EIU
TRƯỜNG ĐẠI HỌC
QUỐC TẾ
MIỀN ĐÔNG
EASTERN
INTERNATIONAL
UNIVERSITY

Mobile Programming

Chapter 5: WORK WITH DATA
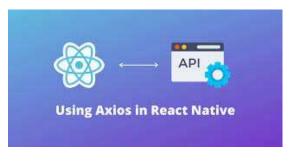
Mobile programming

1

Axios in React Native

➢ Axios is a popular JavaScript library for making HTTP requests.

➢ It works both in the browser and in Node.js environments.

➢ Simplifies the process of making asynchronous HTTP requests.

➢ Intercept request and response

➢ Transform request and response data

➢ Cancel requests

Using Axios in React Native

Mobile programming

2

2

Axios in React Native

➢ Install

$ npm install axios

$ yarn add axios

➢ Import

import axios from 'axios';

Website: https://github.com/axios/axios#axios-api

3

Axios in React Native

Making GET Request

```
axios.get('https://api.example.com/data')
  .then(response => {
    // Handle success
    console.log('Data:', response.data);
  })
  .catch(error => {
    // Handle error
    console.error('Error:', error);
  });
```

4

Axios in React Native

Making POST Request

```
const postData = {
  key1: 'value1',
  key2: 'value2',
};
axios.post('https://api.example.com/create', postData)
  .then(response => {
    // Handle success
    console.log('Response:', response.data);
  })
  .catch(error => {
    // Handle error
    console.error('Error:', error);
  });
```

Mobile programming

5

5

Axios in React Native

Making PUT Request

```
const updatedData = {
  key1: 'new_value1',
  key2: 'new_value2',
};
axios.put('https://api.example.com/update/123',
updatedData)
  .then(response => {
    // Handle success
    console.log('Response:', response.data);
  })
  .catch(error => {
    // Handle error
    console.error('Error:', error);
  });
```

Mobile programming

6

6

Axios in React Native

Making DELETE Request

```
axios.delete('https://api.example.com/delete/123')
  .then(response => {
    // Handle success
    console.log('Response:', response.data);
  })
  .catch(error => {
    // Handle error
    console.error('Error:', error);
  });
```

Mobile programming   7

7

Axios in React Native

Authentication with Axios

```
const authToken = 'your_auth_token';

axios.get('https://api.example.com/data', {
 headers: {
   Authorization: `Bearer ${authToken}`,
 },
})
  .then(response => {
    // Handle success
    console.log('Data:', response.data);
  })
  .catch(error => {
    // Handle error
    console.error('Error:', error);
  });
```

Mobile programming   8

8

Example

https://snack.expo.dev/@taitv/demo-axios

9

React Native FS

React Native FS is a package that allows you to easily work with the native file system in React Native

For RN < 0.57 and/or Gradle < 3 you MUST install react-native-fs at version @2.11.17!
For RN >= 0.57 and/or Gradle >= 3 you MUST install react-native-fs at version >= @2.13.2!
For RN >= 0.61 please install react-native-fs at version >= @2.16.0!

10

React Native FS

Install

npm install react-native-fs --save

Note: If your react-native version is < 0.40 install with this tag instead:

npm install react-native-fs@2.0.1-rc.2 --save

Import

```
import RNFS from 'react-native-fs';
```

https://www.npmjs.com/package/react-native-fs?activeTab=versions

Mobile programming

11

11

React Native FS

Adjust the information on the Android application

android/settings.gradle
...
include ':react-native-fs'
project(':react-native-fs').projectDir = new File(settingsDir, '../node_modules/react-native-fs/android')

android/app/build.gradle
...
dependencies {
   ...
   implementation project(':react-native-fs')
}

Mobile programming

12

12

React Native FS

Adjust the information on the Android application

Register module (in MainActivity.java)
import com.rnfs.RNFSPackage; // <------- add package

public class MainApplication extends Application implements ReactApplication {
  // ...
  @Override
  protected List<ReactPackage> getPackages() {
   return Arrays.<ReactPackage>asList(
    new MainReactPackage(), // <---- add comma
    new RNFSPackage() // <---------- add package
  );
  }

Mobile programming

13

13

React Native FS

Read

```javascript
import React, { useState, useEffect } from 'react';
import { View, Text, Button } from 'react-native';
import RNFS from 'react-native-fs';

const ReadFileExample = () => {
 const [readText, setReadText] = useState('');

 useEffect(() => {
    readFile();
 }, []);

 const readFile = async () => {
  try {
    const filePath = `${RNFS.DocumentDirectoryPath}/example.txt`;
    const content = await RNFS.readFile(filePath, 'utf8');
    setReadText(content);
  } catch (error) {
    console.error('Error reading file:', error);
  }
 };
```

Mobile programming

14

14

React Native FS

Write

```
const WriteFileExample = () => {
  const [textToWrite, setTextToWrite] = useState('');

  const writeFile = async () => {
    try {
      const filePath =
`${RNFS.DocumentDirectoryPath}/example.txt`;
      await RNFS.writeFile(filePath, textToWrite,
'utf8');
      console.log('File written successfully!');
    } catch (error) {
      console.error('Error writing file:', error);
    }
  };
```

Mobile programming

15

15

React Native FS

Delete

```
const DeleteFileExample = () => {
  const deleteFile = async () => {
    try {
      const filePath =
`${RNFS.DocumentDirectoryPath}/example.txt`;
      await RNFS.unlink(filePath);
      console.log('File deleted successfully!');
    } catch (error) {
      console.error('Error deleting file:', error);
    }
  };
```

Mobile programming

16

16

React Native FS

Demo

https://github.com/taitv86/DemoReactNativeFS

17

Exercise

1. In Lab 3, Lab 4: Replacing functions using Fetch API using the Axios library.
2. Write a function to create a file with a 100-element integer sequence of random numbers. Then read the series of numbers and output them to the screen.

18

Q&A

19

19