

# Produzindo e consumindo mensagens da fila SQS

Criamos um exemplo de como você pode criar duas classes Java simples: uma para produzir (enviar) uma mensagem para uma fila do Amazon SQS e outra para consumir (receber) mensagens dessa fila.

## Pré-requisitos:

**AWS SDK para Java:** Certifique-se de ter o AWS SDK para Java 2.x adicionado ao seu projeto. Se você estiver usando Maven, adicione a seguinte dependência ao seu `pom.xml`:

Copiar código

```
<dependency>
    <groupId>software.amazon.awssdk</groupId>
    <artifactId>sqs</artifactId>
    <version>2.20.0</version> <!-- Verifique a versão mais recente -->
</dependency>
```

- 1.
2. **Credenciais da AWS:** Configure suas credenciais da AWS no ambiente onde o código será executado. Você pode fazer isso através do arquivo `~/.aws/credentials` ou usando variáveis de ambiente.

---

## Classe para Produzir Mensagens (SqsProducer.java)

java

Copiar código

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.sqs.SqsClient;
import software.amazon.awssdk.services.sqs.model.SendMessageRequest;

public class SqsProducer {

    private static final String QUEUE_URL =
"https://sqs.<região>.amazonaws.com/<seu-id-da-conta>/<nome-da-fila>";

    public static void main(String[] args) {

        // Cria o cliente SQS
        SqsClient sqsClient = SqsClient.builder()
```

```
        .region(Region.<SUA_REGIAO>)
        .build();

    // Envia uma mensagem
    sendMessage(sqsClient, "Olá, esta é uma mensagem de
teste!");

    // Fecha o cliente
    sqsClient.close();
}

public static void sendMessage(SqsClient sqsClient, String
message) {
    SendMessageRequest sendMsgRequest =
SendMessageRequest.builder()
        .queueUrl(QUEUE_URL)
        .messageBody(message)
        .delaySeconds(0)
        .build();

    sqsClient.sendMessage(sendMsgRequest);
    System.out.println("Mensagem enviada: " + message);
}
}
```

---

## Classe para Consumir Mensagens (SqsConsumer.java)

java

Copiar código

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.sqs.SqsClient;
import
software.amazon.awssdk.services.sqs.model.DeleteMessageRequest;
import software.amazon.awssdk.services.sqs.model.Message;
import
software.amazon.awssdk.services.sqs.model.ReceiveMessageRequest;

import java.util.List;

public class SqsConsumer {
```

```

    private static final String QUEUE_URL =
"https://sqs.<região>.amazonaws.com/<seu-id-da-conta>/<nome-da-fila>";

    public static void main(String[] args) {

        // Cria o cliente SQS
        SqsClient sqsClient = SqsClient.builder()
            .region(Region.<SUA_REGIAO>)
            .build();

        // Recebe e processa mensagens
        receiveMessages(sqsClient);

        // Fecha o cliente
        sqsClient.close();
    }

    public static void receiveMessages(SqsClient sqsClient) {
        ReceiveMessageRequest receiveRequest =
ReceiveMessageRequest.builder()
            .queueUrl(QUEUE_URL)
            .maxNumberOfMessages(5)
            .waitTimeSeconds(10) // Long polling
            .build();

        List<Message> messages =
sqsClient.receiveMessage(receiveRequest).messages();

        for (Message message : messages) {
            System.out.println("Mensagem recebida: " +
message.body());

            // Exclui a mensagem da fila após o processamento
            DeleteMessageRequest deleteRequest =
DeleteMessageRequest.builder()
                .queueUrl(QUEUE_URL)
                .receiptHandle(message.receiptHandle())
                .build();
            sqsClient.deleteMessage(deleteRequest);
        }
    }

```

```
}  
}
```

---

## Notas Importantes:

- **Substitua os espaços reservados:**
  - `<região>`: A região AWS onde sua fila SQS está localizada (por exemplo, `us-east-1`).
  - `<seu-id-da-conta>`: Seu ID de conta AWS.
  - `<nome-da-fila>`: O nome da sua fila SQS.
  - `<SUA_REGIAO>`: A região AWS (por exemplo, `Region.US_EAST_1`).
- **Permissões:** Certifique-se de que as credenciais usadas têm permissão para acessar e manipular a fila SQS especificada.
- **Configuração das Credenciais:** As credenciais da AWS podem ser configuradas de várias maneiras, incluindo arquivos de configuração, variáveis de ambiente ou perfis do IAM se estiver executando em uma instância EC2.

---

## Explicação:

- **SqsProducer:**
  - Cria um cliente SQS usando a região especificada.
  - Envia uma mensagem para a fila usando `sendMessage`.
- **SqsConsumer:**
  - Cria um cliente SQS usando a região especificada.
  - Recebe mensagens da fila usando `receiveMessages`.
  - Processa cada mensagem e, em seguida, a exclui da fila para evitar processamento duplicado.

---

## Execução:

1. **Produzir Mensagens:**
  - Compile e execute `SqsProducer`.
  - Verifique o console para confirmar que a mensagem foi enviada.
2. **Consumir Mensagens:**
  - Compile e execute `SqsConsumer`.
  - Verifique o console para ver as mensagens recebidas e confirmadas.

---

Espero que este exemplo ajude você a começar a trabalhar com o Amazon SQS em Java! Se tiver alguma dúvida ou precisar de mais assistência, não hesite em perguntar.