

Exercício - Promise

Questão 1. Implemente uma função ***delayedMessage*** que retorna uma Promise. A função deve receber dois argumentos: uma string message e um número delay (em milissegundos). A Promise deve resolver com a message após o tempo especificado por delay.

Questão 2. Dada a função ***getUserData***, que retorna uma Promise que pode ser resolvida com os dados do usuário ou rejeitada com um erro, implemente uma função ***displayUserName*** que extrai o nome do usuário e o imprime no console. Se ocorrer um erro, a função deve imprimir "Erro ao obter os dados do usuário" seguido da mensagem de erro.

```
function getUserData() {  
  return new Promise((resolve, reject) => {  
    setTimeout(() => {  
      // Simulando um caso de erro 50% das vezes  
      if (Math.random() > 0.5) {  
        resolve({ name: "Alice" });  
      } else {  
        reject(new Error("Falha na conexão"));  
      }  
    }, 1000);  
  });  
}
```

Questão 3. Refaça a função ***displayUserName*** utilizando ***async/await*** para trabalhar com a Promise retornada por ***getUserData***. A função deve extrair e imprimir o nome do usuário se a Promise for resolvida. Se ocorrer um erro, a função deve imprimir "Erro ao obter os dados do usuário" seguido da mensagem de erro.

Dica: use o bloco try-catch para tratar exceção.