ITSFT-406-2001

Ing. James Attard

OVERVIEW OF REACT

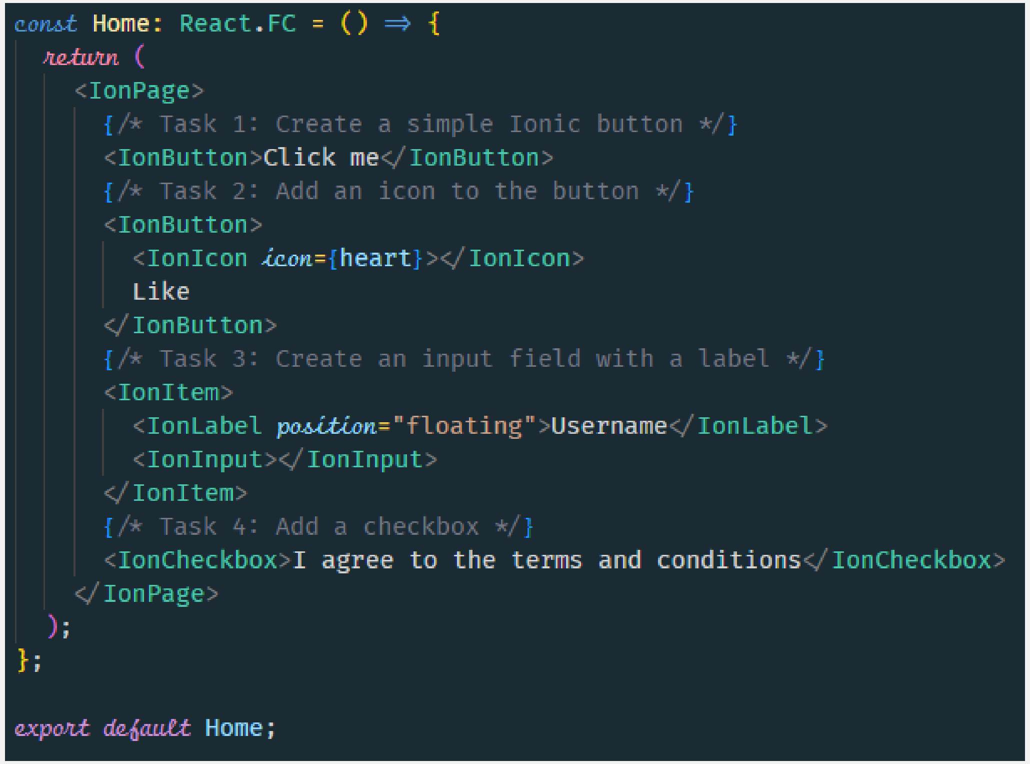
While we will be using Ionic to build the UI part of the application, we will be adding advanced functionality through React.

Full Documentation:

1. https://react.dev/

REACT COMPONENTS

We have been using React all the time to render our Ionic Components because each page is a React component!



```
const Home: React.FC = () => {
  return (
    <IonPage>
      {/* Task 1: Create a simple Ionic button */}
      <IonButton>Click me</IonButton>
      {/* Task 2: Add an icon to the button */}
      <IonButton>
        <IonIcon icon={heart}></IonIcon>
        Like
      </IonButton>
      {/* Task 3: Create an input field with a label */}
      <IonItem>
        <IonLabel position="floating">Username</IonLabel>
        <IonInput></IonInput>
      </IonItem>
      {/* Task 4: Add a checkbox */}
      <IonCheckbox>I agree to the terms and conditions</IonCheckbox>
    </IonPage>
  );
};

export default Home;
```

REACT COMPONENTS

- React is used to render a root component that is made up of other Ionic or HTML components. Ionic pages are essentially React components defined like this:

```
const Home: React.FC = () => {
  return (
    // Ionic Components defined here
  );
};
```

- A React component can also be defined like this:

```
const Home = () => {
  return (
    // Ionic Components defined here
  );
};
```

REACT STATES

- In React we use special variables called **State Variables**. Whenever we want to render something dynamic on the mobile device (such as a dynamic list, or a changing number, or other components that change), we need to store that value inside a **State**.

- We can define a React state like this:

```
const [count, setCount] = useState(0);
```

- You can only update the value of a state variable (Ex. count) by using the **State Setter Function** (Ex. setCount).

TYPESCRIPT

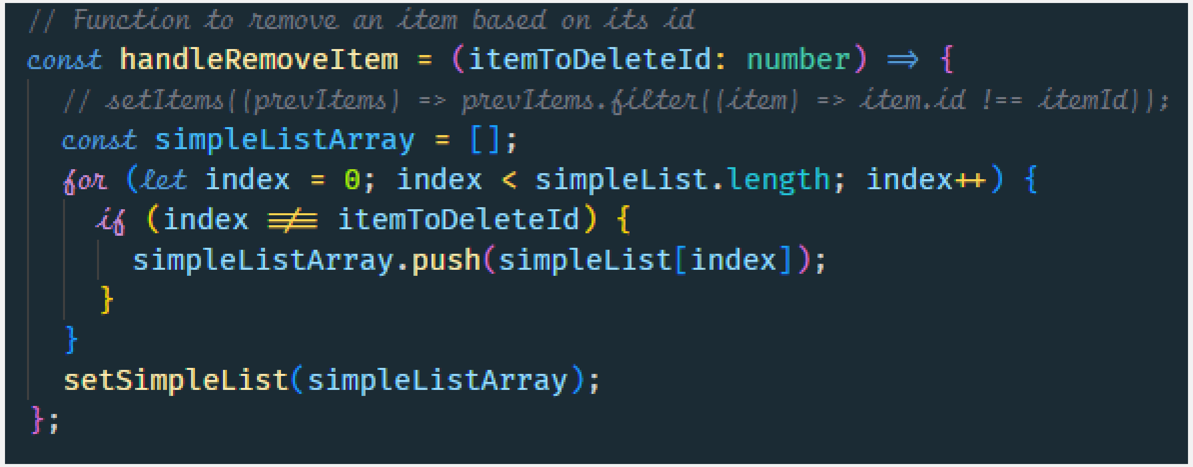
- Since by default, JavaScript does not check the variable types, this can lead to bugs (ex. Storing a string inside a variable you intended to store numbers).

- The **TypeScript** library helps to solve this problem. TypeScript is installed by default with Ionic-React.

TYPESCRIPT

- TypeScript is enforced by default in the following scenarios:
  1. Complex data types (ex. Arrays and objects)
  2. Function arguments

- Example:



```
// Function to remove an item based on its id
const handleRemoveItem = (itemToDeleteId: number) => {
  // setItems((prevItems) => prevItems.filter((item) => item.id !== itemId));
  const simpleListArray = [];
  for (let index = 0; index < simpleList.length; index++) {
    if (index !== itemToDeleteId) {
      simpleListArray.push(simpleList[index]);
    }
  }
  setSimpleList(simpleListArray);
};
```