# Guida Completa per Inviare Messaggi Multipli a SQS con Spring Boot e LocalStack

## 1. Introduzione

Questa guida spiega come configurare un progetto Spring Boot per inviare più messaggi   
alla coda SQS utilizzando AWS SDK V2 e LocalStack. La guida copre la configurazione del progetto,   
la creazione della coda SQS su LocalStack e il test dell'applicazione con Postman.

## 2. Configurazione del Progetto Spring Boot

### 2.1 Modifica del Controller

Modifica il controller per accettare una lista di messaggi in formato JSON   
e inviarli alla coda SQS.

@RestController  
@RequestMapping("/api/sqs")  
public class SqsController {  
  
 private final SqsDelegate sqsDelegate;  
  
 public SqsController(SqsDelegate sqsDelegate) {  
 this.sqsDelegate = sqsDelegate;  
 }  
  
 @PostMapping("/send-batch")  
 public String sendMessages(@RequestBody List<Message> messages) {  
 messages.forEach(sqsDelegate::handleSendMessage);  
 return "Inviati " + messages.size() + " messaggi alla coda SQS.";  
 }  
}

### 2.2 Classe Message

La classe Message rappresenta ogni messaggio inviato alla coda.

@Data  
@NoArgsConstructor  
@AllArgsConstructor  
public class Message {  
 private String body;  
}

### 2.3 Delegate e Servizio

Il Delegate e il Service si occupano di inviare i messaggi alla coda SQS.

@Component  
public class SqsDelegate {  
 private final SqsService sqsService;  
  
 public SqsDelegate(SqsService sqsService) {  
 this.sqsService = sqsService;  
 }  
  
 public void handleSendMessage(Message message) {  
 sqsService.sendMessage(message.getBody());  
 }  
}

@Service  
public class SqsService {  
 private final SqsClient sqsClient;  
  
 @Value("${cloud.aws.sqs.queue-url}")  
 private String queueUrl;  
  
 public SqsService(SqsClient sqsClient) {  
 this.sqsClient = sqsClient;  
 }  
  
 public void sendMessage(String messageBody) {  
 SendMessageRequest request = SendMessageRequest.builder()  
 .queueUrl(queueUrl)  
 .messageBody(messageBody)  
 .build();  
  
 sqsClient.sendMessage(request);  
 }  
}

## 3. Configurazione della Coda SQS

### 3.1 Configurazione AWS SDK V2

@Bean  
public SqsClient sqsClient() {  
 return SqsClient.builder()  
 .region(Region.US\_EAST\_1)  
 .endpointOverride(URI.create("http://localhost:4566"))  
 .credentialsProvider(StaticCredentialsProvider.create(  
 AwsBasicCredentials.create("test", "test")))  
 .build();  
}

### 3.2 File application.properties

cloud.aws.sqs.queue-url=http://localhost:4566/000000000000/my-queue

### 3.3 Creazione della Coda SQS

Esegui il seguente comando AWS CLI per creare una coda SQS su LocalStack:

aws --endpoint-url=http://localhost:4566 sqs create-queue --queue-name my-queue

## 4. Test dell'Applicazione

### 4.1 Avvia LocalStack

Esegui LocalStack con il comando:

docker run --rm -it -p 4566:4566 localstack/localstack

### 4.2 Avvia l'Applicazione Spring Boot

Esegui l'applicazione con Maven:

mvn spring-boot:run

### 4.3 Invia Più Messaggi con Postman

Configura la richiesta POST su Postman:

\*\*URL\*\*: http://localhost:8080/api/sqs/send-batch

\*\*Body (JSON):\*\*

[  
 {"body": "Primo messaggio"},  
 {"body": "Secondo messaggio"},  
 {"body": "Terzo messaggio"}  
]

### 4.4 Verifica i Messaggi sulla Coda

Ricevi i messaggi dalla coda con il comando AWS CLI:

aws --endpoint-url=http://localhost:4566 sqs receive-message --queue-url http://localhost:4566/000000000000/my-queue --max-number-of-messages 10