

Returning values from Async Functions

Before learning:

- ▲ CASE 1: We have to learn why main program execute entirely and do not execute the ASYNC funcion in between the main code program ?

```
JS index.js U ●

Promise-5-Async-Returning values > JS index.js > ...
 ↳
27 //-----
28 //Async function, which returns a promise
29 /-----
30 //but has some synchronous calling inside it
31 //so the 3 calling inside it, are executed one after the end of the other.
32 const getDocPic = async() => {
33   try {
34     //await: call synchronously this function as a promise
35     const data = await readFilePro(`$__dirname/dog.txt`); //this is a promise, called synchronously
36     console.log(`Breed, ${data}`);
37
38     //await: call synchronously this function as a promise
39     const res = await superagent.get(`https://dog.ceo/api/breed/${data}/images/random`); //returns a Promise
40     console.log(res.body.message);
41
42     //await: call synchronously this function as a promise
43     //returns a promise, waiting for it to write
44     //no need here to create a variable in this case
45     await writeFilePro('dog-img.txt', res.body.message);
46     console.log('Random dog image saved to file!');
47   } catch (error) {
48     console.log(error);

```



```
49 | }
50 };
51
52 //call this async function (have some synchronous callings inside)
53 ①console.log('1: Will get doc pics!');
54 ②getDocPic();Async calling
55 ③console.log('2: Done!');
56
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL CONSOLE GITLENS COMMENTS

PS C:\Users\glauc\git-root_Exemplos_Node.js\Promise-5-Async-Returning values> nodemon index.js

```
[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node index.js`
```

①1: Will get doc pics!

③2: Done!

②Breed, labrador Async calling

https://images.dog.ceo/breeds/labrador/n02099712_7164.jpg

Random dog image saved to file!

[nodemon] clean exit - waiting for changes before restart

Why Console.log Done execute before the ASYNC function ?
It is because the thread of main program have to execute until the end,
independent of other ASYNC functions.

So, the ASYNC function was executed after the main code of program.

- ▲ CASE 2: Trying to write a result of a promise, but Javascript does not know yet the result, in this point of execution of code.



JS index.js U X

```
Promise-5-Async-Returning values > JS index.js > ...
20
27 //-----
28 //Async function, which returns a promise
29 //-----
30 //but has some synchronous calling inside it
31 //so the 3 calling inside it, are executed one after the end of the other.
32 const getDocPic = async () => {
33   try {
34     //await: call synchronously this function as a promise
35     const data = await readFilePro(`$__dirname__/dog.txt`); //this is a promise, called synchronously
36     console.log(`Breed, ${data}`);
37
38     //await: call synchronously this function as a promise
39     const res = await superagent.get(`https://dog.ceo/api/breed/${data}/images/random`); //returns a Promise
40     console.log(res.body.message);
41
42     //await: call synchronously this function as a promise
43     //returns a promise, waiting for it to write
44     //no need here to create a variable in this case
45     await writeFilePro('dog-img.txt', res.body.message);
46     console.log('Random dog image saved to file!');
47   } catch (error) {
48     console.log(error);
49   }
50   return '2: ready 🐶';
51 };
52
53 //call this async function (have some synchronous callings inside)
54 console.log('1: Will get doc pics!');
55 const x = getDocPic();
```



```
56  console.log(x);
57  console.log('2: Done!');

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL CONSOLE GITLENS COMMENTS > nod

[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node index.js`
1: Will get doc pics!
Promise { <pending> } ← Still, javascript does not know what to print here, because promise is pending on this point of execution yet.
2: Done!
Breed, labrador
https://images.dog.ceo/breeds/labrador/n02099712_9374.jpg
Random dog image saved to file!
[nodemon] clean exit - waiting for changes before restart
```

▲ SOLUTION 1 - (SIMPLE) for executing commands in the sequence that I really need:

BUT, I really want to get this promise returned value.



JS index.js U X

Promise-5-Async-Returning values > JS index.js > ...

```
32 const getDocPic = async () => {
33   try {
34     //await: call synchronously this function as a promise
35     const data = await readFilePro(`__dirname}/dog.txt`); //this is a promise, called synchronously
36     console.log(` Breed, ${data}`);
37
38     //await: call synchronously this function as a promise
39     const res = await superagent.get(`https://dog.ceo/api/breed/${data}/images/random`); //returns a Promise
40     console.log(` ` + res.body.message);
41
42     //await: call synchronously this function as a promise
43     //returns a promise, waiting for it to write
44     //no need here to create a variable in this case
45     await writeFilePro('dog-img.txt', res.body.message);
46     console.log(` Random dog image saved to file!`);
47   } catch (error) {
48     console.log(error);
49   }
50   return '2: ready 🐶';
51 }
```

//call this async function (have some synchronous callings inside)

54 console.log('1: Will get doc pics!');

```
55   getDocPic().then((x) => {
56     console.log(x);
57     console.log('3: Done!');
58   });
59 // console.log('2: Done!');
```

SOLUTION

for returning the result of promise,
in the correct sequence that I need

A Call this function, using a .then()

It will wait for success of this function called in promise

B Move this code inside this new function,

So it will execute ONLY after the sucess of the promise above.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL CONSOLE GITLENS COMMENTS

node - P

PS C:\Users\alecdu\git-root\Exemplos_Node.js\Promise-5-Async-Returning values> nodemon index.js



Generated by Notion to PDF

```
[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node index.js`  
1: Will get doc pics! ← 1 first console.log command
Breed, labrador
https://images.dog.ceo/breeds/labrador/n02099712_5844.jpg
Random dog image saved to file!
2: ready 🐶 ← 2 secondly, the promise is waited for its sucess
3: Done! ← 3 third, after sucess of promise we call console.log('3:Done')  
[nodemon] clean exit - waiting for changes before restart
```

EXECUTION explained:

▲ SOLUTION 2 (BETTER)

Adding catch manipulation for returning promise errors



JS index.js u X

BUT we need to manipulate ERRORS

Promise-5-Async-Returning values > JS index.js > ...

```
27 //-----  
28 //Async function, which returns a promise  
29 //-----  
30 //but has some synchronous calling inside it  
31 //so the 3 calling inside it, are executed one after the end of the other.  
32 const getDocPic = async () => {  
33   try {  
34     //await: call synchronously this function as a promise  
35     const data = await readFilePro(`$__dirname`/doGGg.txt`); //this is a promise, called synchronously  
36     console.log(` Breed, ${data}`);  
37  
38     //await: call synchronously this function as a promise  
39     const res = await superagent.get(`https://dog.ceo/api/breed/${data}/images/random`); //returns a Promise  
40     console.log(` ` + res.body.message);  
41  
42     //await: call synchronously this function as a promise  
43     //returns a promise, waiting for it to write  
44     //no need here to create a variable in this case  
45     await writeFilePro('dog-img.txt', res.body.message);  
46     console.log(` Random dog image saved to file!`);  
47   } catch (error) {  
48     console.log(error);  
49     throw error;  
50   }  
51 }
```



```
49     |   throw error;
50   | }
51   | return '2: ready 🎉';
52 };
53
54 //call this async function (have some synchronous callings inside)
55 console.log('1: Will get doc pics!');
56 getDocPic()
57   .then((x) => {
58     console.log(x);
59     console.log('3: Done!');
60   })
61   .catch((err) => {
62     console.log(`ERROR ⚡`);
63   });
64 // console.log('2: Done!');
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL CONSOLE GITLENS COMMENTS

```
PS C:\Users\glauc\git-root\_Exemplos_Node.js\Promise-5-Async-Returning values> nodemon index.js
[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node index.js`
1: Will get doc pics!
I could not find a file 🤔
ERROR ⚡
[nodemon] clean exit - waiting for changes before restart
```

▲ SOLUTION 3 - also is BETTER



JS index.js u X

Promise-5-Async-Returning values > JS index.js > ...

```
27 //-----
28 //Async function, which returns a promise
29 //-----
30 //but has some synchronous calling inside it
31 //so the 3 calling inside it, are executed one after the end of the other.
32 const getDogPic = async () => {
33   try {
34     //await: call synchronously this function as a promise
35     const data = await readFilePro(`$__dirname__/dog.txt`); //this is a promise, called synchronously
36     console.log(` Breed, ${data}`);
37
38     //await: call synchronously this function as a promise
39     const res = await superagent.get(`https://dog.ceo/api/breed/${data}/images/random`); //returns a Promise
40     console.log(` ` + res.body.message);
41
42     //await: call synchronously this function as a promise
43     //returns a promise, waiting for it to write
44     //no need here to create a variable in this case
45     await writeFilePro('dog-img.txt', res.body.message);
46     console.log(` Random dog image saved to file!`);
47   } catch (error) {
48     console.log(error);
49     throw error;
50   }
51   return '2: ready 🐶';
52 };
53
54(async () => {
55   try {
56     console.log('1: Will get doc pics!');
57     const x = await getDogPic();
58     console.log(x);
59     console.log('3: Done!');
60   } catch (error) {
61   }
62 })()
```

Create your own async function

And call all functions you need, inside it.



```
61     |     console.log(`ERROR ⚡`);  
62     }  
63 })();  
64
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL CONSOLE GITLENS COMMENTS

```
PS C:\Users\glau\git-root\_Exemplos_Node.js\Promise-5-Async-Returning values> nodemon index.js  
[nodemon] 3.0.2  
[nodemon] to restart at any time, enter `rs`  
[nodemon] watching path(s): *.*  
[nodemon] watching extensions: js,mjs,cjs,json  
[nodemon] starting `node index.js`  
1: Will get doc pics!  
Breed, labrador  
https://images.dog.ceo/breeds/labrador/n02099712\_3698.jpg  
Random dog image saved to file!  
2: ready 🐶  
3: Done!  
[nodemon] clean exit - waiting for changes before restart
```



Generated by Notion to PDF