

# JavaScript Async/Await

@codingtute

## Async

When we append the keyword “async” to the function, this function returns the Promise by default on execution. Async keyword provides extra information to the user of the function:

- The function contains some Asynchronous Execution
- The returned value will be the Resolved Value for the Promise.

```
async function f() {  
    return 1;  
}  
f().then(alert); // 1
```

---

## Await

The keyword "await" makes JavaScript wait until that promise settles and returns its result

- The 'await' works only inside async functions

```
async function f() {  
    let promise = new Promise((resolve, reject) => {  
        setTimeout(() => resolve("done!"), 1000)  
    });  
    let result = await promise; // wait until the promise resolves  
    alert(result); // "done!"  
}  
f();
```

# JavaScript Async/Await

@codingtute

## Async

When we append the keyword “async” to the function, this function returns the Promise by default on execution. Async keyword provides extra information to the user of the function:

- The function contains some Asynchronous Execution
- The returned value will be the Resolved Value for the Promise.

```
async function f() {  
    return 1;  
}  
f().then(alert); // 1
```

## Await

The keyword "await" makes JavaScript wait until that promise settles and returns its result

- The 'await' works only inside async functions

```
async function f() {  
    let promise = new Promise((resolve, reject) => {  
        setTimeout(() => resolve("done!"), 1000)  
    });  
    let result = await promise; // wait until the promise resolves  
    alert(result); // "done!"  
}  
f();
```