

Promises

As *Promises* no JavaScript estão presentes em praticamente todo o ecossistema da linguagem. Elas são um padrão de desenvolvimento que visam representar a conclusão de operações assíncronas.

Uma Promise possui diferentes estados, sendo alguns deles:

- ♦ Pendente (*Pending*);
- ♦ Resolvida (*Resolved*);
- ♦ Rejeitada (*Rejected*);
- ♦ Realizada (*Fulfilled*);
- ♦ Estabelecida (*Settled*).

Geralmente os estados mais utilizados são: Resolvida e Rejeitada.

Ao criar uma *Promise*, ela se inicia como pendente (*pending*), assim, os estados em que ela pode se apresentar são aqueles informados anteriormente. Se ela estiver no estado de resolvida (*resolved*) é porque tudo deu certo, ou seja, a *Promise* foi criada e processada com sucesso, porém, em casos de falhas, ela estará no estado de rejeitada (*rejected*).

Uma das maneiras de fazer esse tratamento é por meio das funções *then* e *catch*, para sucesso ou falha, respectivamente.

Para resolver uma *Promise*, podemos utilizar a função `resolve`, passando como parâmetro um valor que será acessível através de nossa *Promise* resolvida:

```
function cadastrarUsuario() {  
  return new Promise((resolve, reject) => {  
    setTimeout(() => {  
      resolve("Usuário cadastrado");  
    }, 5000);  
  });  
}
```

Para pegar o valor passado para a função `resolve`, utilizamos o operador `then()`. Exemplo:

```
cadastrarUsuario().then((mensagem) => {  
  console.log(mensagem);  
  //será escrito na tela a mensagem Usuário Cadastrado  
});
```

Para rejeitar uma *Promise*, usamos a função `reject` (da mesma forma que a `resolve`). Exemplo:

```
function cadastrarUsuario() {  
  return new Promise((resolve, reject) => {  
    setTimeout(() => {  
      reject("Erro ao cadastrar usuário");  
    }, 5000);  
  });  
}
```

Assim como temos o `then()` para sucesso, também temos o `catch()` para erros:

```
cadastroUsuario()  
  .then((mensagem) => {  
    console.log(mensagem);  
  })  
  .catch((mensagem) => {  
    console.log(mensagem); //será escrito na tela a mensagem "Erro ao cadastr  
  });
```

Tanto o `resolve` com o `then` ou `reject` com `catch` funcionam em conjunto da mesma maneira, porém cada um tendo seu propósito: o primeiro para sucesso e o segundo para erro.

Referência

CASTIGLIONI, M. **Trabalhando Com Promises Em Javascript** .
Disponível em: <<https://blog.matheuscastiglioni.com.br/trabalhando-com-promises-em-javascript/>>. Acesso em: 10 mar. 2023.