

## 17 Promise

### What is a Promise?

Promise is an object that provides a useful construct when dealing with asynchronous tasks. A promise is called a "Promise" because it guarantees it will run upon success or failure of that task.

Working with a promise consists of two parts; (A) Creating a promise, and (B) Using a promise.

```
// (A) Create a promise
const p = new Promise((resolve, reject)=>{
  // Do some async task
  setTimeout(()=>{
    if(condition){
      resolve('Successful login');
    } else {
      reject('Login failed');
    }
  }, 2000)
})
```

### What is an Async task?

An async task is one in which a third-party process is doing the task.

Examples:

- Requesting/sending data to a database
- Requesting/sending data via HTTP protocol
- Working with the file system of the computer

```
// (B) Using a promise
p.then((res)=>{
  console.log(res)
})
.catch((err)=>{
  console.log(err)
})
```

**Note:** 90% of the time you will be working with pre-existing promises. The step of "Creating a promise" would be done for you either by a library, framework or environment you are using. Examples of promises: fetch