Async/Await in JavaScript

What is a async/await ?

Async/await is a syntactic feature of programming languages that allows an Asynchronous function to be structured in a way similar to an ordinary synchronous function.

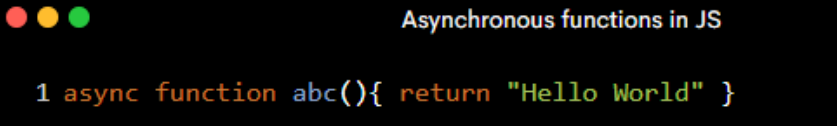
Async/Await is usually preferred over promise chaining as it's very much easy to write and analyze the code written

Creating async function

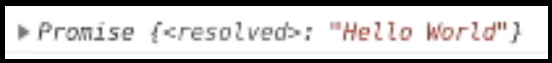
Asynchronous functions in JS

```
1 async function abc(){ YOUR CODE HERE }
```

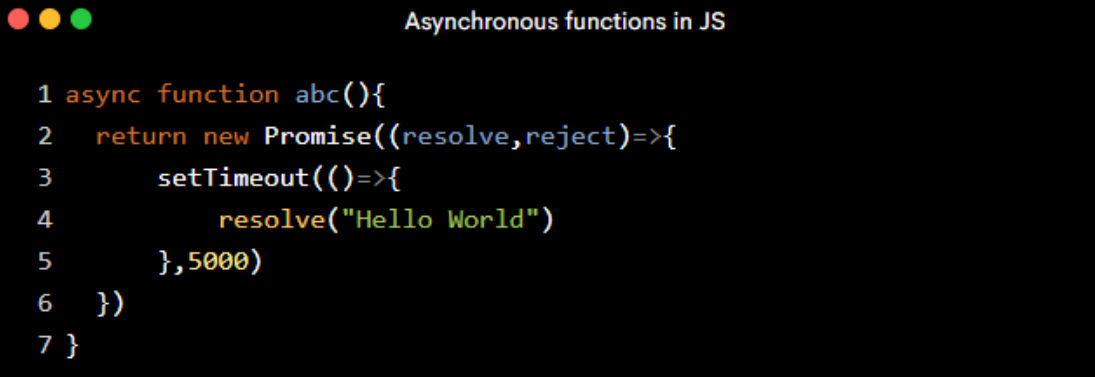
Async functions always return a promise



Output



Async function returning promise



This is how promise is returned from async functions

Use of Await Keyword

Await keyword is used to wait for a promise, it waits until the promise is settled. It doesn't matters whether its resolved or rejected. The main thing is that the promise is not pending anymore.

Example of async await

Let's create promise that gets resolved after 2 seconds

Asynchronous functions in JS

```
async function getSpecificNumber(){
  return new Promise((resolve,reject)=>{
      setTimeout(()=>{
          resolve(42)
      },2000)
  })
}
```

Now we'll call this function, swipe to see

When function abc is called it literally wait for 2 seconds for promise returned by getSpecificNumber function to get resolved or rejected.

```
9 //Calling above function from another function
10 async function abc(){
11     const specificNumber = await getSpecificNumber();
12     console.log(specificNumber);
13 }
14 abc()
15 //Output will be 42 displayed after 2 seconds
```

However, important point is whole application doesn't get paused while this abc function is waiting for promise to get resolve. JS engine can execute other functions and events as usual in the mean time.

Follow Rohit Sharma for more such updates