

Sessão 11 (05/02/2026) – Módulo 02 – Fundamentos JavaScript – Aula 07 – scope & closures

Listar de Exercícios

```
const ligarLuz = (cor, tempo) => {
```

```
  return new Promise((resolve) => {
```

```
    setTimeout(() => {
```

```
      console.log(`Luz ${cor} ligada!`);
```

```
      resolve();
```

```
    }, tempo);
```

```
  });
```

```
};
```

// EXERCÍCIO: Encadeia as luzes para que:

// 1. Ligue o Verde (2s) -> 2. Ligue o Amarelo (1s) -> 3. Ligue o Vermelho (2s)

```
ligarLuz("VERDE 🔥", 2000)
```

// Continua o encadeamento aqui...

```
const apiNumeros = () => Promise.resolve([10, 20, 30]);
```

// EXERCÍCIO:

// 1. No primeiro .then, multiplica todos os números por 2.

// 2. No segundo .then, calcula a média e retorna-a.

// 3. No último .then, imprime: "A média final é X".

```
apiNumeros()
```

```
  .then((lista) => {
```

// Teu código aqui

```
})
```

// CÓDIGO ANTIGO (Não mexer aqui, apenas transformar)

```
function passo1(cb) { setTimeout(() => { console.log("Passo 1"); cb(); }, 1000); }
```

```
function passo2(cb) { setTimeout(() => { console.log("Passo 2"); cb(); }, 1000); }
```

// EXERCÍCIO: Transforma as funções acima em Promises e usa .then()

```
const passo1Promessa = () => { /* cria a promise aqui */};
```

```
const passo2Promessa = () => { /* cria a promise aqui */};
```

```

passo1Promessa().then(() => passo2Promessa());

const verificarIdade = (idade) => {
  return new Promise((resolve, reject) => {
    // Se idade < 18, reject "Acesso Negado"
    // Se idade >= 18, resolve "Acesso Autorizado"
  });
};

// EXERCÍCIO: Testa a função com idade 15 e captura o erro no .catch

const buscarPreco = () => Promise.resolve(100);

// EXERCÍCIO:
// 1. Busca o preço.
// 2. No .then, diminui 20€ do valor.
// 3. No próximo .then, imprime "Preço com desconto: X€".

// EXERCÍCIO: Cria uma promise que após 3 segundos
// imprime "Pode clicar no botão agora!".

const buscarUsuario = (email) => {
  return new Promise((resolve, reject) => {
    if (email === "admin@email.com") resolve({ id: 1, nome: "Admin" });
    else reject("Utilizador não encontrado!");
  });
};

// EXERCÍCIO:
// 1. Chama buscarUsuario com um email qualquer.
// 2. Se encontrar, faz console.log("Bem-vindo " + user.nome).
// 3. Se falhar, faz console.error do erro.

```

```
const obterTaxaCâmbio = () => Promise.resolve(1.10); // 1€ = 1.10$
```

// EXERCÍCIO:

// 1. Pega na taxa de câmbio.

// 2. No .then, multiplica essa taxa por um valor de 50€ e retorna o resultado.

// 3. No próximo .then, imprime: "O valor convertido é \$X".

```
const parte1 = () => Promise.resolve("As Promises");
```

```
const parte2 = (texto) => Promise.resolve(` ${texto} são`);
```

```
const parte3 = (texto) => Promise.resolve(` ${texto} poderosas!`);
```

// EXERCÍCIO: Encadeia as 3 funções para imprimir a frase completa.

```
const calcularArea = (lado) => {
  return new Promise((resolve, reject) => {
    if (lado > 0) resolve(lado * lado);
    else reject("Lado inválido!");
  });
};
```

// EXERCÍCIO: Tenta calcular a área com lado = -5 e usa o .catch para mostrar o erro.

```
const carregarFoto = (url) => {
  return new Promise((resolve, reject) => {
    if (url.includes(".jpg")) resolve("Foto Carregada");
    else reject("URL Inválido!");
  });
};
```

// EXERCÍCIO: Cria uma função async que tenta carregar "foto.png".

// No catch, define a variável resultado como "default.jpg" e imprime no fim.

```
const getPrecoVenda = () => Promise.resolve(100);
const getImposto = () => Promise.resolve(23);

// EXERCÍCIO: No .then de getPrecoVenda, chama o getImposto.
// No final, imprime a soma dos dois valores.

const saldo = 50;
const levantarDinheiro = (valor) => {
    return new Promise((res, rej) => {
        if (valor <= saldo) res("Dinheiro entregue");
        else rej("Saldo insuficiente");
    });
};

// EXERCÍCIO: Tenta levantar 100€ e trata o erro com uma mensagem amigável.

const getNomeApi = () => Promise.resolve("joão ronaldo");

// EXERCÍCIO:
// 1. Recebe o nome.
// 2. No .then, transforma em MAIÚSCULAS.
// 3. No próximo .then, imprime "Utilizador: JOÃO RONALDO".
```