

Parallel Promises



Mosh Hamedani
programmingwithmosh.com

Activities

Visual Studio Code

File Explorer

Terminal

Run and Debug

Source Control

Python

PC

VS Code

Firefox

Chrome

Ubuntu

File Edit Selection View Go Run Terminal Help

EXPLORER

OPEN EDITORS

APPLICATIONS

10. Running Parallel Promises

async-demo

package.json

promise-api.js

README.md

11. Async And Await

12. Exercise

README.md

OUTLINE

TIMELINE

promise-api.js

10. Running Parallel Promises > async-demo > JS promise-api.js > ...

```
1  const p1 = new Promise((resolve) => {
2      setTimeout(() => {
3          console.log('Async operation 1...');
4          resolve(1);
5      }, 2000);
6  });
7
8  const p2 = new Promise((resolve) => {
9      setTimeout(() => {
10         console.log('Async operation 2...');
11         resolve(2);
12     }, 2000);
13 });
14
15 Promise.all([p1, p2])
16     .then(result => console.log(result));
```

DEBUG CONSOLE

PROBLEMS

OUTPUT

TERMINAL

4: bash

```
dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/06. Asynchronous JavaScript/Applications/10. Running Parallel Promises/async-dem
o$ node promise-api.js
Async operation 1...
Async operation 2...
[ 1, 2 ]
dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/06. Asynchronous JavaScript/Applications/10. Running Parallel Promises/async-dem
o$
```

master*

0 0 0

javascript | promise-api.js

Ln 16, Col 42

Spaces: 4

UTF-8

LF

Babel JavaScript

JavaScript Standard Style

ActivitiesVisual Studio Code

Fri 13:56 •promise-api.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

JS promise-api.js X

10. Running Parallel Promises > async-demo > JS promise-api.js > ...

18const p1 = new Promise((resolve, reject) => {

19 setTimeout(() => {

20 console.log('Async operation 1...');

21 reject(new Error('because something failed.'));

22 }, 2000);

23});

24

25const p2 = new Promise((resolve) => {

26 setTimeout(() => {

27 console.log('Async operation 2...');

28 resolve(2);

29 }, 2000);

30});

31

32Promise.all([p1, p2])

33 .then(result => console.log(result))

34 .catch(err => console.log('Error', err.message));

DEBUG CONSOLEPROBLEMSOUTPUTTERMINAL

4: bash

dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/06. Asynchronous JavaScript/Applications/10. Running Parallel Promises/async-demo\$ node promise-api.js

Async operation 1...

Async operation 2...

Error because something failed.

dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/06. Asynchronous JavaScript/Applications/10. Running Parallel Promises/async-demo\$

master*0 0 0javascript | promise-api.js

Ln 33, Col 41Spaces: 4UTF-8LFBabel JavaScriptJavaScript Standard Style

ActivitiesVisual Studio Code

Fri 14:01284B/s1.66KB/s

promise-api.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

JS promise-api.js

10. Running Parallel Promises > async-demo > JS promise-api.js > ...

```
36 const p1 = new Promise((resolve) => {
37     setTimeout(() => {
38         console.log('Async operation 1...');
39         resolve(1);
40     }, 2000);
41 });
42
43 const p2 = new Promise((resolve) => {
44     setTimeout(() => {
45         console.log('Async operation 2...');
46         resolve(2);
47     }, 2000);
48 });
49
50 Promise.race([p1, p2])
51     .then(result => console.log(result))
52     .catch(err => console.log('Error', err.message));
53
54 // The Promise.race() method returns a promise that fulfills
55 // or rejects as soon as one of the promises in an iterable fulfills
56 // or rejects, with the value or reason from that promise.
```

DEBUG CONSOLEPROBLEMSOUTPUTTERMINAL

4: bash

```
dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/06. Asynchronous JavaScript/Applications/10. Running Parallel Promises/async-demo$ node promise-api.js
Async operation 1...
Async operation 2...
1
dandy@My-Ubuntu:~/Keep/GitHub Repositories/Node.js/06. Asynchronous JavaScript/Applications/10. Running Parallel Promises/async-demo$
```

master*0 0 0javascript | promise-api.jsLn 56, Col 15Spaces: 4UTF-8LFBabel JavaScriptJavaScript Standard Style