
Amazon Polly

Guia do desenvolvedor



Amazon Polly: Guia do desenvolvedor

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens de marcas da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

O que é o Amazon Polly?	1
Você é um usuário iniciante do Amazon Polly?	2
Como ele funciona	3
Próximas etapas	3
Conceitos básicos	4
Etapa 1: Configurar uma conta e um usuário	4
Etapa 1.1: Cadastre-se no AWS	4
Etapa 1.2: Criar um usuário do IAM	5
Próxima etapa	5
Etapa 2: Conceitos básicos (console)	5
Exercício 1: Início rápido de sintetização de fala (console)	6
Exercício 2: Sintetização de fala com entrada de texto sem formatação (console)	6
Próxima etapa	7
Etapa 3: Conceitos básicos (AWS CLI)	7
Etapa 3.1: Configurar a AWS CLI	7
Etapa 3.2: Exercício de conceitos básicos	9
Exemplos de Python	10
Configurar o Python e testar um exemplo (SDK)	10
Vozes no Amazon Polly	12
Vozes disponíveis	12
Vozes bilíngues	14
Vozes bilíngues com sotaque	14
Vozes totalmente bilíngues	15
Como ouvir as vozes	16
Velocidade da voz	16
Modificar a velocidade da voz	17
Idiomas compatíveis com o Amazon Polly	18
Tabelas de fonemas e visemas para os idiomas compatíveis	19
TTS neural	90
Compatibilidade de recursos e regiões	90
O mecanismo de voz	91
Escolher o mecanismo de voz (Console)	91
Escolher o mecanismo de voz (CLI)	92
Vozes neurais	93
Estilo de locutor do NTTS	94
Marcas de fala	96
Tipos de marca de fala	96
Visemas e Amazon Polly	96
Uso de marcas de fala	97
Solicitação de marcas de fala	97
Saída de marca de fala	98
Exemplos de marca de fala	99
Solicitação de marcas de fala (Console)	100
Uso de SSML	102
Caracteres reservados	102
Usar SSML no console	104
Como usar o SSML no AWS CLI	105
Como usar o SSML com o comando SynthesizeSpeech	105
Sintetizar um documento aprimorado por SSML	106
Como usar o SSML para tarefas comuns do Amazon Polly	107
Tags SSML compatíveis	109
Identificar um texto aprimorado por SSML	110
Adicionar uma pausa	111
Enfatizar palavras	111

Especificar outro idioma para palavras específicas	112
Colocar uma tag personalizada no texto	113
Adicionar uma pausa entre parágrafos	113
Usar a pronúncia fonética	114
Controlar o volume, a velocidade e o tom da fala	115
Configurar uma duração máxima para fala sintetizada	116
Adicionar uma pausa entre as frases	119
Controlar como tipos especiais de palavras são ditas	119
Como pronunciar acrônimos e abreviaturas	122
Melhorar a pronúncia especificando partes da fala	122
Adicionar o som da respiração	123
Estilo de locutor	126
Adicionar compactação de intervalo dinâmico	126
Falar suavemente	128
Controlar o timbre	128
Sussurrar	129
Gerenciamento de léxicos	131
Aplicação de vários léxicos	131
Gerenciamento de léxicos usando o console do	132
Upload dos léxicos usando o console	132
Aplicação dos léxicos usando o console (sintetizar fala)	133
Filtro da lista de léxicos usando o console	134
Download dos léxicos usando o console	134
Exclusão de um léxico usando o Console	135
Gerenciamento de léxicos usando a AWS CLI	135
PutLexicon	135
GetLexicon	140
ListLexicons	140
DeleteLexicon	141
Criar arquivos de áudio longos	142
Configurar a política do IAM para a síntese assíncrona	142
Criar arquivos de áudio longos (console)	143
Criar arquivos de áudio longos (CLI)	144
Exemplos de código e de aplicativo	147
Código de exemplo	147
Exemplos de Java	147
Exemplos do Python	153
Aplicações de exemplo	158
Exemplo Python	158
Exemplo de Java	168
Exemplo de iOS	172
Exemplo no Android	174
Amazon Polly para Windows (SAPI)	176
Instalação e configuração do Amazon Polly para Windows (SAPI)	176
Criar um usuário do IAM para oAWSCliente	176
Instalar aAWS CLIpara Windows	177
Criar um perfil para oAWSCliente	177
Instalar o plug-in Amazon Polly para Windows	177
Usando o Amazon Polly em aplicativos	178
AWSPlug-in do WordPress	181
Pré-requisitos de instalação	181
Criar uma conta da AWS	181
Criação de um usuário do IAM	182
Criar um site WordPress	183
Instalar e configurar o plug-in	183
Configurar o plug-in	184
Personalizar o WordPress	185

Ajuste as configurações de plug-in para ajustar os arquivos de áudio	185
Use SSML no seu conteúdo para modificar como ele será falado	185
Use tags de Apenas áudio e Apenas palavras em seu conteúdo	185
Adição de texto traduzido em sua postagem	186
Amazon Pollycast	187
Posicionar o reprodutor	187
Como armazenar os arquivos de áudio	188
Cotas	189
Regiões compatíveis	189
Throttling	189
Léxicos de pronúncia	189
Operação de API SynthesizeSpeech	190
Operações de API SpeechSynthesisTask	190
Speech Synthesis Markup Language (SSML)	190
Segurança	191
Proteção de dados	191
Criptografia em repouso	192
Criptografia em trânsito	192
Privacidade do tráfego entre redes	192
Identity and Access Management	192
Audience	192
Autenticação com identidades	193
Gerenciamento do acesso usando políticas	195
Como o Amazon Polly funciona com o IAM	196
Exemplos de políticas baseadas em identidade	198
Referência de permissões da API do Amazon Polly	203
Registro em log e monitoramento	204
Validação de conformidade	204
Resiliência	205
Segurança da infraestrutura	205
Melhores práticas de segurança	205
Registrar chamadas de API do Amazon Polly com AWS CloudTrail	206
Informações do Amazon Polly no CloudTrail	206
Exemplo: Entradas dos arquivos de log do Amazon Polly	207
Integração do CloudWatch	209
Obmissão das métricas do CloudWatch (Console)	209
Obmissão das métricas do CloudWatch (CLI)	209
Métricas do Amazon Polly	210
Dimensões para métricas do Amazon Polly	211
Referência de API	212
Actions	212
DeleteLexicon	213
DescribeVoices	215
GetLexicon	218
GetSpeechSynthesisTask	220
ListLexicons	222
ListSpeechSynthesisTasks	224
PutLexicon	227
StartSpeechSynthesisTask	229
SynthesizeSpeech	235
Tipos de dados	239
Lexicon	240
LexiconAttributes	241
LexiconDescription	243
SynthesisTask	244
Voice	247
Histórico do documento	249

AWSGlossário	253
.....	ccliv

O que é o Amazon Polly?

O Amazon Polly é um serviço de nuvem que converte texto em fala realista. Você pode usar o Amazon Polly para desenvolver aplicações que aumentam o envolvimento e a acessibilidade. O Amazon Polly oferece suporte a vários idiomas e inclui diversas vozes realistas, para que você possa criar aplicações compatíveis com fala que funcionam em vários locais e use a voz ideal para os seus clientes. Com o Amazon Polly, você paga apenas pelo texto que sintetizar. Você também pode armazenar em cache e reproduzir as falas geradas pelo Amazon Polly sem custo adicional.

Além disso, o Amazon Polly inclui diversas vozes de texto em fala neural (NTTS), oferecendo melhorias inovadoras na qualidade da fala por meio de uma nova abordagem de machine learning, oferecendo assim aos clientes as vozes de conversão de texto em fala mais naturais e semelhantes possíveis à voz humana. A tecnologia de TTS neural também oferece suporte a um estilo de locutor que é personalizado para casos de uso de narração jornalística.

Os casos de uso comuns para o Amazon Polly incluem, mas não de forma exclusiva, aplicações móveis, como leitores de notícias, jogos, plataformas de eLearning, aplicações de acessibilidade para deficientes visuais e o segmento em franco crescimento da Internet das Coisas (IoT).

O Amazon Polly é certificado para uso com cargas de trabalho regulamentadas para a HIPAA (Lei de Portabilidade e Responsabilidade de Seguros de Health de 1996) e o Padrão de Segurança de Dados do Setor de Cartões de Pagamento (PCI DSS).

Alguns dos benefícios de usar o Amazon Polly incluem:

- **Alta qualidade**— O Amazon Polly oferece a nova tecnologia TTS neural e a melhor tecnologia TTS padrão da categoria para sintetizar a fala natural com alta precisão de pronúncia (incluindo abreviaturas, expansões de acrônimos, interpretações de data/hora e desambiguação de homógrafos).
- **Baixa latência**— O Amazon Polly garante tempos de resposta rápidos, o que faz dele uma opção viável para casos de uso de baixa latência, como sistemas de diálogo.
- **Support para um grande portfólio de idiomas e vozes**— O Amazon Polly oferece suporte a dezenas de vozes e vários idiomas, oferecendo opções de vozes masculinas e femininas para a maioria dos idiomas. Atualmente, o TTS neural é compatível com três vozes do inglês britânico e oito vozes do inglês dos EUA. Esse número continuará a aumentar à medida que colocarmos mais vozes neurais online. As vozes do inglês dos EUA, Matthew e Joanna, também podem usar o estilo de fala neural de locutor, semelhante ao que você pode ouvir de um âncora de notícias profissional.
- **Econômica**— O modelo de pagamento conforme o uso do Amazon Polly significa que não há custos de configuração. Você pode começar pequeno e escalonar conforme o crescimento do aplicativo.
- **Solução baseada em nuvem**— Soluções de TTS no próprio dispositivo exigem recursos de computação significativos, particularmente potência de CPU, RAM e espaço em disco. Isso pode resultar em custo de desenvolvimento mais alto e consumo de energia em dispositivos como tablets, smartphones, etc. Por outro lado, conversão de texto em fala no Nuvem AWS reduz drasticamente os requisitos de recursos locais. Isso permite o suporte a todos os idiomas e vozes disponíveis com a melhor qualidade possível. Além disso, os aprimoramentos das falas são disponibilizados instantaneamente a todos os usuários finais, sem necessidade de atualizações adicionais nos dispositivos.

Você é um usuário iniciante do Amazon Polly?

Caso seja um usuário iniciante do Amazon Polly, recomendamos que você leia as seguintes seções no pedido listado:

1. [Como funciona o Amazon Polly \(p. 3\)](#)— Esta seção apresenta várias entradas e opções do Amazon Polly com as quais você pode trabalhar a fim de criar uma experiência completa.
2. [Conceitos básicos da Amazon Polly \(p. 4\)](#)— Nesta seção, você abre sua conta e testa a síntese de fala do Amazon Polly.
3. [Aplicações de exemplo \(p. 158\)](#)— Esta seção apresenta exemplos adicionais que você pode usar para explorar o Amazon Polly.

Como funciona o Amazon Polly

O Amazon Polly converte o texto inserido em fala realista. Chame um dos métodos de síntese de fala, forneça o texto que deseja sintetizar, selecione uma das vozes disponíveis de texto em fala neural (NTTS) ou texto em fala padrão (TTS) e especifique um formato de saída de áudio. Em seguida, o Amazon Polly sintetizará o texto fornecido em um stream de áudio de fala de alta qualidade.

- **Texto inserido**— Forneça o texto que deseja sintetizar para o Amazon Polly retornar um stream de áudio. É possível fornecer a entrada como texto sem formatação ou em formato SSML (Speech Synthesis Markup Language). Com SSML, você pode controlar diversos aspectos da fala, como pronúncia, volume, tom e velocidade da fala. Para obter mais informações, consulte [Geração de fala a partir de documentos SSML \(p. 102\)](#).
- **Vozes disponíveis**— O Amazon Polly oferece um portfólio de diversos idiomas e uma variedade de vozes, incluindo uma voz bilingue (para inglês e hindi). Para a maioria dos idiomas, é possível escolher entre várias vozes, masculinas e femininas. Ao iniciar uma tarefa de síntese de fala, especifique o ID de voz e o Amazon Polly usa essa voz para converter o texto em fala. O Amazon Polly não é um serviço de tradução. A fala é sintetizada no mesmo idioma que o texto. No entanto, se o texto estiver em um idioma diferente do designado para a voz, os números representados como dígitos (por exemplo, 53, não cinquenta e três) serão sintetizados no idioma da voz e não no idioma do texto. Para obter mais informações, consulte [Vozes no Amazon Polly](#).
- **Formato de saída**— O Amazon Polly pode fornecer fala sintetizada em vários formatos. Selecione o formato de áudio que atenda às suas necessidades. Por exemplo, você pode solicitar a fala no formato MP3 ou Ogg Vorbis para consumo por aplicativos móveis e web. Ou, pode solicitar o formato de saída PCM para consumo por dispositivos AWS IoT e soluções de telefonia.

Próximas etapas

Caso esteja iniciando o Amazon Polly, recomendamos que leia os tópicos a seguir, em ordem:

- [Conceitos básicos da Amazon Polly \(p. 4\)](#)
- [Aplicações de exemplo \(p. 158\)](#)
- [Cotas no Amazon Polly \(p. 189\)](#)

Conceitos básicos da Amazon Polly

O Amazon Polly fornece operações de API simples que você pode integrar com facilidade às aplicações existentes. Para ver uma lista das operações suportadas, consulte [Actions \(p. 212\)](#). Você pode usar uma das opções a seguir:

- **AWSSDKs** — Ao usar os SDKs, suas solicitações ao Amazon Polly são automaticamente assinadas e autenticadas usando as credenciais fornecidas por você. Esta é a escolha recomendada para a criação de suas aplicações.
- **AWS CLI** — Você pode usar o AWS CLI para acessar qualquer uma das funcionalidades do Amazon Polly sem necessidade de escrever código algum.

As seções a seguir descrevem como configurar e fornecer um exercício introdutório.

Tópicos

- [Etapa 1: Configurar uma solicitação AWS Conta e Criar um Usuário \(p. 4\)](#)
- [Etapa 2: Conceitos básicos \(console\) \(p. 5\)](#)
- [Etapa 3: Conceitos básicos \(AWS CLI\) \(p. 7\)](#)
- [Exemplos de Python \(p. 10\)](#)

Etapa 1: Configurar uma solicitação AWS Conta e Criar um Usuário

Antes de usar o Amazon Polly pela primeira vez, conclua as seguintes tarefas:

1. [Etapa 1.1: Cadastre-se no AWS \(p. 4\)](#)
2. [Etapa 1.2: Criar um usuário do IAM \(p. 5\)](#)

Etapa 1.1: Cadastre-se no AWS

Ao se cadastrar na Amazon Web Services (AWS), sua AWS conta é automaticamente cadastrada em todos os serviços do AWS, incluindo Amazon Polly. Você será cobrado apenas pelos serviços que usar.

Com o Amazon Polly, você paga apenas pelos recursos que usa. Se você for um novo cliente AWS, você pode começar a usar o Amazon Polly gratuitamente. Para obter mais informações, consulte [Nível de uso gratuito da AWS](#).

Se já tiver uma conta da AWS, passe para a próxima etapa. Se você não tiver uma solicitação AWS, execute as etapas no procedimento a seguir para criar uma.

Para criar uma conta da AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga as instruções online.

Parte do procedimento de cadastro envolve uma chamada telefônica e a digitação de um código de verificação usando o teclado do telefone.

Observe o seu AWSID da conta porque você precisará dele na próxima etapa.

Etapa 1.2: Criar um usuário do IAM

Serviços em AWS, como o Amazon Polly, exigem que você forneça credenciais ao acessá-los, para que o serviço possa determinar se você tem permissões para acessar os recursos pertencentes a esse serviço. O console requer sua senha. Você pode criar chaves de acesso para a conta da AWS para acessar a AWS CLI ou a API. No entanto, não recomendamos que você acesse a AWS usando as credenciais de sua conta da AWS. Em vez disso, recomendamos o uso de AWS Identity and Access Management (IAM). Crie um usuário do IAM, adicione o usuário a um grupo do IAM com permissões administrativas e, em seguida, conceda permissões administrativas ao usuário do IAM criado. Em seguida, você pode acessar a AWS usando uma URL especial e as credenciais desse usuário do IAM.

Se você se inscreveu no AWS, mas você ainda não criou um usuário do IAM para si mesmo, crie um usando o console do IAM.

Os exercícios deste guia pressupõem que você tenha um usuário do IAM (`adminuser`) com privilégios de administrador. Siga o procedimento para criar `adminuser` na conta.

Para criar um usuário administrador e fazer login no console

1. Crie um usuário administrador chamado `adminuser`, em sua conta da AWS. Para obter instruções, consulte [Criação do primeiro grupo de usuários e administradores do IAM](#) no Guia do usuário do IAM.
2. Um usuário pode fazer login no AWS Management Console usando um URL especial. Para obter mais informações, consulte [Como usuários fazem login em sua conta](#) no Guia do usuário do IAM.

Important

Os exercícios dos conceitos básicos usam as credenciais do usuário administrador. Para uma segurança maior, ao compilar e testar a aplicação de produção, recomendamos que você crie um usuário do administrador específico do serviço que tenha permissões somente para as ações do Amazon Polly. Para um exemplo de política que concede permissões específicas do Amazon Polly, consulte [Exemplo 1: Permitir todas as ações do Amazon Polly \(p. 201\)](#).

Para obter mais informações sobre IAM, consulte o seguinte:

- [AWS Identity and Access Management \(IAM\)](#)
- [Conceitos básicos](#)
- [IAM User Guide](#)

Próxima etapa

[Etapa 2: Conceitos básicos \(console\) \(p. 5\)](#)

Etapa 2: Conceitos básicos (console)

O console do Amazon Polly é a maneira mais fácil de começar a testar e usar a síntese de fala do Amazon Polly. O console do Amazon Polly oferece suporte à síntese de fala com base em texto sem formatação ou entrada SSML.

Tópicos

- [Exercício 1: Início rápido de sintetização de fala \(console\) \(p. 6\)](#)
- [Exercício 2: Sintetização de fala com entrada de texto sem formatação \(console\) \(p. 6\)](#)
- [Próxima etapa \(p. 7\)](#)

Exercício 1: Início rápido de sintetização de fala (console)

O início rápido o orienta quanto à maneira mais rápida de testar a síntese de fala do Amazon Polly quanto à qualidade da fala. Quando você seleciona a opção `Text-to-Speech`, o campo de texto para informar seu texto é pré-carregado com texto de exemplo, para que você possa experimentar rapidamente o Amazon Polly.

Como testar o Amazon Polly (Console)

1. Faça login no `AWS Management Console` e abra o console Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Selecione a guia `Text-to-Speech`.
3. Desligue o `SML`.
4. Under `Mecanismo`, escolha `Standard` ou `Neural`.
5. Escolha um idioma e `AWS Região` e, depois, escolha uma voz. Se escolher o `Neural` para `Mecanismo`, apenas os idiomas e vozes que suportam `NTTS` estão disponíveis. Todas as vozes padrão estão desabilitadas.
6. Selecione `Oiça`.

Para testes mais aprofundados, consulte os tópicos a seguir:

- [Exercício 2: Sintetização de fala com entrada de texto sem formatação \(console\) \(p. 6\)](#)
- [Usar SSML \(Console\) \(p. 104\)](#)
- [Aplicação dos léxicos usando o console \(sintetizar fala\) \(p. 133\)](#)

Exercício 2: Sintetização de fala com entrada de texto sem formatação (console)

O procedimento a seguir sintetiza a fala usando entrada de texto sem formatação. Observe como "W3C" e a data "10/3" (3 de outubro) são sintetizados.

Como sintetizar fala usando entrada de texto sem formatação (console)

1. Depois de fazer login no console do Amazon Polly, selecione `Experimente Amazon Polly` e, em seguida, escolha a `Text-to-Speech Guia`.
2. Desligue o `SML`.
3. Digite ou cole este texto na caixa de entrada.

```
He was caught up in the game.  
In the middle of the 10/3/2014 W3C meeting  
he shouted, "Score!" quite loudly.
```

4. Em `Engine (Mecanismo)`, escolha `Standard` ou `Neural`.

5. Escolha um idioma e uma AWS Região e, depois, escolha uma voz. Se escolher o `neural` para o mecanismo, apenas os idiomas e vozes que suportam NTTS estão disponíveis. Todas as vozes padrão estão desabilitadas.
6. Para ouvir o discurso imediatamente, selecione `Oiça`.
7. Para salvar a fala em um arquivo, siga uma das etapas a seguir:
 - a. Escolha `Download`.
 - b. Para alterar para um formato de arquivo diferente, expanda `Configurações adicionais`, ative `Configurações do formato do arquivo de fala`, escolha o formato de arquivo desejado e, em seguida, escolha `Baixar`.

Para exemplos mais detalhados, consulte os tópicos a seguir:

- [Aplicação dos léxicos usando o console \(sintetizar fala\) \(p. 133\)](#)
- [Usar SSML \(Console\) \(p. 104\)](#)

Próxima etapa

[Etapa 3: Conceitos básicos \(AWS CLI\) \(p. 7\)](#)

Etapa 3: Conceitos básicos (AWS CLI)

É possível executar quase todas as operações do Amazon Polly que podem ser executadas usando o console do Amazon Polly usando a AWS Command Line Interface (AWS CLI). Não é possível ouvir a fala sintetizada usando a AWS CLI. Você deve salvá-la em um arquivo e depois abrir o arquivo em uma aplicação que possa reproduzi-la.

Tópicos

- [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\) \(p. 7\)](#)
- [Etapa 3.2: Exercício de conceitos básicos usando o AWS CLI \(p. 9\)](#)

Etapa 3,1: Configurar a AWS Command Line Interface (AWS CLI)

Siga as etapas para fazer download e configurar a AWS CLI.

Important

A AWS CLI não é necessária para executar as etapas neste exercício. No entanto, alguns dos exercícios neste guia usam a AWS CLI. Você pode ignorar esta etapa e ir até [Etapa 3.2: Exercício de conceitos básicos usando o AWS CLI \(p. 9\)](#) e, em seguida, configurar a AWS CLI depois quando precisar dela.

Para configurar a AWS CLI

1. Faça download e configure a AWS CLI. Para obter instruções, consulte os tópicos a seguir no AWS Command Line Interface Guia do usuário:
 - [Começar a usar a AWS Command Line Interface](#)

- [Configuração do AWS Command Line Interface](#)

2. Adicione um perfil nomeado para o usuário administrador no arquivo config da AWS CLI. Você poderá usar esse perfil ao executar os comandos da AWS CLI. Para obter mais informações sobre perfis nomeados, consulte [Perfis nomeados](#) no [AWS Command Line Interface Guia do usuário](#).

```
[profile adminuser]
aws_access_key_id = adminuser access key ID
aws_secret_access_key = adminuser secret access key
region = aws-region
```

Para obter uma lista de disponíveis AWS Regiões e aquelas suportadas pelo Amazon Polly, consulte [Regiões e endpoints](#) do [Referência geral da Amazon Web Services](#).

Note

Se estiver usando a região compatível com o Amazon Polly que especificou ao configurar a AWS CLI, omita a seguinte linha do AWS CLI Exemplos de código do .

```
--region aws-region
```

3. Verifique a configuração digitando o comando de ajuda a seguir no prompt de comando.

```
aws help
```

Uma lista de válidos AWS devem aparecer na caixa de diálogo AWS CLI Janela.

Para habilitar o Amazon Polly no AWS CLI (opcional)

Se você já tiver feito download e configurado a anteriormente, o AWS CLI, o Amazon Polly pode não estar disponível, a menos que você reconfigure a AWS CLI. Esse procedimento verifica se isso é necessário e fornece instruções, caso o Amazon Polly não esteja disponível automaticamente.

1. Verifique a disponibilidade do Amazon Polly digitando o comando de ajuda a seguir no AWS CLIPrompt.

```
aws polly help
```

Se uma descrição do Amazon Polly e uma lista de comandos válidos forem exibidas na AWS CLI, o Amazon Polly está disponível na janela AWS CLI e pode ser usado imediatamente. Nesse caso, você poderá ignorar o restante deste procedimento. Se isso não for exibido, vá para a etapa 2.

2. Use uma das duas opções a seguir para habilitar o Amazon Polly:

- a. Desinstale e reinstale a AWS CLI.

Para obter instruções, consulte [Instalar a AWS Command Line Interface](#) no [AWS Command Line Interface Guia do usuário](#).

ou

- b. Faça download do arquivo [service-2.json](#).

No prompt de comando, execute o seguinte comando da .

```
aws configure add-model --service-model file://service-2.json --service-name polly
```

3. Verifique novamente a disponibilidade do Amazon Polly.

```
aws polly help
```

A descrição do Amazon Polly deve estar visível.

Próxima etapa

[Etapa 3.2: Exercício de conceitos básicos usando oAWS CLI \(p. 9\)](#)

Etapa 3.2: Exercício de conceitos básicos usando oAWS CLI

Agora você pode testar a síntese de fala oferecida pelo Amazon Polly. Neste exercício, você acessa a operação `SynthesizeSpeech` ao inserir texto de amostra. Você pode salvar o áudio resultante como arquivo e verificar seu conteúdo.

1. Execute o comando `synthesize-speech` da AWS CLI para sintetizar texto de exemplo em um arquivo de áudio (`hello.mp3`).

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (`\`) no final de cada linha por um circunflexo (`^`) e use aspas completas (`"`) ao redor do texto de entrada por aspas simples (`'`) para tags internas.

```
aws polly synthesize-speech \
  --output-format mp3 \
  --voice-id Joanna \
  --text 'Hello, my name is Joanna. I learned about the W3C on 10/3 of last year.' \
  hello.mp3
```

Na chamada para `synthesize-speech`, você fornece texto de exemplo para a síntese, a voz a ser usada (ao fornecer uma ID de voz, explicado na etapa 3 a seguir) e o formato de saída. O comando salva o áudio resultante no arquivo `hello.mp3`.

Além do arquivo MP3, a operação acima envia a saída a seguir para o console.

```
{
  "ContentType": "audio/mpeg",
  "RequestCharacters": "71"
}
```

2. Reproduza o arquivo `hello.mp3` resultante para verificar a fala sintetizada.
3. Obtenha a lista de vozes disponíveis usando a operação `DescribeVoices`. Execute o seguinte comando `describe-voices` da AWS CLI.

```
aws polly describe-voices
```

Em resposta, o Amazon Polly retorna a lista de todas as vozes disponíveis. Para cada voz, a resposta fornece os seguintes metadados: ID da voz, código do idioma, nome do idioma e sexo da voz. Veja a seguir uma resposta de exemplo.

```
{
  "Voices": [
    {
      "Gender": "Female",
```

```
        "Name": "Salli",
        "LanguageName": "US English",
        "Id": "Salli",
        "LanguageCode": "en-US"
    },
    {
        "Gender": "Female",
        "Name": "Joanna",
        "LanguageName": "US English",
        "Id": "Joanna",
        "LanguageCode": "en-US"
    }
  ]
}
```

Opcionalmente, você pode especificar o código do idioma para encontrar as vozes disponíveis para um idioma específico. Amazon Polly é compatível com dezenas de vozes. O exemplo a seguir lista todas as vozes de português do Brasil.

```
aws polly describe-voices \
  --language-code pt-BR
```

Para obter uma lista de códigos de idioma, consulte [Idiomas compatíveis com o Amazon Polly](#) (p. 18). Esses códigos de idioma são tags de identificação de idioma W3C (*código ISO 639 para o nome do idioma*–*código ISO 3166 para o país*). Por exemplo, en-US (inglês norte-americano), en-GB (inglês britânico) e es-ES (espanhol), etc.

Também é possível usar a opção `help` na AWS CLI para obter a lista de códigos de idioma:

```
aws polly describe-voices help
```

Exemplos de Python

Este guia fornece exemplos adicionais, sendo que alguns são exemplos de código Python que usam o AWS SDK for Python (Boto) para fazer chamadas de API para o Amazon Polly. Recomendamos que você configure o Python e teste o código de exemplo fornecido na seção a seguir. Para obter exemplos adicionais, consulte [Aplicações de exemplo](#) (p. 158).

Configurar o Python e testar um exemplo (SDK)

Para testar o código de exemplo do Python, você precisará do AWS SDK for Python (Boto). Para obter instruções, consulte [AWS SDK for Python \(Boto3\)](#).

Como testar o código Python de exemplo

O exemplo de código Python a seguir realiza as seguintes ações:

- Usa o AWS SDK for Python (Boto) para enviar uma solicitação `synthesizeSpeech` ao Amazon Polly (ao fornecer texto sem formatação como entrada).
- Acessa o fluxo de áudio resultante na resposta e salva o áudio em um arquivo (`speech.mp3`) no seu disco local.
- Reproduz o arquivo de áudio com o player de áudio padrão do seu sistema local.

Salve o código em um arquivo (`exemplo.py`) e execute-o.


```
"""Getting Started Example for Python 2.7+/3.3+"""
from boto3 import Session
from botocore.exceptions import BotoCoreError, ClientError
from contextlib import closing
import os
import sys
import subprocess
from tempfile import gettempdir

# Create a client using the credentials and region defined in the [adminuser]
# section of the AWS credentials file (~/.aws/credentials).
session = Session(profile_name="adminuser")
polly = session.client("polly")

try:
    # Request speech synthesis
    response = polly.synthesize_speech(Text="Hello world!", OutputFormat="mp3",
                                       VoiceId="Joanna")
except (BotoCoreError, ClientError) as error:
    # The service returned an error, exit gracefully
    print(error)
    sys.exit(-1)

# Access the audio stream from the response
if "AudioStream" in response:
    # Note: Closing the stream is important because the service throttles on the
    # number of parallel connections. Here we are using contextlib.closing to
    # ensure the close method of the stream object will be called automatically
    # at the end of the with statement's scope.
    with closing(response["AudioStream"]) as stream:
        output = os.path.join(gettempdir(), "speech.mp3")

        try:
            # Open a file for writing the output as a binary stream
            with open(output, "wb") as file:
                file.write(stream.read())
        except IOError as error:
            # Could not write to file, exit gracefully
            print(error)
            sys.exit(-1)
    else:
        # The response didn't contain audio data, exit gracefully
        print("Could not stream audio")
        sys.exit(-1)

# Play the audio using the platform's default player
if sys.platform == "win32":
    os.startfile(output)
else:
    # The following works on macOS and Linux. (Darwin = mac, xdg-open = linux).
    opener = "open" if sys.platform == "darwin" else "xdg-open"
    subprocess.call([opener, output])
```

Para obter mais exemplos, incluindo um exemplo de aplicação, consulte [Aplicações de exemplo \(p. 158\)](#).

Vozes no Amazon Polly

Vozes disponíveis

O Amazon Polly fornece uma variedade de vozes diferentes em vários idiomas para sintetizar a fala do texto.

Linguagem	Nome/ID	Sexo	Voz neural	Voz padrão
Árabe (arb)	Zeina	Feminino	Não	Sim
Chinês, mandarim (cmn-CN)	Zhiyu	Feminino	Não	Sim
Dinamarquês (da-DK)	Naja	Feminino	Não	Sim
	Mads	Masculino	Não	Sim
Holandês (nl-NL)	Lotte	Feminino	Não	Sim
	Ruben	Masculino	Não	Sim
Inglês australiano (en-AU)	Nicole	Feminino	Não	Sim
	Olivia	Feminino	Sim	Não
	Russell	Masculino	Não	Sim
Inglês britânico (en-GB)	Amy**	Feminino	Sim	Sim
	Emma	Feminino	Sim	Sim
	Brian	Masculino	Sim	Sim
Inglês indiano (en-IN)	Aditi*	Feminino	Não	Sim
	Raveena	Feminino	Não	Sim
Inglês dos EUA (en-US)	Ivy	Feminino (criança)	Sim	Sim
	Joanna**	Feminino	Sim	Sim
	Kendra	Feminino	Sim	Sim
	Kimberly	Feminino	Sim	Sim
	Salli	Feminino	Sim	Sim
	Joey	Masculino	Sim	Sim
	Justin	Masculino (criança)	Sim	Sim
	Kevin	Masculino (criança)	Sim	Não
	Matthew**	Masculino	Sim	Sim

Linguagem	Nome/ID	Sexo	Voz neural	Voz padrão
Inglês galês (en-GB-WLS)	Geraint	Masculino	Não	Sim
Francês (fr-FR)	Céline/Celine	Feminino	Não	Sim
	Léa	Feminino	Não	Sim
	Mathieu	Masculino	Não	Sim
Francês canadense (fr-CA)	Chantal	Feminino	Não	Sim
	Gabrielle	Feminino	Sim	Não
Alemão (de-DE)	Marlene	Feminino	Não	Sim
	Vicki	Feminino	Sim	Sim
	Hans	Masculino	Não	Sim
Hindi (hi-IN)	Aditi*	Feminino	Não	Sim
Islandês (is-IS)	Dóra/Dora	Feminino	Não	Sim
	Karl	Masculino	Não	Sim
Italiano (it-IT)	Carla	Feminino	Não	Sim
	Bianca	Feminino	Não	Sim
	Giorgio	Masculino	Não	Sim
Japonês (ja-JP)	Mizuki	Feminino	Não	Sim
	Takumi	Masculino	Não	Sim
Coreano (ko-KR)	Seoyeon	Feminino	Sim	Sim
Norueguês (nb-NO)	Liv	Feminino	Não	Sim
Polonês (pl-PL)	Ewa	Feminino	Não	Sim
	Maja	Feminino	Não	Sim
	Jacek	Masculino	Não	Sim
	Jan	Masculino	Não	Sim
Português brasileiro (pt-BR)	Camila	Feminino	Sim	Sim
	Vitória/Vitoria	Feminino	Não	Sim
	Ricardo	Masculino	Não	Sim
Português europeu (pt-PT)	Inês/Ines	Feminino	Não	Sim
	Cristiano	Masculino	Não	Sim

Linguagem	Nome/ID	Sexo	Voz neural	Voz padrão
Romeno (ro-RO)	Carmen	Feminino	Não	Sim
Russo (ru-RU)	Tatyana	Feminino	Não	Sim
	Maxim	Masculino	Não	Sim
Espanhol europeu (es-ES)	Conchita	Feminino	Não	Sim
	Lúcia	Feminino	Não	Sim
	Enrique	Masculino	Não	Sim
Espanhol mexicano (es-MX)	Mia	Feminino	Não	Sim
Espanhol dos EUA (es-US)	Lupe**	Feminino	Sim	Sim
	Penélope/ Penelope	Feminino	Não	Sim
		Masculino	Não	Sim
	Miguel			
Sueco (sv-SE)	Astrid	Feminino	Não	Sim
Turco (tr-TR)	Filiz	Feminino	Não	Sim
Galês (cy-GB)	Gwyneth	Feminino	Não	Sim

* Esta voz é bilíngue e pode falar inglês e hindi. Para obter mais informações, consulte [Vozes bilíngues \(p. 14\)](#).

** Essas vozes podem ser usadas com estilos de locutor quando usadas com o formato neural. Para obter mais informações, consulte [Estilo de locutor do NTTS \(p. 94\)](#).

Além das vozes acima, o Amazon Polly pode criar uma voz de marca personalizada que reflete a personalidade de sua marca, proporcionando a oportunidade de oferecer vozes NTTS únicas e exclusivas aos seus clientes. Para saber mais sobre as vozes de marca do Polly, consulte [Voz de marca](#).

Vozes bilíngues

O Amazon Polly tem duas formas de produzir vozes bilíngues:

- [Vozes bilíngues com sotaque \(p. 14\)](#)
- [Vozes totalmente bilíngues \(p. 15\)](#)

Vozes bilíngues com sotaque

Vozes bilíngues com sotaque podem ser criadas usando qualquer voz do Amazon Polly, mas somente ao usar tags SSML.

Normalmente, todas as palavras no texto de entrada são faladas no idioma padrão da voz especificada que você está usando.

Por exemplo, se você estiver usando a voz da Joanna (que fala inglês dos EUA), o Amazon Polly falará a seguinte frase na voz da Joanna, sem sotaque francês:

```
< speak>  
  Why didn't she just say, 'Je ne parle pas français?'  
</ speak>
```

Neste caso, as palavras *Je ne parle pas français* são ditas como seriam se fossem em inglês.

No entanto, se você usar a voz da Joanna com a `<lang>` tag, a Amazon Polly falará a frase com a voz da Joanna em francês com sotaque americano:

```
< speak>  
  Why didn't she just say, < lang xml:lang="fr-FR">'Je ne parle pas français?'</ lang>.  
</ speak>
```

Como Joanna não é uma voz nativa do Francês, a pronúncia se baseia no idioma nativo dela, que é o Inglês dos EUA. Por exemplo, apesar da pronúncia perfeita em Francês ter uma vibrante uvular /R/ na palavra *français*, a voz em inglês dos EUA da Joanna pronuncia esse fonema como o som /r/ correspondente.

Se você usar a voz de Giorgio, que fala italiano, com o seguinte texto, Amazon Polly falará a frase na voz de Giorgio com uma pronúncia italiana:

```
< speak>  
  Mi piace Bruce Springsteen.  
</ speak>
```

Vozes totalmente bilíngues

Uma voz totalmente bilíngue como Aditi (inglês indiano e hindi) pode falar fluentemente dois idiomas. Isso oferece a capacidade de usar palavras e frases de ambos os idiomas em um único texto usando a mesma voz.

No momento, Aditi é a única voz totalmente bilíngue disponível.

Como usar uma voz bilíngue (Aditi)

Aditi fala ambos os idiomas inglês indiano (en-IN) e hindi (hi-IN) com fluência. Você pode sintetizar fala em inglês e hindi, e a voz pode alternar entre os dois idiomas, mesmo dentro da mesma frase.

O hindi pode ser usado em duas formas diferentes:

- Devanagari: "उसेन कहाँ, खेल तोह अब शुरू होगा"
- Romanagari (usando o alfabeto latino): Usne kahan, khel toh ab shuru hoga.

Além disso, é possível mistura inglês e hindi de uma ou ambas as formas em uma única frase:

- Devanagari + Inglês: "This is the song की की त्"
- Romanagari + Português: "This is the song from the movie Jaane Tu Ya Jaane Na."
- Devanagari + Romanagari + Inglês: "This is the song की की त्from the movie Jaane Tu Ya Jaane Na."

Como Aditi é uma voz bilíngue, o texto em todos esses casos será lido corretamente, já que o Amazon Polly pode diferenciar entre os idiomas e scripts.

O Amazon Polly também oferece suporte a números, datas, horas e expansão de moeda em inglês (algarismos árabes) e hindi (algarismos Devanagari). Por padrão, os algarismos árabes são lidos em inglês

indiano. Para fazer com que o Amazon Polly os leia em hindi, é necessário usar o `ohi-INO` parâmetro de código de idioma.

Como ouvir as vozes

Você pode usar o console do Amazon Polly para ouvir uma amostra de qualquer uma das vozes disponíveis no Amazon Polly

Para ouvir uma voz no Amazon Polly

1. Faça login no AWS Management Console e abra o console Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Selecione a guia Text-to-Speech.
3. Em Engine (Mecanismo), escolha Standard (Padrão) ou Neural.
4. Escolha um idioma e uma região e depois escolha uma voz
5. Insira o texto a ser falado pela voz ou use a frase padrão e, em seguida, escolha Ouvir.

Você pode escolher qualquer um dos idiomas oferecidos pelo Amazon Polly e o console exibirá as vozes disponíveis para esse idioma. Na maioria dos casos, haverá pelo menos uma voz masculina e uma feminina, muitas vezes mais de uma de cada. Apenas alguns têm apenas uma voz. Para obter uma lista completa, consulte [Vozes no Amazon Polly \(p. 12\)](#)

Note

O inventário de vozes e o número de idiomas incluído estão sempre sendo atualizados para incluir opções adicionais. Para sugerir um novo idioma ou voz, fique à vontade para fazer comentários nesta página. Infelizmente, não podemos comentar planos específicos para novos idiomas a serem lançados.

Cada voz é criada usando falantes nativos, portanto, há variações entre as vozes, ainda que no mesmo idioma. Ao selecionar uma voz para o seu projeto, você deve testar cada uma das vozes possíveis com uma passagem de texto para ver a que melhor atende às suas necessidades.

Velocidade da voz

Devido à variação natural entre as vozes, cada voz disponível falará o texto em velocidades ligeiramente diferentes. Por exemplo, com vozes do inglês dos EUA, Ivy e Joanna são um pouco mais rápidas do que Matthew ao falarem "Mary had a little lamb" e consideravelmente mais rápidas que Joey.

Como há tantas variações entre as vozes, e o grau dessa variação pode depender do texto que estiver sendo falado, nenhuma velocidade padrão (palavras por minuto) está disponível para as vozes do Amazon Polly. No entanto, você pode encontrar o tempo necessário para a voz dizer o texto selecionado usando marcações de fala. Para obter mais informações sobre marcações de fala no Amazon Polly, consulte [Uso de marcas de fala \(p. 97\)](#)

Para ver o tempo necessário aproximado para falar uma passagem de texto

1. Abra a AWS CLI.
2. Execute o seguinte código, preenchendo conforme necessário

```
aws polly synthesize-speech \
  --language-code optional language code if needed \
  --output-format json \
  --voice-id [name of desired voice] \
```

```
--text '[desired text]' \  
--speech-mark-types='["viseme"]' \  
LengthOfText.txt
```

3. Abra o LengthOfText.txt

Se o texto fosse “Mary had a little lamb”, as últimas linhas retornadas pelo Amazon Polly seriam:

```
{ "time":882, "type": "viseme", "value": "t" }  
{ "time":964, "type": "viseme", "value": "a" }  
{ "time":1082, "type": "viseme", "value": "p" }
```

O último visema, essencialmente o som para as últimas letras em "lamb", começa 1.082 milissegundos após o início da fala. Embora essa não seja exatamente a duração do áudio, é próxima e pode servir como base de comparação entre vozes.

Modificar a velocidade da voz

Para determinados aplicativos, você pode preferir a voz escolhida com mais ou menos velocidade. Se a velocidade da voz for uma preocupação, o Amazon Polly fornece a capacidade de modificar isso usando tags SSML.

Por exemplo:

Sua organização está elaborando um aplicativo que lê livros para públicos de imigrantes. O público fala inglês, mas a fluência é limitada. Nesse caso, você pode considerar reduzir a velocidade da fala para oferecer ao seu público um pouco mais de tempo para a compreensão enquanto o aplicativo está falando.

O Amazon Polly ajuda você a diminuir a velocidade da fala usando a `<prosody>` tag SSML, como em:

```
<speak>  
  In some cases, it might help your audience to <prosody rate="85%">slow  
  the speaking rate slightly to aid in comprehension.</prosody>  
</speak>
```

ou

```
<speak>  
  In some cases, it might help your audience to <prosody rate="85%">slow  
  the speaking rate slightly to aid in comprehension.</prosody>  
</speak>
```

Duas opções de velocidade estão disponíveis ao usar SSML com o Amazon Polly:

- Velocidades predefinidas: `x-slow`, `slow`, `medium`, `fast` e `x-fast`. Nesses casos, a velocidade de cada opção é aproximada, dependendo da sua voz de preferência. A opção `medium` é a velocidade normal da voz.
- n% de taxa de velocidade: pode ser usada qualquer porcentagem da velocidade da fala, entre 20% e 200%. Nesses casos, você pode escolher exatamente a velocidade desejada. No entanto, a velocidade real da voz é aproximada, dependendo da voz escolhida. 100% é considerado a velocidade normal da voz.

Como a velocidade de cada opção é aproximada e depende da voz que você escolher, recomendamos que teste a voz selecionada em diversas velocidades para ver o que exatamente atende às suas necessidades.

Para obter mais informações sobre como usar a tag `prosody` para obter o melhor efeito, consulte [Controlar o volume, a velocidade e o tom da fala](#) (p. 115)

Idiomas compatíveis com o Amazon Polly

Os seguintes idiomas são compatíveis com o Amazon Polly e podem ser usados para sintetizar a fala. Cada idioma é acompanhado de seu código. Esses códigos de idioma são tags de identificação de idioma W3C ([ISO 639-3](#) para o nome do idioma e [ISO 3166](#) para o código do país).

Para obter tabelas detalhadas que mostrem os fonemas e visemas associados a cada idioma, escolha o link em cada idioma na tabela abaixo.

Linguagem	Código do idioma
Árabe (p. 20)	arb
Chinês, mandarim (p. 22)	cmn-CN
Dinamarquês (p. 25)	da-DK
Holandês (p. 28)	nl-NL
Inglês australiano (p. 31)	en-AU
Inglês britânico (p. 36)	en-GB
Inglês indiano (p. 33)	en-IN
Inglês norte-americano (p. 39)	en-US
Inglês de Gales (p. 41)	en-GB-WLS
Francês (p. 44)	fr-FR
Francês canadense (p. 46)	fr-CA
Hindi (p. 51)	hi-IN
Alemão (p. 48)	de-DE
Islandês (p. 53)	is-IS
Italiano (p. 56)	it-IT
Japonês (p. 58)	ja-JP
Coreano (p. 60)	ko-KR
Norueguês (p. 62)	nb-NO
Polonês (p. 65)	pl-PL
Português brasileiro (p. 69)	pt-BR
Português Europeu (p. 67)	pt-PT
Romeno (p. 71)	ro-RO
Russo (p. 73)	ru-RU

Linguagem	Código do idioma
Espanhol Europeu (p. 76)	es-ES
Espanhol, mexicano (p. 78)	es-MX
Espanhol, EUA (p. 79)	es-US
Sueco (p. 81)	sv-SE
Turco (p. 84)	tr-TR
Galês (p. 87)	cy-GB

Para obter mais informações, consulte [Tabelas de fonemas e visemas para os idiomas compatíveis \(p. 19\)](#).

Tabelas de fonemas e visemas para os idiomas compatíveis

As tabelas a seguir listam os fonemas para os idiomas compatíveis com o Amazon Polly, juntamente com exemplos e visemas correspondentes.

Tópicos

- [Árabe \(arb\) \(p. 20\)](#)
- [Chinês, mandarim \(cmn-CN\) \(p. 22\)](#)
- [Dinamarquês \(da-DK\) \(p. 25\)](#)
- [Holandês \(nl-NL\) \(p. 28\)](#)
- [Inglês australiano \(en-AU\) \(p. 31\)](#)
- [Inglês indiano \(en-IN\) \(p. 33\)](#)
- [Inglês britânico \(en-GB\) \(p. 36\)](#)
- [Inglês norte-americano \(en-US\) \(p. 39\)](#)
- [Inglês galês \(en-GB-WSL\) \(p. 41\)](#)
- [Francês \(fr-FR\) \(p. 44\)](#)
- [Francês canadense \(fr-CA\) \(p. 46\)](#)
- [Alemão \(de-DE\) \(p. 48\)](#)
- [Hindi \(hi-IN\) \(p. 51\)](#)
- [Islandês \(is-IS\) \(p. 53\)](#)
- [Italiano \(it-IT\) \(p. 56\)](#)
- [Japonês \(ja-JP\) \(p. 58\)](#)
- [Coreano \(ko-KR\) \(p. 60\)](#)
- [Norueguês \(nb-NO\) \(p. 62\)](#)
- [Polonês \(pl-PL\) \(p. 65\)](#)
- [Português \(pt-PT\) \(p. 67\)](#)
- [Português brasileiro \(pt-BR\) \(p. 69\)](#)
- [Romeno \(ro-RO\) \(p. 71\)](#)
- [Russo \(ru-RU\) \(p. 73\)](#)

- [Espanhol \(es-ES\) \(p. 76\)](#)
- [Espanhol mexicano \(es-MX\) \(p. 78\)](#)
- [Espanhol dos EUA \(es-US\) \(p. 79\)](#)
- [Sueco \(sv-SE\) \(p. 81\)](#)
- [Turco \(tr-TR\) \(p. 84\)](#)
- [Galês \(cy-GB\) \(p. 87\)](#)

Árabe (arb)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma árabe da Zeina, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
ʔ	ʔ	oclusiva glotal	أنا	
ʕ	ʔ\	fricativa faríngea sonora	عَمَر	k
b	b	oclusiva bilabial sonora	بَلَد	p
d	d	oclusiva alveolar sonora	دَارِي	t
dʕ	d_ʔ\	oclusiva alveolar sonora enfática	ضَوء	t
ɖʒ	dZ	africada pós-alveolar sonora	جَمِيل	S
ð	D	fricativa dental sonora	ذَلِكَ	T
ðʕ	D_ʔ\	fricativa dental sonora enfática	ظَلَام	T
f	f	fricativa labiodental surda	فَصْل	f
g	g	oclusiva velar sonora	إِنْجَلْتِرا	k
ɣ	G	fricativa velar sonora	عَرَب	k
h	h	fricativa glotal surda	هَذَا	k
j	j	aproximante palatal	يَمَشِي	i
k	k	oclusiva velar surda	كَلْب	k
l	l	aproximante lateral alveolar	لَاقِي	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
lʲ	l_G	aproximante lateral alveolar enfática	الله	t
m	m	nasal bilabial	ماذا	p
n	n	nasal alveolar	نور	t
p	p	oclusiva bilabial surda	حَبَس	p
q	q	oclusiva uvular surda	قَرِيب	k
r	r	vibrante alveolar	رَمَل	r
s	s	fricativa alveolar surda	سُؤَال	s
sʰ	s_?\\	fricativa alveolar surda enfática	صاحِب	s
ʃ	S	fricativa pós-alveolar surda	شُكِر	S
t	t	oclusiva alveolar surda	تَمَر	t
tʰ	t_?\\	oclusiva alveolar surda enfática	طالِب	t
θ	T	fricativa dental surda	ثَلَاث	T
v	v	fricativa labiodental sonora	فِي تَامِين	f
w	w	Aproximante labiovelar	وَلَد	u
x	x	fricativa velar surda	خَوْف	k
ħ	X\\	fricativa faríngea surda	حَوْلَ	k
z	z	fricativa alveolar sonora	زُور	s
Vogais				
a	a	vogal anterior aberta não arredondada	بَرَد	a
a:	a:	vogal anterior aberta longa não arredondada	دار	a
ɑʰ	A_?\\	vogal posterior aberta enfática não arredondada	طَبَل	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɑː	A_?ː	vogal posterior aberta longa enfática não arredondada	ظالم	a
u	u	vogal posterior fechada arredondada	شرب	u
uː	uː	vogal posterior fechada longa arredondada	سور	u
u̠	u_?\\	vogal posterior fechada enfática arredondada	بَدَّ	u
u̠ː	u_?ː	vogal posterior fechada longa enfática arredondada	طول	u
i	i	vogal anterior fechada não arredondada	بنت	i
iː	iː	vogal anterior fechada longa não arredondada	حَزِين	i
i̠	i_?\\	vogal anterior fechada enfática não arredondada	ضَدَّ	i
i̠ː	i_?ː	vogal anterior fechada longa enfática não arredondada	ماضي	i
p	p	vogal anterior semifechada não arredondada	ماركت	p
eː	pː	vogal anterior média fechada longa não arredondada	موديل	p
ɔ	U	vogal posterior média aberta arredondada	تكنولوجي	U
ɔː	Uː	vogal posterior média aberta longa arredondada	تلفزيون	U

Chinês, mandarim (cmn-CN)

A tabela a seguir lista o conjunto completo dos fonemas do Pinyin e o Alfabeto Fonético Internacional (AFI) para a voz do chinês mandarim que é compatível com o Amazon Polly. O Pinyin é o padrão internacional padrão para romanização do chinês padrão. O AFI e o X-SAMPA não são comumente usados, mas

estão disponíveis para suporte em inglês. Os símbolos do AFI e do X-SAMPA na tabela são apenas para referência e não devem ser usadas para transcrição do chinês. Os exemplos do Pinyin e os visemas correspondentes também são mostrados.

Para que o Amazon Polly use a pronúncia fonética win Pinyin, use `phoneme alphabet="x-amazon-phonetic standard used"` Tag.

Os exemplos a seguir mostram isso com cada padrão.

Pinyin:

```
<speack>
  ## <phoneme alphabet="x-amazon-pinyin" ph="bo2">#</phoneme>#
  ## <phoneme alphabet="x-amazon-pinyin" ph="bao2">#</phoneme>#
</speack>
```

AFI:

```
<speack>
  ## <phoneme alphabet="ipa" ph="p##k##n">pecan</phoneme>#
  ## <phoneme alphabet="ipa" ph="#pi.kæn">pecan</phoneme>#
</speack>
```

X-SAMPA:

```
<speack>
  ## <phoneme alphabet='x-sampa' ph='pI"kA:n'>pecan</phoneme>#
  ## <phoneme alphabet='x-sampa' ph='"pi.k{n'>pecan</phoneme>#
</speack>
```

Note

Amazon Polly aceita entrada em chinês mandarim codificada em UTF-8. No momento, o padrão de codificação GB 18030 não tem suporte no Amazon Polly.

Tabela de fonemas/visemas

Pinyin	AFI	X-SAMPA	Descrição	Exemplo do Pinyin	Visema
Consoantes					
f	f	f	fricativa labiodental surda	发, fa1	f
h	h	h	fricativa glotal surda	和, he2	k
g	k	k	oclusiva velar surda	古, gu3	k
k	k ^h	k_h	aspirated voiceless velar plosive	苦, ku3	k
l	l	l	aproximante lateral alveolar	拉, la1	t
m	m	m	nasal bilabial	骂, ma4	p
n	n	n	nasal alveolar	那, na4	t
ng	ŋ	N	nasal velar	正, zheng4	k
b	p	p	oclusiva bilabial surda	爸, ba4	p

Pinyin	AFI	X-SAMPA	Descrição	Exemplo do Pinyin	Visema
p	p ^h	p_h	oclusiva bilabial surda aspirada	怕, pa4	p
s	s	s	fricativa alveolar surda	四, si4	s
x	ɕ	s\	fricativa palato-alveolar surda	西, xi1	J
sh	ʃ	s`	fricativa retroflexa surda	是, shi4	S
d	t	t	oclusiva alveolar surda	打, da3	t
t	t ^h	t_h	oclusiva alveolar surda aspirada	他, ta1	t
zh	ʈʂ	t's`	africada retroflexa surda	之, zhi1	S
ch	ʈʂ ^h	t's`_h	africada retroflexa aspirada surda	吃, chi1	S
s	ʈs	ts	africada alveolar surda	字, zi4	s
j	ʈɕ	ts\	africada palato-alveolar surda	鸡, ji1	J
q	ʈɕ ^h	ts_h	africada palato-alveolar surda aspirada	七, qi1	J
c	ʈs ^h	ts_h	africada alveolar surda aspirada	次, ci4	s
w	w	w	Aproximante labiovelar	我, wo3	u
r	ʐ	z`	fricativa retroflexa sonora	日, ri4	S
sílabas coloridas "er" e "r"					
er	ə	@`	vogal média central colorida pelo r	二, er4	@
-r			sílaba colorida r	馅儿, xianr4	@
Vogais					
p	ɤ	7	vogal posterior semifechada não arredondada	恶, e4	p
p	ə	@	vogal central média	恩, en1	@
a	a	a	vogal anterior aberta não arredondada	安, an1	a
ai	aɪ	al	ditongo	爱, ai4	a
ao	aʊ	aU	ditongo	奥, ao4	a
ei	eɪ	p	ditongo	诶, ei4	p

Pinyin	AFI	X-SAMPA	Descrição	Exemplo do Pinyin	Visema
p	ɛ	E	vogal anterior média aberta não arredondada	姐, jie3	E
i	i	i	vogal anterior fechada não arredondada	鸡, ji1	i
ou	ou	oU	ditongo	欧, ou1	o
o	ɔ	U	vogal posterior média aberta arredondada	哦, o4	o
u	u	u	vogal posterior fechada arredondada	主, zhu3	u
yu	y	y	vogal anterior fechada arredondada	于, yu2	u
Marcas de tom e símbolos adicionais					
1			tom de alto nível	淤, yu1	
2			tom crescente	鱼, yu2	
3			tom baixo (descrescente-crescente)	语, yu3	
4			tom decrescente	育, yu4	
0			tom neutro	的, de0	
-	.	.	separação silábica	语音 yu3-yin1	

Dinamarquês (da-DK)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma dinamarquês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bat	p
d	d	oclusiva alveolar sonora	da	t
ð	D	fricativa dental sonora	mad, thriller	T
f	f	fricativa labiodental surda	fat	f

AFI	X-SAMPA	Descrição	Exemplo	Visema
g	g	oclusiva velar sonora	gat	k
h	h	fricativa glotal surda	hat	k
j	j	aproximante palatal	jo	i
k	k	oclusiva velar surda	kat	k
l	l	aproximante lateral alveolar	ladt	t
m	m	nasal bilabial	mat	p
n	n	nasal alveolar	nay	t
ŋ	N	nasal velar	lang	k
p	p	oclusiva bilabial surda	pande	p
r	r	vibrante alveolar	thriller, story	r
ʁ	R	fricativa uvular sonora	rat	k
s	s	fricativa alveolar surda	sat	s
t	t	oclusiva alveolar surda	tal	t
v	v	fricativa labiodental sonora	vat	f
w	w	aproximante labiovelar	hav, weekend	u
Vogais				
ø	2	vogal anterior semifechada arredondada	øst	o
ø:	2:	vogal anterior semifechada longa arredondada	øse	o
ɐ	6	vogal central quase aberta	mor	a
œ	9	vogal anterior média aberta arredondada	skøn, grønt	U
œ:	9:	vogal anterior média aberta longa arredondada	høne, gøre	U
ə	@	vogal central média	ane	@

AFI	X-SAMPA	Descrição	Exemplo	Visema
æ:	{:	vogal anterior quase aberta longa não arredondada	male	a
a	a	vogal anterior aberta não arredondada	man	a
æ	{	vogal anterior quase aberta não arredondada	adresse	a
ɑ	A	vogal posterior aberta não arredondada	lak, tak	a
ɑ:	A:	vogal posterior aberta longa não arredondada	rase	a
p	p	vogal anterior semifechada não arredondada	midt	p
p:	p:	vogal anterior média fechada longa não arredondada	mele	p
ɛ	E	vogal anterior média aberta não arredondada	mæt	E
ɛ:	E:	vogal anterior média aberta longa não arredondada	mæle	E
i	i	vogal anterior fechada não arredondada	mit	i
i:	i:	vogal anterior fechada longa não arredondada	mile	i
o	o	vogal posterior semifechada arredondada	foto	o
o:	o:	vogal posterior média fechada longa arredondada	mole	o
ɔ	U	vogal posterior média aberta arredondada	mund	U
ɔ:	U:	vogal posterior média aberta longa arredondada	måle	U

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɒ:	P:	vogal posterior aberta longa arredondada	morse	U
u	u	vogal posterior fechada arredondada	lusk	u
u:	u:	vogal posterior fechada longa arredondada	mule	u
ʌ	V	posterior média aberta não arredondada	kører	E
y	y	vogal anterior fechada arredondada	yt	u
y:	y:	vogal anterior fechada longa arredondada	hyle	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Holandês (nl-NL)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma holandês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bak	p
d	d	oclusiva alveolar sonora	dak	t
ɖ	dZ	africada pós-alveolar sonora	manager	S
f	f	fricativa labiodental surda	fel	f
g	g	oclusiva velar sonora	goal	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɣ	G	fricativa velar sonora	hoed	k
ɦ	h\	fricativa glotal sonora	hand	k
j	j	aproximante palatal	ja	i
k	k	oclusiva velar surda	kap	k
l	l	aproximante lateral alveolar	land	t
m	m	nasal bilabial	met	p
n	n	nasal alveolar	net	t
ŋ	N	nasal velar	bang	k
p	p	oclusiva bilabial surda	pak	p
r	r	vibrante alveolar	rand	r
s	s	fricativa alveolar surda	sein	s
ʃ	S	fricativa pós-alveolar surda	show	S
t	t	oclusiva alveolar surda	tak	t
v	v	fricativa labiodental sonora	vel	f
ʋ	v\	aproximante labiodental	wit	f
x	x	fricativa velar surda	toch	k
z	z	fricativa alveolar sonora	ziin	s
ʒ	Z	fricativa pós-alveolar sonora	bagage	S
Vogais				
ø:	2:	vogal anterior semifechada longa arredondada	neus	o
œy	9y	ditongo	buit	U
ə	@	vogal central média	de	@
a:	a:	vogal anterior aberta longa não arredondada	baad	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɑ:	A	vogal posterior aberta não arredondada	bad	a
p:	p:	vogal anterior média fechada longa não arredondada	beet	p
ɜ:	3:	vogal média central aberta longa não arredondada	barrière	E
ɛ	E	vogal anterior média aberta não arredondada	bed	E
ɛi	Ei	ditongo	beet	E
i	i	vogal anterior fechada não arredondada	vier	i
ɪ	l	vogal quase anterior quase fechada não arredondada	pit	i
o:	o:	vogal posterior média fechada longa arredondada	boot	o
ɔ	U	vogal posterior média aberta arredondada	pot	U
u	u	vogal posterior fechada arredondada	hoed	u
ʌu	Vu	ditongo	fout	E
y:	y:	vogal anterior fechada longa arredondada	fuut	u
ʏ	Y	vogal quase anterior quase fechada arredondada	hut	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Inglês australiano (en-AU)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma inglês australiano, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bed	p
d	d	oclusiva alveolar sonora	dig	t
ɖ	dZ	africada pós-alveolar sonora	jump	S
ð	D	fricativa dental sonora	then	T
f	f	fricativa labiodental surda	five	f
g	g	oclusiva velar sonora	game	k
h	h	fricativa glotal surda	house	k
j	j	aproximante palatal	yes	i
k	k	oclusiva velar surda	cat	k
l	l	aproximante lateral alveolar	lay	t
ɭ	l=	aproximante lateral alveolar silábica	battle	t
m	m	nasal bilabial	mouse	p
ṁ	m=	nasal bilabial silábica	anthem	p
n	n	nasal alveolar	nap	t
ṅ	n=	nasal alveolar silábica	nap	t
ŋ	N	nasal velar	thing	k
p	p	oclusiva bilabial surda	pin	p
r	r\	aproximante alveolar	red	r
s	s	fricativa alveolar surda	seem	s

AFI	X-SAMPA	Descrição	Exemplo	Visema
ʃ	S	fricativa pós-alveolar surda	ship	S
t	t	oclusiva alveolar surda	task	t
tʃ	tS	africada pós-alveolar surda	chart	S
θ	T	fricativa dental surda	thin	T
v	v	fricativa labiodental sonora	vest	f
w	w	aproximante labiovelar	west	u
z	z	fricativa alveolar sonora	zero	s
ʒ	Z	fricativa pós-alveolar sonora	vision	S
Vogais				
ə	@	vogal central média	arena	@
əʊ	@U	ditongo	goat	@
æ	{	vogal anterior quase aberta não arredondada	trap	a
aɪ	al	ditongo	price	a
aʊ	aU	ditongo	mouth	a
ɑ:	A:	vogal posterior aberta longa não arredondada	father	a
eɪ	el	ditongo	face	p
ɜ:	3:	vogal média central aberta longa não arredondada	nurse	E
ɛ	E	vogal anterior média aberta não arredondada	dress	E
ɛə	E@	ditongo	square	E
i:	i	vogal anterior fechada longa não arredondada	fleece	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɪ	I	vogal quase anterior quase fechada não arredondada	kit	i
ɪə	I@	ditongo	near	i
ɔ:	OI	vogal posterior média aberta longa arredondada	thought	U
ɔɪ	OI	Ditongo	choice	U
ɒ	Q	vogal posterior aberta arredondada	lot	U
u:	u:	vogal posterior fechada longa arredondada	goose	u
ʊ	U	vogal quase posterior quase fechada arredondada	foot	u
ʊə	U@	ditongo	cure	u
ʌ	V	Vogal posterior média aberta não arredondada	strut	E
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Inglês indiano (en-IN)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma inglês indiano, compatíveis com o Amazon Polly.

Para obter fonemas adicionais usados em conjunto com o inglês indiano, consulte [Hindi \(hi-IN\)](#) (p. 51).

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bed	p

AFI	X-SAMPA	Descrição	Exemplo	Visema
d	d	oclusiva alveolar sonora	dig	t
ɖ	dZ	africada pós-alveolar sonora	jump	S
ð	D	fricativa dental sonora	then	T
f	f	fricativa labiodental surda	five	f
g	g	oclusiva velar sonora	game	k
h	h	fricativa glotal surda	house	k
j	j	aproximante palatal	yes	i
k	k	oclusiva velar surda	cat	k
l	l	aproximante lateral alveolar	lay	t
ɭ	l=	aproximante lateral alveolar silábica	battle	t
m	m	nasal bilabial	mouse	p
ɱ	m=	nasal bilabial silábica	anthem	p
n	n	nasal alveolar	nap	t
ɳ	n=	nasal alveolar silábica	nap	t
ŋ	N	nasal velar	thing	k
p	p	oclusiva bilabial surda	pin	p
ɹ	r\	aproximante alveolar	red	r
s	s	fricativa alveolar surda	seem	s
ʃ	S	fricativa pós-alveolar surda	ship	S
t	t	oclusiva alveolar surda	task	t
ʈ	tS	africada pós-alveolar surda	chart	S
θ	T	fricativa dental surda	thin	T
v	v	fricativa labiodental sonora	vest	f

AFI	X-SAMPA	Descrição	Exemplo	Visema
w	w	aproximante labiovelar	west	u
z	z	fricativa alveolar sonora	zero	s
ʒ	Z	fricativa pós-alveolar sonora	vision	S
Vogais				
ə	@	vogal central média	arena	@
əʊ	@U	ditongo	goat	@
æ	{	vogal anterior quase aberta não arredondada	trap	a
aɪ	al	ditongo	price	a
aʊ	aU	ditongo	mouth	a
ɑ:	A:	vogal posterior aberta longa não arredondada	father	a
eɪ	el	ditongo	face	p
ɜ:	3:	vogal média central aberta longa não arredondada	nurse	E
ɛ	E	vogal anterior média aberta não arredondada	dress	E
ɛə	E@	ditongo	square	E
i:	i	vogal anterior fechada longa não arredondada	fleece	i
ɪ	l	vogal quase anterior quase fechada não arredondada	kit	i
ɪə	l@	ditongo	near	i
ɔ:	OI	vogal posterior média aberta longa arredondada	thought	U
ɔɪ	OI	Ditongo	choice	U
ɒ	Q	vogal posterior aberta arredondada	lot	U

AFI	X-SAMPA	Descrição	Exemplo	Visema
u:	u:	vogal posterior fechada longa arredondada	goose	u
ʊ	U	vogal quase posterior quase fechada arredondada	foot	u
ʊə	U@	ditongo	cure	u
ʌ	V	Vogal posterior média aberta não arredondada	strut	E
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Inglês britânico (en-GB)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma inglês britânico, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bed	p
d	d	oclusiva alveolar sonora	dig	t
ɖʒ	dZ	africada pós-alveolar sonora	jump	S
ð	D	fricativa dental sonora	then	T
f	f	fricativa labiodental surda	five	f
g	g	oclusiva velar sonora	game	k
h	h	fricativa glotal surda	house	k
j	j	aproximante palatal	yes	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
k	k	oclusiva velar surda	cat	k
l	l	aproximante lateral alveolar	lay	t
ɫ	l=	aproximante lateral alveolar silábica	battle	t
m	m	nasal bilabial	mouse	p
ɱ	m=	nasal bilabial silábica	anthem	p
n	n	nasal alveolar	nap	t
ɳ	n=	nasal alveolar silábica	botão	t
ŋ	N	nasal velar	thing	k
p	p	oclusiva bilabial surda	pin	p
ɹ	r\	aproximante alveolar	red	r
s	s	fricativa alveolar surda	seem	s
ʃ	S	fricativa pós-alveolar surda	ship	S
t	t	oclusiva alveolar surda	task	t
tʃ	tS	africada pós-alveolar surda	chart	S
θ	T	fricativa dental surda	thin	T
v	v	fricativa labiodental sonora	vest	f
w	w	aproximante labiovelar	west	u
z	z	fricativa alveolar sonora	zero	s
ʒ	Z	fricativa pós-alveolar sonora	vision	S
Vogais				
ə	@	vogal central média	arena	@
əʊ	@U	ditongo	goat	@
æ	{	vogal anterior quase aberta não arredondada	trap	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
aɪ	aɪ	ditongo	price	a
aʊ	aʊ	ditongo	mouth	a
ɑ:	A:	vogal posterior aberta longa não arredondada	father	a
eɪ	eɪ	ditongo	face	p
ɜ:	3:	vogal média central aberta longa não arredondada	nurse	E
ɛ	E	vogal anterior média aberta não arredondada	dress	E
ɛə	E@	ditongo	square	E
i:	i	vogal anterior fechada longa não arredondada	fleece	i
ɪ	ɪ	vogal quase anterior quase fechada não arredondada	kit	i
ɪə	ɪ@	ditongo	near	i
ɔ:	U:	vogal posterior média aberta longa arredondada	thought	U
ɔɪ	Oɪ	Ditongo	choice	U
ɒ	Q	vogal posterior aberta arredondada	lot	U
u:	u:	vogal posterior fechada longa arredondada	goose	u
ʊ	U	vogal quase posterior quase fechada arredondada	foot	u
ʊə	U@	ditongo	cure	u
ʌ	V	Vogal posterior média aberta não arredondada	strut	E
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	

AFI	X-SAMPA	Descrição	Exemplo	Visema
.	.	separação silábica	A.la.ba.ma	

Inglês norte-americano (en-US)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma inglês norte-americano, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bed	p
d	d	oclusiva alveolar sonora	dig	t
dʒ	dZ	africada pós-alveolar sonora	jump	S
ð	D	fricativa dental sonora	then	T
f	f	fricativa labiodental surda	five	f
g	g	oclusiva velar sonora	game	k
h	h	fricativa glotal surda	house	k
j	j	aproximante palatal	yes	i
k	k	oclusiva velar surda	cat	k
l	l	aproximante lateral alveolar	lay	t
m	m	nasal bilabial	mouse	p
n	n	nasal alveolar	nap	t
ŋ	N	nasal velar	thing	k
p	p	oclusiva bilabial surda	speak	p
r	r\	aproximante alveolar	red	r
s	s	fricativa alveolar surda	seem	s
ʃ	S	fricativa pós-alveolar surda	ship	S

AFI	X-SAMPA	Descrição	Exemplo	Visema
t	t	oclusiva alveolar surda	trap	t
tʃ	tS	africada pós-alveolar surda	chart	S
θ	T	fricativa dental surda	thin	T
v	v	fricativa labiodental sonora	vest	f
w	w	aproximante labiovelar	west	u
z	z	fricativa alveolar sonora	zero	s
ʒ	Z	fricativa pós-alveolar sonora	vision	S
Vogais				
ə	@	vogal central média	arena	@
ə˞	@˘	vogal média central colorida pelo r	reader	@
æ	{	vogal anterior quase aberta não arredondada	trap	a
aɪ	al	ditongo	price	a
aʊ	aU	ditongo	mouth	a
ɑ	A	vogal posterior aberta longa não arredondada	father	a
eɪ	el	ditongo	face	p
ɜ˞	3˘	vogal aberta média central não arredondada colorida pelo r	nurse	E
ɛ	E	vogal anterior média aberta não arredondada	dress	E
i	i	vogal anterior fechada longa não arredondada	fleece	i
ɪ	l	vogal quase anterior quase fechada não arredondada	kit	i
oʊ	oU	ditongo	goat	o

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɔ	U	vogal posterior média aberta longa arredondada	thought	U
ɔɪ	OI	ditongo	choice	U
u	u	vogal posterior fechada longa arredondada	goose	u
ʊ	U	vogal quase posterior quase fechada arredondada	foot	u
ʌ	V	vogal posterior média aberta não arredondada	strut	E
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Inglês galês (en-GB-WSL)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma inglês galês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bed	p
d	d	oclusiva alveolar sonora	dig	t
ɖʒ	dZ	africada pós-alveolar sonora	jump	S
ð	D	fricativa dental sonora	then	T
f	f	fricativa labiodental surda	five	f
g	g	oclusiva velar sonora	game	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
h	h	fricativa glotal surda	house	k
j	j	aproximante palatal	yes	i
k	k	oclusiva velar surda	cat	k
l	l	aproximante lateral alveolar	lay	t
ɭ	l=	aproximante lateral alveolar silábica	battle	t
m	m	nasal bilabial	mouse	p
ᵹ	m=	nasal bilabial silábica	anthem	p
n	n	nasal alveolar	nap	t
ɳ	n=	nasal alveolar silábica	nap	t
ŋ	N	nasal velar	thing	k
p	p	oclusiva bilabial surda	pin	p
ɹ	r\	aproximante alveolar	red	r
s	s	fricativa alveolar surda	seem	s
ʃ	S	fricativa pós-alveolar surda	ship	S
t	t	oclusiva alveolar surda	task	t
tʃ	tS	africada pós-alveolar surda	chart	S
θ	T	fricativa dental surda	thin	T
v	v	fricativa labiodental sonora	vest	f
w	w	aproximante labiovelar	west	u
z	z	fricativa alveolar sonora	zero	s
ʒ	Z	fricativa pós-alveolar sonora	vision	S
Vogais				
ə	@	vogal central média	arena	@
əʊ	@U	ditongo	goat	@

AFI	X-SAMPA	Descrição	Exemplo	Visema
æ	{	vogal anterior quase aberta não arredondada	trap	a
aɪ	al	ditongo	price	a
aʊ	aU	ditongo	mouth	a
ɑ:	A:	vogal posterior aberta longa não arredondada	father	a
eɪ	el	ditongo	face	p
ɜ:	3:	vogal média central aberta longa não arredondada	nurse	E
ɛ	E	vogal anterior média aberta não arredondada	dress	E
ɛə	E@	ditongo	square	E
i:	i	vogal anterior fechada longa não arredondada	fleece	i
ɪ	l	vogal quase anterior quase fechada não arredondada	kit	i
ɪə	l@	ditongo	near	i
ɔ:	OI	vogal posterior média aberta longa arredondada	thought	U
ɔɪ	OI	Ditongo	choice	U
ɒ	Q	vogal posterior aberta arredondada	lot	U
u:	u:	vogal posterior fechada longa arredondada	goose	u
ʊ	U	vogal quase posterior quase fechada arredondada	foot	u
ʊə	U@	ditongo	cure	u
ʌ	V	Vogal posterior média aberta não arredondada	strut	E
Símbolos adicionais				

AFI	X-SAMPA	Descrição	Exemplo	Visema
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Francês (fr-FR)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma francês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	boire	p
d	d	oclusiva alveolar sonora	madame	t
f	f	fricativa labiodental surda	femme	f
g	g	oclusiva velar sonora	grand	k
ɥ	H	aproximante labiopalatal	bruit	u
j	j	aproximante palatal	meilleur	i
k	k	oclusiva velar surda	quatre	k
l	l	aproximante lateral alveolar	malade	t
m	m	nasal bilabial	maison	p
n	n	nasal alveolar	astronome	t
ɲ	J	nasal palatal	baigner	J
ŋ	N	nasal velar	parking	k
p	p	oclusiva bilabial surda	pomme	p
ʁ	R	fricativa uvular sonora	amoureux	k
s	s	fricativa alveolar surda	santé	s

AFI	X-SAMPA	Descrição	Exemplo	Visema
ʃ	S	fricativa pós-alveolar surda	chat	S
t	t	oclusiva alveolar surda	téléphone	t
v	v	fricativa labiodental sonora	vrai	f
w	w	aproximante labiovelar	soir	u
z	z	fricativa alveolar sonora	raison	s
ʒ	Z	fricativa pós-alveolar sonora	aubergine	S
Vogais				
ø	2	vogal anterior semifechada arredondada	deux	o
œ	9	vogal anterior média aberta arredondada	neuf	U
œ̃	9~	vogal anterior média aberta nasal arredondada	brun	U
ə	@	vogal central média	je	@
a	a	vogal anterior aberta não arredondada	table	a
ã	A~	vogal posterior aberta nasal não arredondada	camembert	a
p	p	vogal anterior semifechada não arredondada	marché	p
ɛ	E	vogal anterior média aberta não arredondada	neige	E
ɛ̃	E~	vogal anterior média aberta nasal não arredondada	sapin	E
i	i	vogal anterior fechada não arredondada	mille	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
o	o	vogal posterior semifechada arredondada	hôpital	o
ɔ	U	vogal posterior média aberta arredondada	homme	U
õ	U~	vogal posterior média aberta nasal arredondada	bon	U
u	u	vogal posterior fechada arredondada	sous	u
y	y	vogal anterior fechada arredondada	dur	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Francês canadense (fr-CA)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma francês canadense, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	boire	p
d	d	oclusiva alveolar sonora	madame	t
f	f	fricativa labiodental surda	femme	f
g	g	oclusiva velar sonora	grand	k
ɥ	H	aproximante labiopalatal	bruit	u
j	j	aproximante palatal	meilleur	i
k	k	oclusiva velar surda	quatre	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
l	l	aproximante lateral alveolar	malade	t
m	m	nasal bilabial	maison	p
n	n	nasal alveolar	astronome	t
ɲ	J	nasal palatal	baigner	J
ŋ	N	nasal velar	parking	k
p	p	oclusiva bilabial surda	pomme	p
ɸ	R	fricativa uvular sonora	amoureux	k
s	s	fricativa alveolar surda	santé	s
ʃ	S	fricativa pós-alveolar surda	chat	S
t	t	oclusiva alveolar surda	téléphone	t
v	v	fricativa labiodental sonora	vrai	f
w	w	aproximante labiovelar	soir	u
z	z	fricativa alveolar sonora	raison	s
ʒ	Z	fricativa pós-alveolar sonora	aubergine	S
Vogais				
ø	2	vogal anterior semifechada arredondada	deux	o
œ	9	vogal anterior média aberta arredondada	neuf	U
œ̃	9~	vogal anterior média aberta nasal arredondada	brun	U
ə	@	vogal central média	je	@
a	a	vogal anterior aberta não arredondada	table	a
ɑ̃	A~	vogal posterior aberta nasal não arredondada	camembert	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
p	p	vogal anterior semifechada não arredondada	marché	p
ɛ	E	vogal anterior média aberta não arredondada	neige	E
ẽ	E~	vogal anterior média aberta nasal não arredondada	sapin	E
i	i	vogal anterior fechada não arredondada	mille	i
o	o	vogal posterior semifechada arredondada	hôpital	o
ɔ	U	vogal posterior média aberta arredondada	homme	U
õ	U~	vogal posterior média aberta nasal arredondada	bon	U
u	u	vogal posterior fechada arredondada	sous	u
y	y	vogal anterior fechada arredondada	dur	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Alemão (de-DE)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma alemão, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
ʔ	ʔ	oclusiva glotal		

AFI	X-SAMPA	Descrição	Exemplo	Visema
b	b	oclusiva bilabial sonora	Bier	p
d	d	oclusiva alveolar sonora	Dach	t
ç	C	fricativa palatal surda	ich	k
ḋ	dZ	africada pós-alveolar sonora	Dschungel	S
f	f	Fricativa labiodental surda	Vogel	f
g	g	Oclusiva velar sonora	Gabel	k
h	h	Fricativa glotal surda	Haus	k
j	j	Fricativa glotal surda	jemand	i
k	k	Oclusiva velar surda	Kleid	k
l	l	Aproximante lateral alveolar	Loch	t
m	m	Nasal bilabial	Milch	p
n	n	Nasal alveolar	Natur	t
ŋ	N	Nasal velar	klingen	k
p	p	Oclusiva bilabial surda	Park	p
pf	pf	Africada labiodental surda	Apfel	
r	R	Vibrante uvular	Regen	
s	s	fricativa alveolar surda	Messer	s
ʃ	S	Fricativa pós-alveolar surda	Fischer	S
t	t	Oclusiva alveolar surda	Topf	T
ts	Ts	Africada alveolar surda	Zahl	
tʃ	tS	Africada pós-alveolar surda	deutsch	S
v	v	Fricativa labiodental sonora	Wasser	f
x	x	Fricativa velar surda	kochen	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
z	z	Fricativa alveolar sonora	Consulte	s
ʒ	Z	Fricativa pós-alveolar sonora	Orange	S
Vogais				
ø:	2:	vogal anterior semifechada longa arredondada	böse	o
ɐ	6	vogal central quase aberta	besser	a
ɜ̃	6_^	vogal central quase aberta não silábica	Klar	a
œ	9	vogal anterior média aberta arredondada	können	U
ə	@	vogal central média	Rede	@
a	a	vogal anterior aberta não arredondada	Salz	a
a:	a:	vogal anterior aberta longa não arredondada	Sahne	a
aɪ	al	ditongo	nein	a
au	aU	ditongo	Augen	a
ã	A~	vogal posterior aberta nasal não arredondada	Restaurant	a
p:	p:	vogal anterior média fechada longa não arredondada	Rede	p
ɛ	E	vogal anterior média aberta não arredondada	Keller	E
ẽ	E~	vogal anterior média aberta nasal não arredondada	Terrain	E
i:	i:	vogal anterior fechada longa não arredondada	Lied	i
ɪ	l	vogal quase anterior quase fechada não arredondada	bitte	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
o:	o:	vogal posterior média fechada longa arredondada	Kohl	o
ɔ	U	vogal posterior média aberta arredondada	Koffer	U
õ	U~	vogal posterior média aberta nasal arredondada	Annonce	U
ɔʏ	OY	ditongo	neu	U
u:	u:	vogal posterior fechada longa arredondada	Bruder	u
ʊ	U	vogal quase posterior quase fechada arredondada	Wunder	u
y:	y:	vogal anterior fechada longa arredondada	kühl	u
ʏ	Y	vogal quase anterior quase fechada arredondada	Küche	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Hindi (hi-IN)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e o tipo de som do fonema para vozes do idioma hindi, compatíveis com o Amazon Polly.

Para obter fonemas adicionais usados em conjunto com o hindi, consulte [Inglês indiano \(en-IN\) \(p. 33\)](#).

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo
Consoantes			
p ^h	p_h	oclusiva bilabial aspirada surda	फूल (phool)

AFI	X-SAMPA	Descrição	Exemplo
b ^h	b_h	oclusiva bilabial aspirada sonora	भारी (bhaari)
t̪	t_d	oclusiva dental surda	तापमान (taapmaan)
t̪ ^h	t_d_h	oclusiva dental aspirada surda	थोड़ा (thoda)
d̪	d_d	oclusiva dental sonora	दिल्ली (dilli)
d̪ ^h	d_d_h	oclusiva dental aspirada sonora	धोबी (dhobi)
t̠	t̠	oclusiva retroflexa surda	कटोरा (katora)
t̠ ^h	t̠_h	oclusiva retroflexa aspirada surda	ठंड (thand)
d̠	d̠	oclusiva retroflexa sonora	डर (darr)
d̠ ^h	d̠_h	oclusiva retroflexa aspirada sonora	ढाल (dhal)
tʃ ^h	tS_h	africada palatal aspirada surda	छाल (chaal)
dʒ ^h	dZ_h	africada palatal aspirada sonora	झाल (jhaal)
k ^h	k_h	oclusiva velar aspirada surda	खान (khan)
g ^h	g_h	oclusiva velar aspirada sonora	घान (ghaan)
ŋ	n̠	nasal retroflexa	क्षण (kshan)
r	ɽ	flape alveolar	राम (ram)
ɽ	r̠	flape retroflexa simples	बड़ा (bada)
ɽ ^h	r̠_h	flape retroflexa aspirada sonora	बढ़ी (barhi)
ʋ	v\	aproximadamente bilabial	वसूल (wasool)
Vogais			
ə	@_o	vogal central média	अच्छा (achhaa)
ẽ	@~	vogal central média nasal	हँसना (hansnaa)
a	A_o	vogal anterior aberta não arredondada	आग (aag)
ã	A~	vogal anterior aberta não arredondada nasal	घड़ियाँ (ghariyaan)
ɪ	I_o	vogal quase anterior quase fechada não arredondada	इक्कीस (ikkees)

AFI	X-SAMPA	Descrição	Exemplo
ĩ	l~	vogal quase anterior quase fechada não arredondada nasal	संचिई (sinchai)
i	i_o	vogal anterior fechada não arredondada	बिल्ली (billee)
ĩ	i~	vogal anterior fechada não arredondada nasal	नही (nahin)
ʊ	U_o	vogal quase posterior quase fechada arredondada	उलूल (ullu)
ũ	U~	vogal quase posterior quase fechada arredondada nasal	मुँह (munh)
u	u_o	vogal posterior fechada arredondada	फूल (phool)
ũ	u~	vogal posterior fechada arredondada nasal	ऊँट (oont)
ɔ	O_o	vogal posterior média aberta arredondada	कौन (kaun)
õ	U~	vogal posterior média aberta arredondada nasal	भौ (bhaun)
o	o	vogal posterior semifechada arredondada	सोना (sona)
õ	o~	vogal posterior semifechada arredondada nasal	क्यो (kyon)
ɛ	E_o	vogal anterior média aberta não arredondada	पैसा (paisa)
ẽ	E~	vogal anterior média aberta não arredondada nasal	मैं (main)
p	p	vogal anterior semifechada não arredondada	एक (ek)
ṗ	p~	vogal anterior semifechada não arredondada nasal	कतिबें (kitabein)

Islandês (is-IS)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma islandês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	grasbakkanum	0
c	c	oclusiva palatal surda	pakkin	k
c ^h	c_h	oclusiva palatal surda aspirada	anarkistai	k
ç	C	fricativa palatal surda	héðan	k
d	d	oclusiva alveolar sonora	bónði	t
ð	D	fricativa dental sonora	borð	T
f	f	fricativa labiodental surda	duft	f
g	g	oclusiva velar sonora	holgóma	k
ɣ	G	fricativa velar sonora	hugur	k
h	h	fricativa glotal surda	heili	k
j	j	aproximante palatal	jökull	i
k ^h	k_h	aspirated voiceless velar plosive	ósköpunum	k
l	l	aproximante lateral alveolar	gólf	t
ɭ	l_0	aproximante lateral alveolar surda	fólk	t
m	m	nasal bilabial	september	p
m̥	m_0	nasal bilabial surda	kompá	p
n	n	nasal alveolar	númer	t
n̥	n_0	nasal alveolar surda	pöntun	t
ɲ	J	nasal palatal	pælingar	J
ŋ	N	nasal velar	söngvarann	k
ŋ̥	N_0	nasal velar surda	frænka	k
p ^h	p_h	oclusiva bilabial surda aspirada	afplánun	p
r	r	vibrante alveolar	afskrifta	r

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɾ	r_0	vibrante alveolar surda	andvörpum	r
s	s	fricativa alveolar surda	baðhús	s
tʰ	t_h	oclusiva alveolar surda aspirada	tanki	t
θ	T	fricativa dental surda	þeldökki	T
v	v	fricativa labiodental sonora	silfur	f
w	w	aproximante labiovelar		u
x	x	fricativa velar surda	samfélags	k
Vogais				
œ	9	vogal anterior média aberta arredondada	þröskuldinum	U
œ:	9:	vogal anterior média aberta longa arredondada	tvö	U
a	a	vogal anterior aberta não arredondada	nefna	a
a:	a:	vogal anterior aberta longa não arredondada	fara	a
au	au	ditongo	átta	a
au:	au:	ditongo	átján	a
ɛ	E	vogal anterior média aberta não arredondada	kennari	E
ɛ:	E:	vogal anterior média aberta longa não arredondada	dreka	E
i	i	vogal anterior fechada não arredondada	Gúliver	i
i:	i:	vogal anterior fechada longa não arredondada	þír	i
ɪ	l	vogal quase anterior quase fechada não arredondada	samspil	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
i:	I:	vogal quase anterior quase fechada longa não arredondada	stig	i
ɔ	U	vogal posterior média aberta arredondada	regndropar	U
ɔ:	U:	vogal posterior média aberta longa arredondada	ullarbolur	U
ou	Ou	ditongo	tólf	U
ou:	Ou:	ditongo	fjórir	U
u	u	vogal posterior fechada arredondada	stúlkán	u
u:	u:	vogal posterior fechada longa arredondada	frú	u
ɣ	Y	vogal quase anterior quase fechada arredondada	tíu	u
ɣ:	Y	vogal quase anterior quase fechada longa arredondada	gruninn	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Italiano (it-IT)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma italiano, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bacca	p
d	d	oclusiva alveolar sonora	dama	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
dz	dz	africada alveolar sonora	zero	s
dʒ	dZ	africada pós-alveolar sonora	giro	S
f	f	fricativa labiodental surda	famiglia	f
g	g	oclusiva velar sonora	gatto	k
h	h	fricativa glotal surda	horror	k
j	j	aproximante palatal	dieci	i
k	k	oclusiva velar surda	campo	k
l	l	aproximante lateral alveolar	lido	t
ʎ	L	aproximante lateral palatal	aglio	J
m	m	nasal bilabial	mille	p
n	n	nasal alveolar	nove	t
ɲ	J	nasal palatal	lasagne	J
p	p	oclusiva bilabial surda	pizza	p
r	r	vibrante alveolar	risata	r
s	s	fricativa alveolar surda	sei	s
ʃ	S	fricativa pós-alveolar surda	scienza	S
t	t	oclusiva alveolar surda	tavola	t
ts	ts	africada alveolar surda	forza	s
tʃ	tS	africada pós-alveolar surda	cielo	S
v	v	fricativa labiodental sonora	venti	f
w	w	aproximante labiovelar	quattro	u
z	z	fricativa alveolar sonora	bisogno	s

AFI	X-SAMPA	Descrição	Exemplo	Visema
ʒ	Z	fricativa pós-alveolar sonora	bijou	S
Vogais				
a	a	vogal anterior aberta não arredondada	arco	a
ɐ	p	vogal anterior semifechada não arredondada	tre	p
ɛ	E	vogal anterior média aberta não arredondada	ettaro	E
i	i	vogal anterior fechada não arredondada	impero	i
o	o	vogal posterior semifechada arredondada	cento	o
ɔ	U	vogal posterior média aberta arredondada	otto	U
u	u	vogal posterior fechada arredondada	uno	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Japonês (ja-JP)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma japonês, compatíveis com o Amazon Polly.

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	練習, renshuu	t
ʔ	ʔ	oclusiva glotal	あつ, atsu'	
b	b	oclusiva bilabial sonora	舞踊, buyou	p

AFI	X-SAMPA	Descrição	Exemplo	Visema
β	B	fricativa bilabial sonora	ヴェンテージ, vinteeji	B
c	c	oclusiva palatal surda	ききょう, kikyō	k
ç	C	fricativa palatal surda	人, hito	k
d	d	oclusiva alveolar sonora	濁点, dakuten	t
dz	dz\	africada palato-alveolar sonora	純, jun	J
g	g	oclusiva velar sonora	ご飯, gohan	k
h	h	fricativa glotal surda	本, hon	k
j	j	aproximante palatal	屋根, yane	i
ɟ	ɟ\	oclusiva palatal sonora	行儀, gyōgi	J
k	k	oclusiva velar surda	漢字, kanji	k
ɭ	ɭ\	flape lateral alveolar	釣り, tsuri	r
ɭj	ɭj\	flape lateral alveolar, aproximante palatal	流行, ryūkō	r
m	m	nasal bilabial	飯, meshi	p
n	n	nasal alveolar	猫, neko	t
ɲ	J	nasal palatal	日本, nippon	J
ŋ	N\	nasal uvular	缶, kan	k
p	p	oclusiva bilabial surda	パン, pan	p
ɸ	p\	fricativa bilabial surda	福, huku	f
s	s	fricativa alveolar surda	層, sou	s
ʃ	s\	fricativa palato-alveolar surda	書簡, shokan	J
t	t	oclusiva alveolar surda	手紙, tegami	t
ts	ts	africada alveolar surda	釣り, tsuri	s
tʃ	ts\	africada palato-alveolar surda	吉, kichi	J
w	w	aproximante labiovelar	電話, denwa	u

AFI	X-SAMPA	Descrição	Exemplo	Visema
z	z	fricativa alveolar sonora	座敷, zashiki	s
Vogais				
ä:	a_"	vogal central aberta longa não arredondada	羽蟻, haari	a
ä	a_	vogal aberta central não arredondada	仮名, kana	a
e:	e:_o	vogal anterior média longa não arredondada	学生, gakusei	@
p	e_o	vogal anterior média não arredondada	歴, reki	@
i	i	vogal anterior fechada não arredondada	気, ki	i
i:	i:	vogal anterior fechada longa não arredondada	詩歌, shiika	i
u	M	vogal posterior fechada não arredondada	運, un	i
u:	M:	vogal posterior fechada longa não arredondada	宗教, shuukyou	i
o:	o:_o	vogal anterior média longa arredondada	購読, koodoku	o
o	o_o	vogal anterior média arredondada	読者, dokusha	o

Coreano (ko-KR)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma coreano, compatíveis com o Amazon Polly.

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
k	k	oclusiva velar surda	강, [g]ang	k
k#	k_t	oclusiva velar surda forte	깨, [kk]e	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
n	n	nasal alveolar	남, [n]am	t
t	t	oclusiva alveolar surda	도, [d]o	t
t#	t_t	oclusiva alveolar surda forte	때, [tt]e	t
r	4	flape alveolar	사랑, sa[r]ang	t
l	l	aproximante lateral alveolar	돌, do[l]	t
m	m	nasal bilabial	무, [m]u	p
p	p	oclusiva bilabial surda	봄, [b]om	p
p#	p_t	oclusiva bilabial surda forte	빨, [pp]eol	p
s	s	fricativa alveolar surda	새, [s]e	s
s#	s_t	fricativa alveolar surda forte	씨, [ss]i	s
ŋ	N	nasal velar	방, ba[ng]	k
tɕ	ts\	africada palato-alveolar surda	조, [j]o	J
tɕ#	ts_t	africada palato-alveolar surda forte	찌, [jj]i	J
tɕ ^h	ts_h	africada palato-alveolar surda aspirada	차, [ch]a	J
k ^h	k_h	aspirated voiceless velar plosive	코, [k]o	k
t ^h	t_h	oclusiva alveolar surda aspirada	통, [t]ong	t
p ^h	p_h	oclusiva bilabial surda aspirada	패, [p]e	p
h	h	fricativa glotal surda	힘, [h]im	k
j	j	aproximante palatal	양, [y]ang	i
w	w	aproximante labiovelar	왕, [w]ang	u
ɯ	M\	aproximadamente velar>	의, [wj]i	i
Vogais				

AFI	X-SAMPA	Descrição	Exemplo	Visema
a	a	vogal anterior aberta não arredondada	밥, b[a]b	a
ʌ	V	vogal posterior média aberta não arredondada	정, j[eo]ng	E
ɛ	E	vogal anterior média aberta não arredondada	배, b[e]	E
o	o	vogal posterior semifechada arredondada	노, n[o]	o
u	u	vogal posterior fechada arredondada	둘, d[u]l	u
ʊ	M	vogal posterior fechada não arredondada	은, [eu]n	i
i	i	vogal anterior fechada não arredondada	김, k[i]m	i

Norueguês (nb-NO)

O gráfico a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI) e os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala), bem como os visemas correspondentes, conforme compatíveis com o Amazon Polly para vozes do idioma norueguês.

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	prøv	t
b	b	oclusiva bilabial sonora	labb	p
ç	C	fricativa palatal surda	kino	k
d	d	oclusiva alveolar sonora	ladd	t
ɖ	d'	oclusiva retroflexa sonora	verdi	t
f	f	fricativa labiodental surda	fot	f
g	g	oclusiva velar sonora	tagg	k
h	h	fricativa glotal surda	ha	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
j	j	aproximante palatal	gi	i
k	k	oclusiva velar surda	takk	k
l	l	aproximante lateral alveolar	fall, ball	t
ɭ	l̥	aproximante lateral retroflexa	ærlig	t
m	m	nasal bilabial	lam	p
n	n	nasal alveolar	vann	t
ɳ	n̥	nasal retroflexa	garn	t
ŋ	ŋ	nasal velar	sang	k
p	p	oclusiva bilabial surda	hopp	p
s	s	fricativa alveolar surda	lass	s
ʂ	s̥	fricativa retroflexa surda	års	S
ʃ	S	fricativa pós-alveolar surda	skyt	S
t	t	oclusiva alveolar surda	lat	t
ɖ	t̥	oclusiva retroflexa surda	hardt	t
ʋ	v̥	aproximante labiodental	vin	f
w	w	aproximante labiovelar	will	x
Vogais				
ø:	2:	vogal anterior semifechada longa arredondada	søt	o
œ	9	vogal anterior média aberta arredondada	søtt	U
ə	@	vogal central média	ape	@
æ:	{:	vogal anterior quase aberta longa não arredondada	vær	a
ʊ	}	vogal fechada central arredondada	lund	u

AFI	X-SAMPA	Descrição	Exemplo	Visema
ʊ:	}:	vogal fechada central longa arredondada	lun	u
æ	{	vogal anterior quase aberta não arredondada	vært	a
ɑ	A	vogal posterior aberta não arredondada	hatt	a
ɑ:	A:	vogal posterior aberta longa não arredondada	hat	a
p:	p:	vogal anterior média fechada longa não arredondada	sen	p
ɛ	E	vogal anterior média aberta não arredondada	send	E
i:	i:	vogal anterior fechada longa não arredondada	vin	i
ɪ	l	vogal quase anterior quase fechada não arredondada	vind	i
o:	o:	vogal posterior média fechada longa arredondada	våt	o
ɔ	U	vogal posterior média aberta arredondada	vått	U
u:	u:	vogal posterior fechada longa arredondada	bok	u
ʊ	U	vogal quase posterior quase fechada arredondada	bukk	u
y:	y:	vogal anterior fechada longa arredondada	lyn	u
ʏ	Y	vogal quase anterior quase fechada arredondada	lynne	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	

AFI	X-SAMPA	Descrição	Exemplo	Visema
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Polonês (pl-PL)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma polonês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bobas, belka	p
d	d	oclusiva alveolar sonora	dar, do	t
ɖ	dz	africada alveolar sonora	dzwon, widzowie	s
ɗ	dz\	africada palato-alveolar sonora	dźwięk	J
ɖ̞	dz`	africada retroflexa sonora	dżem, dżungla	S
f	f	fricativa labiodental surda	furtka, film	f
g	g	oclusiva velar sonora	gazeta, waga	k
h	h	fricativa glotal surda	chleb, handel	k
j	j	aproximante palatal	jak, maja	i
k	k	oclusiva velar surda	kura, marek	k
l	l	aproximante lateral alveolar	lipa, alicja	t
m	m	nasal bilabial	matka, molo	p
n	n	nasal alveolar	norka	t
ɲ	J	nasal palatal	koń, toruń	J
p	p	oclusiva bilabial surda	pora, stop	p
r	r	vibrante alveolar	rok, park	r

AFI	X-SAMPA	Descrição	Exemplo	Visema
s	s	fricativa alveolar surda	sum, pas	s
ɕ	s\	fricativa palato-alveolar surda	śruba, śnieg	J
ʂ	s`	fricativa retroflexa surda	szum, masz	S
t	t	oclusiva alveolar surda	tok, stół	t
ʈs	ts	africada alveolar surda	car, co	s
ʈɕ	ts\	africada palato-alveolar surda	ćma, mieć	J
ʈʂ	ts`	africada retroflexa surda	czas, raczej	S
v	v	fricativa labiodental sonora	worek, mewa	f
w	w	aproximante labiovelar	łaska, mało	u
z	z	fricativa alveolar sonora	zero	s
ʐ	z\	fricativa palato-alveolar sonora	żrebię, bieliźnie	J
ʑ	z`	fricativa retroflexa sonora	żar, żona	S
Vogais				
a	a	vogal anterior aberta não arredondada	ja	a
ɛ	E	vogal anterior média aberta não arredondada	echo	E
ẽ	E~	vogal anterior média aberta nasal não arredondada	węże	E
i	i	vogal anterior fechada não arredondada	ile	i
ɔ	U	vogal posterior média aberta arredondada	oczy	U

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɔ̃	U~	vogal posterior média aberta nasal arredondada	wąż	U
u	u	vogal posterior fechada arredondada	uczta	u
ɨ	1	vogal central fechada não arredondada	byk	i
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Português (pt-PT)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma português, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	pira	t
b	b	oclusiva bilabial sonora	dato	p
d	d	oclusiva alveolar sonora	dato	t
f	f	fricativa labiodental surda	facto	f
g	g	oclusiva velar sonora	gato	k
j	j	aproximante palatal	paraguay	i
k	k	oclusiva velar surda	cacto	k
l	l	aproximante lateral alveolar	galo	t
ʎ	L	aproximante lateral palatal	galho	J
m	m	nasal bilabial	mato	p
n	n	nasal alveolar	nato	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɲ	J	nasal palatal	pinha	J
p	p	oclusiva bilabial surda	pato	p
ʀ	R\	vibrante uvular	barroso	k
s	s	fricativa alveolar surda	saca	s
ʃ	S	fricativa pós-alveolar surda	chato	S
t	t	oclusiva alveolar surda	tacto	t
v	v	fricativa labiodental sonora	vaca	f
w	w	aproximante labiovelar	mau	u
z	z	fricativa alveolar sonora	zaca	s
ʒ	Z	fricativa pós-alveolar sonora	jacto	S
Vogais				
a	a	vogal anterior aberta não arredondada	parto	a
ã	a~	vogal anterior aberta nasal não arredondada	pega	a
ɐ	p	vogal anterior semifechada não arredondada	pega	p
õ	p~	vogal anterior nasal semifechada não arredondada	movem	p
ɛ	E	vogal anterior média aberta não arredondada	café	E
i	i	vogal anterior fechada não arredondada	lingueta	i
ĩ	i~	vogal anterior nasal fechada não arredondada	cinto	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
o	o	vogal posterior semifechada arredondada	poder	o
õ	o~	vogal posterior nasal semifechada arredondada	compra	o
ɔ	U	vogal posterior média aberta arredondada	cotó	U
u	u	vogal posterior fechada arredondada	fui	u
ũ	u~	vogal posterior fechada nasal arredondada	sunto	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Português brasileiro (pt-BR)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma português brasileiro, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	pira	t
b	b	oclusiva bilabial sonora	bato	p
d	d	oclusiva alveolar sonora	dato	t
ɖ	dZ	africada pós-alveolar sonora	idade	S
f	f	fricativa labiodental surda	facto	f
g	g	oclusiva velar sonora	gato	k
j	j	aproximante palatal	paraguay	i

AFI	X-SAMPA	Descrição	Exemplo	Visema
k	k	oclusiva velar surda	cacto	k
l	l	aproximante lateral alveolar	galo	t
ʎ	L	aproximante lateral palatal	galho	J
m	m	nasal bilabial	mato	p
n	n	nasal alveolar	nato	t
ɲ	J	nasal palatal	pinha	J
p	p	oclusiva bilabial surda	pato	p
s	s	fricativa alveolar surda	saca	s
ʃ	S	fricativa pós-alveolar surda	chato	S
t	t	oclusiva alveolar surda	tacto	t
tʃ	tS	africada pós-alveolar surda	noite	S
v	v	fricativa labiodental sonora	vaca	f
w	w	aproximante labiovelar	mau	u
x	X	fricativa uvular surda	carro	k
z	z	fricativa alveolar sonora	zaca	s
ʒ	Z	fricativa pós-alveolar sonora	jacto	S
Vogais				
a	a	vogal anterior aberta não arredondada	parto	a
ã	a~	vogal anterior aberta nasal não arredondada	pensamos	a
ɐ	p	vogal anterior semifechada não arredondada	pega	p
ẽ	p~	vogal anterior nasal semifechada não arredondada	movem	p

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɛ	E	vogal anterior média aberta não arredondada	café	E
i	i	vogal anterior fechada não arredondada	lingueta	i
ĩ	i~	vogal anterior nasal fechada não arredondada	cinto	i
o	o	vogal posterior semifechada arredondada	poder	o
õ	o~	vogal posterior nasal semifechada arredondada	compra	o
ɔ	U	vogal posterior média aberta arredondada	cotó	U
u	u	vogal posterior fechada arredondada	fui	u
ũ	u~	vogal posterior fechada nasal arredondada	sunto	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Romeno (ro-RO)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma romeno, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bubă	p
d	d	oclusiva alveolar sonora	după	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɖʒ	dZ	africada pós-alveolar sonora	george	S
f	f	fricativa labiodental surda	afacere	f
g	g	oclusiva velar sonora	agri#	k
h	h	fricativa glotal surda	harpă	k
j	j	aproximante palatal	baie	i
k	k	oclusiva velar surda	co#	k
l	l	aproximante lateral alveolar	lampa	t
m	m	nasal bilabial	mama	p
n	n	nasal alveolar	nor	t
p	p	oclusiva bilabial surda	pilă	p
r	r	vibrante alveolar	rampă	r
s	s	fricativa alveolar surda	soare	s
ʃ	S	fricativa pós-alveolar surda	ma#ină	S
t	t	oclusiva alveolar surda	tata	t
ts	ts	africada alveolar surda	#ară	s
tʃ	tS	africada pós-alveolar surda	ceai	S
v	v	fricativa labiodental sonora	via#ă	f
w	w	aproximante labiovelar	beau	u
z	z	fricativa alveolar sonora	mozol	s
ʒ	Z	fricativa pós-alveolar sonora	joacă	S
Vogais				
ə	@	vogal central média	babă	@
a	a	vogal anterior aberta não arredondada	casa	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
p	p	vogal anterior semifechada não arredondada	elan	p
ɐ	p_^	vogal anterior média fechada não silábica não arredondada	beau	p
i	i	vogal anterior fechada não arredondada	mie	i
o	o	vogal posterior média fechada arredondada	orã	o
oa	o_^a	ditongo	oare	o
u	u	vogal posterior fechada arredondada	unde	u
ɨ	1	vogal central fechada não arredondada	România	i
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Russo (ru-RU)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma russo, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	борт	p
bʲ	b'	oclusiva bilabial sonora palatalizada	бюро	p
d	d	oclusiva alveolar sonora	дом	t
dʲ	d'	oclusiva alveolar sonora palatalizada	дядя	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
f	f	fricativa labiodental surda	флаг	f
fʲ	fʲ	fricativa labiodental surda palatalizada	февраль	f
g	g	oclusiva velar sonora	нога	k
gʲ	gʲ	oclusiva velar sonora palatalizada	герой	k
j	j	aproximante palatal	дизайн, ящик	i
k	k	oclusiva velar surda	кот	k
kʲ	kʲ	oclusiva velar surda palatalizada	кино	k
l	l	aproximante lateral alveolar	лампа	t
lʲ	lʲ	aproximante lateral alveolar palatalizada	лес	t
m	m	nasal bilabial	мама	p
mʲ	mʲ	nasal bilabial palatalizada	мяч	p
n	n	nasal alveolar	нос	t
nʲ	nʲ	nasal alveolar palatalizada	няня	t
p	p	oclusiva bilabial surda	папа	p
pʲ	pʲ	oclusiva bilabial surda palatalizada	перо	p
r	r	vibrante alveolar	роза	r
rʲ	rʲ	vibrante alveolar palatalizada	рюмка	r
s	s	fricativa alveolar surda	сыр	s
sʲ	sʲ	fricativa alveolar surda palatalizada	сердце, русь	s
ɸ:	sɹ:	fricativa palato-alveolar surda longa	щека	J
ʂ	sʲ	fricativa retroflexa surda	шум	S
t	t	oclusiva alveolar surda	точка	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
tʲ	tʰ	oclusiva alveolar surda palatalizada	тѣтя	t
ts	ts	africada alveolar surda	царь	s
tɕ	ts\	africada palato-alveolar surda	час	J
v	v	fricativa labiodental sonora	вор	f
vʲ	vʰ	fricativa labiodental sonora palatalizada	верфь	f
x	x	fricativa velar surda	хор	k
xʲ	xʰ	fricativa velar surda palatalizada	химия	k
z	z	fricativa alveolar sonora	зуб	s
zʲ	zʰ	fricativa alveolar sonora palatalizada	зима	s
ʐ:	z\:	fricativa palato-alveolar sonora longa	уезжать	J
ʐ̞	z`	fricativa retroflexa sonora	жена	S
Vogais				
ə	@	vogal central média	канарейка	@
a	a	vogal anterior aberta não arredondada	два, яблоко	a
ɐ	ɐ	vogal anterior semifechada não arredondada	печь	p
ɛ	E	vogal anterior média aberta não arredondada	это	E
i	i	vogal anterior fechada não arredondada	один, четыре	i
o	o	vogal posterior semifechada arredondada	кот	o
u	u	vogal posterior fechada arredondada	муж, вьюга	u

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɪ	1	vogal central fechada não arredondada	мышь	i

Espanhol (es-ES)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma espanhol, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	pero, bravo, amor, eterno	t
b	b	oclusiva bilabial sonora	bestia	p
β	B	fricativa bilabial sonora	bebé	B
d	d	oclusiva alveolar sonora	cuando	t
ð	D	fricativa dental sonora	arder	T
f	f	fricativa labiodental surda	fase, café	f
g	g	oclusiva velar sonora	gato, lengua, guerra	k
ɣ	G	fricativa velar sonora	trigo, Argos	k
j	j	aproximante palatal	hacia, tierra, radio, viuda	i
ɟ	j\	fricativa palatal sonora	enhielar, sayo, inyectado, desyerba	J
k	k	oclusiva velar surda	caña, laca, quisimos	k
l	l	aproximante lateral alveolar	lino, calor, principal	t
ʎ	L	aproximante lateral palatal	llave, pollo	J
m	m	nasal bilabial	madre, comer, anfibio	p
n	n	nasal alveolar	nido, anillo, sin	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
ɲ	J	nasal palatal	cabaña, ñoquis	J
ŋ	N	nasal velar	cinco, venga	k
p	p	oclusiva bilabial surda	pozo, topo	p
r	r	vibrante alveolar	perro, enrachado	r
s	s	fricativa alveolar surda	saco, casa, puertas	s
t	t	oclusiva alveolar surda	tamiz, átomo	t
ɟ	tS	africada pós-alveolar surda	chubasco	S
θ	T	fricativa dental surda	cereza, zorro, lacero, paz	T
w	w	aproximante labiovelar	fuego, fuimos, cuota, cuadro	u
x	x	fricativa velar surda	jamón, general, suje, reloj	k
z	z	fricativa alveolar sonora	rasgo, mismo	s
Vogais				
a	a	vogal anterior aberta não arredondada	tanque	a
ɐ	p	vogal anterior semifechada não arredondada	peso	p
i	i	vogal anterior fechada não arredondada	cinco	i
o	o	vogal posterior semifechada arredondada	bosque	o
u	u	vogal anterior semifechada não arredondada	publicar	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Espanhol mexicano (es-MX)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma espanhol mexicano, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	pero, bravo, amor, eterno	t
b	b	oclusiva bilabial sonora	bestia	p
β	B	fricativa bilabial sonora	bebé	B
d	d	oclusiva alveolar sonora	cuando	t
ð	D	fricativa dental sonora	arder	T
f	f	fricativa labiodental surda	fase, café	f
g	g	oclusiva velar sonora	gato, lengua, guerra	k
ɣ	G	fricativa velar sonora	trigo, Argos	k
j	j	aproximante palatal	hacia, tierra, radio, viuda	i
ɟ	j\	fricativa palatal sonora	enhielar, sayo, inyectado, desyerba	J
k	k	oclusiva velar surda	caña, laca, quisimos	k
l	l	Aproximante alveolar lateral	lino, calor, principal	t
m	m	nasal bilabial	madre, comer, anfibio	p
n	n	nasal alveolar	nido, anillo, sin	t
ɲ	J	nasal palatal	cabaña, ñoquis	J
ŋ	N	nasal velar	angosto, increíble	k
p	p	oclusiva bilabial surda	pozo, topo	p
r	r	vibrante alveolar	perro, enrachado	r

AFI	X-SAMPA	Descrição	Exemplo	Visema
s	s	fricativa alveolar surda	saco, casa, puertas	s
ʃ	S	fricativa pós-alveolar surda	show, flash	S
t	t	oclusiva alveolar surda	tamiz, átomo	t
tʃ	tS	africada pós-alveolar surda	chubasco	S
w	w	aproximante labiovelar	fuego, fuimos, cuota, cuadro	u
x	x	fricativa velar surda	jamón, general, peaje, reloj	k
z	z	fricativa alveolar sonora	rasgo, mismo	s
Vogais				
a	a	vogal central aberta não arredondada	tanque	a
p	p	vogal anterior semifechada não arredondada	peso	p
i	i	vogal anterior fechada não arredondada	cinco	i
o	o	vogal posterior semifechada arredondada	bosque	o
u	u	vogal posterior fechada arredondada	publicar	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Espanhol dos EUA (es-US)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma espanhol norte-americano, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	pero, bravo, amor, eterno	t
b	b	oclusiva bilabial sonora	bestia	p
β	B	fricativa bilabial sonora	bebé	B
d	d	oclusiva alveolar sonora	cuando	t
ð	D	fricativa dental sonora	arder	T
f	f	fricativa labiodental surda	fase, café	f
g	g	oclusiva velar sonora	gato, lengua, guerra	k
ɣ	G	fricativa velar sonora	trigo, Argos	k
j	j	aproximante palatal	hacia, tierra, radio, viuda	i
ɟ	j\	fricativa palatal sonora	enhielar, sayo, inyectado, desyerba	J
k	k	oclusiva velar surda	caña, laca, quisimos	k
l	l	Aproximante alveolar lateral	lino, calor, principal	t
m	m	nasal bilabial	madre, comer, anfibio	p
n	n	nasal alveolar	nido, anillo, sin	t
ɲ	J	nasal palatal	cabaña, ñoquis	J
ŋ	N	nasal velar	angosto, increíble	k
p	p	oclusiva bilabial surda	pozo, topo	p
r	r	vibrante alveolar	perro, enrachado	r
s	s	fricativa alveolar surda	saco, casa, puertas	s
ʃ	S	fricativa pós-alveolar surda	show, flash	S
t	t	oclusiva alveolar surda	tamiz, átomo	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
tʃ	tS	africada pós-alveolar surda	chubasco	S
w	w	aproximante labiovelar	fuego, fuimos, cuota, cuadro	u
x	x	fricativa velar surda	jamón, general, peaje, reloj	k
z	z	fricativa alveolar sonora	rasgo, mismo	s
Vogais				
a	a	vogal central aberta não arredondada	tanque	a
p	p	vogal anterior semifechada não arredondada	peso	p
i	i	vogal anterior fechada não arredondada	cinco	i
o	o	vogal posterior semifechada arredondada	bosque	o
u	u	vogal posterior fechada arredondada	publicar	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Sueco (sv-SE)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma sueco, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	bil	p

AFI	X-SAMPA	Descrição	Exemplo	Visema
d	d	oclusiva alveolar sonora	dal	t
ɖ	d`	oclusiva retroflexa sonora	bord	t
f	f	fricativa labiodental surda	fil	f
g	g	oclusiva velar sonora	gås	k
h	h	fricativa glotal surda	hal	k
j	j	aproximante palatal	jag	i
k	k	oclusiva velar surda	kal	k
l	l	aproximante lateral alveolar	lös	t
ɭ	l`	aproximante lateral retroflexa	härlig	t
m	m	nasal bilabial	mil	p
n	n	nasal alveolar	nålar	t
ɳ	n`	nasal retroflexa	barn	t
ŋ	ŋ	nasal velar	ring	k
p	p	oclusiva bilabial surda	pil	p
r	r	vibrante alveolar	ris	r
s	s	fricativa alveolar surda	sil	s
ʃ	s\	fricativa palato-alveolar surda	tjock	J
ʂ	s`	fricativa retroflexa surda	fors, schlager	S
t	t	oclusiva alveolar surda	tal	t
ɖ	t`	oclusiva retroflexa surda	hjort	t
v	v	fricativa labiodental sonora	vår	f
w	w	aproximante labiovelar	aula, airways	u
ɣ	x\	fricativa palatal velar surda	sjuk	k

AFI	X-SAMPA	Descrição	Exemplo	Visema
Vogais				
ø	2	vogal anterior semifechada arredondada	föll, förr	o
ø	2:	vogal anterior semifechada longa arredondada	föl, nöt, för	o
ə	8	vogal média central fechada arredondada	buss, full	o
ə	@	vogal central média	pojken	@
ʊ:	}:	vogal fechada central longa arredondada	hus, ful	u
a	a	vogal anterior aberta não arredondada	hall, matt	a
æ	{	vogal anterior quase aberta não arredondada	herr	a
ɑ:	A:	vogal posterior aberta longa não arredondada	hal, mat	a
p:	p:	vogal anterior média fechada longa não arredondada	vet, hel	p
ɛ	E	vogal anterior média aberta não arredondada	vett, rätt, hetta, häll	E
ɛ:	E:	vogal anterior média aberta longa não arredondada	säl, häl, här	E:
i:	i:	vogal anterior fechada longa não arredondada	vit, sil	i:
ɪ	l	vogal quase anterior quase fechada não arredondada	vitt, sill	i
o:	o:	vogal posterior média fechada longa arredondada	håll, mål	o
ɔ	U	vogal posterior média aberta arredondada	håll, moll	U

AFI	X-SAMPA	Descrição	Exemplo	Visema
u:	u:	vogal posterior fechada longa arredondada	sol, bot	u
ʊ	U	vogal quase posterior quase fechada arredondada	bott	u
y	y	vogal anterior fechada arredondada	bytt	u
y:	y:	vogal anterior fechada longa arredondada	syl, syl	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Turco (tr-TR)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma turco, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
r	4	flape alveolar	durum	t
ɾ̥	4_0_r	flape alveolar fricado surdo	bir	t
ɾ	4_r	flape alveolar fricado	raf	t
b	b	oclusiva bilabial sonora	raf	p
c	c	oclusiva palatal surda	kedi	k
d	d	oclusiva alveolar sonora	dede	t
ɖʒ	dZ	africada pós-alveolar sonora	cam	S
f	f	fricativa labiodental surda	fare	f

AFI	X-SAMPA	Descrição	Exemplo	Visema
g	g	oclusiva velar sonora	galibi	k
h	h	fricativa glotal surda	hasta	k
j	j	aproximante palatal	yat	i
ɟ	ɟ	oclusiva palatal sonora	genç	J
k	k	oclusiva velar surda	akıl	k
l	l	aproximante lateral alveolar	lale	t
ɮ	ʁ	aproximante lateral alveolar velarizada	labirent	t
m	m	nasal bilabial	maaş	p
n	n	nasal alveolar	anı	t
p	p	oclusiva bilabial surda	ip	p
s	s	fricativa alveolar surda	ses	s
ʃ	ʃ	fricativa pós-alveolar surda	aşı	S
t	t	oclusiva alveolar surda	ütü	t
tʃ	tʃ	africada pós-alveolar surda	çaba	S
v	v	fricativa labiodental sonora	ekvator, kahveci, akvaryum, isveçli, teşviki, cetvel	f
z	z	fricativa alveolar sonora	ver	s
ʒ	ʒ	fricativa pós-alveolar sonora	azık	S
Vogais				
ø	2	vogal anterior semifechada arredondada	göl	0
œ	9	vogal anterior média aberta arredondada	banliyö	U
a	a	vogal anterior aberta não arredondada	kal	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
a:	a:	vogal anterior aberta longa não arredondada	davacı	a
æ	{	vogal anterior quase aberta não arredondada	özlem, güvenlik, gürel, somersault	a
p	p	vogal anterior semifechada não arredondada	keçi	p
ɛ	E	vogal anterior média aberta não arredondada	dede	E
i	i	vogal anterior fechada não arredondada	bir	i
i:	i:	vogal anterior fechada longa não arredondada	izah	i
ɪ	l	vogal quase anterior quase fechada não arredondada	keçi	i
ʊ	M	vogal posterior fechada não arredondada	kıl	i
o	o	vogal posterior semifechada arredondada	kol	o
o:	o:	vogal posterior média fechada longa arredondada	dolar	o
u	u	vogal posterior fechada arredondada	durum	u
u:	u:	vogal posterior fechada longa arredondada	ruhum	u
ʊ	U	vogal quase posterior quase fechada arredondada	dolu	u
y	y	vogal anterior fechada arredondada	güvenlik	u
ɣ	Y	vogal quase anterior quase fechada arredondada	aşı	u

AFI	X-SAMPA	Descrição	Exemplo	Visema
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

Galês (cy-GB)

A tabela a seguir lista o conjunto completo dos fonemas do Alfabeto Fonético Internacional (AFI), os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala) e os visemas correspondentes para vozes do idioma galês, compatíveis com o Amazon Polly.

Tabela de fonemas/visemas

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	oclusiva bilabial sonora	baban	p
d	d	oclusiva alveolar sonora	deg	t
ɖ	dZ	africada pós-alveolar sonora	garej	S
ð	D	fricativa dental sonora	deuddeg	T
f	f	fricativa labiodental surda	ffacs	f
g	g	oclusiva velar sonora	gadael	k
h	h	fricativa glotal surda	haearn	k
j	j	aproximante palatal	astudio	i
k	k	oclusiva velar surda	cant	k
l	l	aproximante lateral alveolar	lan	t
ɭ	K	fricativa lateral alveolar surda	llan	t
m	m	nasal bilabial	mae	p
ɱ	m_0	nasal bilabial surda	ymhen	p
n	n	nasal alveolar	naw	t
ɳ	n_0	nasal alveolar surda	anhawster	t

AFI	X-SAMPA	Descrição	Exemplo	Visema
ŋ	N	nasal velar	argyfwng	k
ɳ	N_0	nasal velar surda	anghenion	k
p	p	oclusiva bilabial surda	pump	p
r	r	vibrante alveolar	rhoi	r
ɾ	r_0	vibrante alveolar surda	garw	r
s	s	fricativa alveolar surda	saith	s
ʃ	S	fricativa pós-alveolar surda	siawns	S
t	t	oclusiva alveolar surda	tegan	t
tʃ	tS	africada pós-alveolar surda	cytsain	S
θ	T	fricativa dental surda	aberth	T
v	v	fricativa labiodental sonora	prawf	f
w	w	aproximante labiovelar	rhagweld	u
x	X	fricativa uvular surda	chwech	k
z	z	fricativa alveolar sonora	aids	s
ʒ	Z	fricativa pós-alveolar sonora	rouge	S
Vogais				
ə	@	vogal central média	ychwanega	@
a	a	vogal anterior aberta não arredondada	acen	a
ai	ai	ditongo	dau	a
au	au	ditongo	awdur	a
ɑ:	A:	vogal posterior aberta longa não arredondada	mab	a
ɑ:i	A:1	ditongo	aelod	a

AFI	X-SAMPA	Descrição	Exemplo	Visema
p:	p:	vogal anterior média fechada longa não arredondada	peth	p
ɛ	E	vogal anterior média aberta não arredondada	pedwar	E
ɛi	Ei	ditongo	beic	E
i:	i:	vogal anterior fechada longa não arredondada	tri	i
ɪ	l	vogal quase anterior quase fechada não arredondada	miliwn	i
iu	1u	ditongo	unigryw	i
o:	o:	vogal posterior média fechada longa arredondada	oddi	o
ɔ	U	vogal posterior média aberta arredondada	oddieithr	U
ɔi	Oi	ditongo	troi	U
ɔu	Ou	ditongo	rownd	U
u:	u:	vogal posterior fechada longa arredondada	cwch	u
ʊ	U	vogal quase posterior quase fechada arredondada	acwstig	u
ʊi	Ui	ditongo	wyth	u
Símbolos adicionais				
'	"	tonicidade primária	Alabama	
,	%	tonicidade secundária	Alabama	
.	.	separação silábica	A.la.ba.ma	

TTS neural

O Amazon Polly tem um TTS neurais (NTTS) Sistema que pode produzir vozes de qualidade ainda mais alta do que as vozes padrão. O sistema NTTS produz as vozes de texto em fala mais naturais e semelhantes possíveis à voz humana.

As vozes do TTS padrão usam síntese concatenativa. Esse método une (concatena) os fonemas da fala gravada, produzindo uma fala sintetizada muito natural. No entanto, as variações inevitáveis na fala e as técnicas usadas para segmentar as formas de onda limitam a qualidade da fala.

O sistema de TTS neural da Amazon Polly não usa a síntese concatenativa padrão para produzir fala. Ele tem duas partes:

- Uma rede neural que converte uma sequência de fonemas — as unidades mais básicas de um idioma — em uma sequência de espectrogramas, que são snapshots dos níveis de energia em diferentes faixas de frequência
- Um vocoder, que converte os espectrogramas em um sinal de áudio contínuo.

O primeiro componente do sistema TTS neural é um modelo de sequência a sequência. Esse modelo não cria seus resultados exclusivamente a partir da entrada correspondente, mas também considera como a sequência dos elementos da entrada funcionam em conjunto. O modelo escolhe os espectrogramas que ele gera para que suas faixas de frequência enfatizem os recursos acústicos que o cérebro humano usa ao processar a fala.

A saída desse modelo passa para um vocoder neural. Isso converte os espectrogramas em formas de onda de fala. Quando treinado nos grandes conjuntos de dados usados para criar sistemas de síntese concatenativa de uso geral, essa abordagem de sequência a sequência produzirá vozes de melhor qualidade e mais naturais.

Com exceção das vozes de Kevin (inglês dos EUA) e Olivia (inglês australiano), todas as vozes NTTS também têm uma contraparte criada usando o método TTS padrão. As vozes de Kevin e Olivia são suportadas apenas pelo Amazon Polly ao usar o NTTS. Ao usar uma voz somente NTTS, o parâmetro do mecanismo TTS deve ser definido como `neura1`, seja usando o console ou a API do.

Tópicos

- [Compatibilidade de recursos e regiões \(p. 90\)](#)
- [O mecanismo de voz \(p. 91\)](#)
- [Vozes neurais \(p. 93\)](#)
- [Estilo de locutor do NTTS \(p. 94\)](#)

Compatibilidade de recursos e regiões

Vozes neurais não estão disponíveis em todos os AWS Regiões, nem oferecem suporte a todos os recursos do Amazon Polly.

As vozes neurais têm suporte nas seguintes regiões:

- Leste dos EUA (Norte da Virgínia): `us-east-1`
- Oeste dos EUA (Oregon): `us-west-2`

- Ásia-Pacífico (Tóquio): ap-northeast-1
- Ásia-Pacífico (Seul): ap-northeast-2
- Ásia-Pacífico (Singapura): ap-southeast-1
- Ásia-Pacífico (Sydney): ap-southeast-2
- Canadá (Central): ca-central-1
- Europa (Frankfurt): eu-central-1
- Europa (Irlanda): eu-west-1
- Europa (Londres): eu-west-2
- AWSGovCloud (US-West): us-gov-west-1

Os endpoints e protocolos para essas regiões são idênticos aos usados para vozes padrão. Para obter mais informações, consulte [Endpoints e cotas do Amazon Polly](#).

Os seguintes recursos são compatíveis com vozes neurais:

- Operações de síntese de fala em tempo real e assíncrona.
- Estilo de locutor. Para obter mais informações sobre os estilos de locutor, consulte [Estilo de locutor do NTTTS \(p. 94\)](#).
- Todas as marcações de fala.
- Muitas (mas não todas) tags SSML são compatíveis com o Amazon Polly. Para obter mais informações sobre tags SSML compatíveis com NTTTS, consulte [Tags SSML compatíveis \(p. 109\)](#).

Assim como ocorre com vozes padrão, é possível escolher entre várias taxas de amostragem para otimizar a largura de banda e a qualidade do áudio para seu aplicativo. As taxas de amostragem válidas para vozes padrão e neurais são 8 kHz, 16 kHz, 22 kHz ou 24 kHz. O padrão para vozes padrão é 22 kHz. O padrão para vozes neurais é 24 kHz. O Amazon Polly oferece suporte aos formatos de fluxo de áudio MP3, OGG (Vorbis) e PCM bruto.

O mecanismo de voz

O Amazon Polly permite que você use voz neural ou padrão com o `engine` propriedade. Ele tem dois valores possíveis: `Standard` (Padrão) ou `Neural`, indicando se deseja usar uma voz padrão ou neural. `Standard` (Padrão) é o valor padrão.

Important

Se você não estiver em uma das quatro regiões nas quais o NTTTS está disponível, somente o mecanismo de voz padrão será exibido no console. Se o mecanismo neural não for exibido, verifique a sua região. Para obter mais informações sobre as regiões nas quais o NTTTS pode ser usado, consulte [Compatibilidade de recursos e regiões \(p. 90\)](#).

Ao usar uma voz somente NTTTS, o parâmetro do mecanismo TTS deve ser definido como `neural`, seja usando o console ou a API do.

Escolher o mecanismo de voz (Console)

Como escolher um mecanismo de voz (console)

1. Abra o console do Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Na página Text-to-Speech (Texto em fala), em Engine (Mecanismo), escolha `Standard` (Padrão) ou `Neural`.

Text-to-Speech [Info](#)

[Save to S3](#) [Download](#) [Listen](#)

Engine [Info](#)

☒ **Neural**
Produces the most natural and human-like speech possible.

☐ **Standard**
Produces natural-sounding speech.

Language [Info](#)

English, US ▼

Voice [Info](#)

Joanna, Female ▼

Input text [Info](#) ☐ **SSML** [Info](#)

Hi! My name is Joanna. I will read any text you type here.

58 characters used

[Restore default text](#) [Clear text](#)

Se você escolher Neural, somente vozes neurais estarão disponíveis e as vozes somente padrão serão desabilitadas.

Escolher o mecanismo de voz (CLI)

Como escolher um mecanismo de voz (CLI)

O parâmetro `engine` é opcional, com dois valores possíveis: `standard` ou `Neural`. Use essa propriedade ao criar uma operação `SynthesisTask`.

Por exemplo, é possível usar o código a seguir para executar `start-speech-synthesis-task` AWS CLI na região Oeste-2 dos EUA (Oregon)

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (`\`) no final de cada linha por um circunflexo (`^`) e use aspas completas (`"`) ao redor do texto de entrada por aspas simples (`'`) para tags internas.

```
aws polly start-speech-synthesis-task \
  --engine neural \
  --region us-west-2 \
  --endpoint-url "https://polly.us-west-1.amazonaws.com/" \
  --output-format mp3 \
  --output-s3-bucket-name your-bucket-name \
  --output-s3-key-prefix optional/prefix/path/file \
  --voice-id Joanna \
  --text file://text_file.txt
```

Isso resultará em uma resposta semelhante a esta:

```
"SynthesisTask":
{
  "CreationTime": [...],
  "Engine": "neural",
  "OutputFormat": "mp3",
  "OutputUri": "https://s3.us-west-1.amazonaws.com/your-bucket-name/optional/prefix/
path/file.<task_id>.mp3",
  "TextType": "text",
  "RequestCharacters": [...],
  "TaskStatus": "scheduled",
  "TaskId": [task_id],
  "VoiceId": "Joanna"
}
```

Vozes neurais

Vozes neurais estão disponíveis em inglês dos EUA (en-US), inglês britânico (en-GB), inglês australiano (en-AU), português brasileiro (pt-BR), francês canadense (FR-CA), alemão (de-DE), coreano (Ko-KR) e espanhol dos EUA (es-US). A tabela a seguir lista as vozes.

Linguagem	Nome/ID	Sexo
Inglês australiano (en-AU)	Olivia**	Feminino
Inglês britânico (en-GB)	AMY*	Feminino
	Emma	Feminino
	Brian	Masculino
Inglês dos EUA (en-US)	Ivy	Feminino (criança)
	Joanna*	Feminino
	Kendra	Feminino

Linguagem	Nome/ID	Sexo
	Kimberly	Feminino
	Salli	Feminino
	Joey	Masculino
	Justin	Masculino (criança)
	Kevin**	Masculino (criança)
	Matthew*	Masculino
Francês canadense (fr-CA)	Gabrielle	Feminino
Alemão (de-DE)	Vicki	Feminino
Coreano (ko-KR)	Seoyeon	Feminino
Português (Brasil) (pt-BR)	Camila	Feminino
Espanhol dos EUA (es-US)	Lupe*	Feminino

*As vozes de Amy, Joanna, Lupe e Matthew podem ser usadas com o estilo de locutor. Para obter mais informações, consulte [Estilo de locutor do NTTS \(p. 94\)](#).

**As vozes de Gabrielle, Kevin e Olivia estão disponíveis somente em NTTS e não como vozes padrão.

Estilo de locutor do NTTS

As pessoas usam estilos de fala diferentes, dependendo do contexto. Conversação informal, por exemplo, parece muito diferente de um noticiário de TV ou de rádio. Devido à forma como vozes padrão são feitas, elas não podem produzir diferentes estilos de fala. No entanto, vozes neurais podem. Eles podem ser treinados para um estilo de fala específico, com as variações e a ênfase em certas partes da fala inerentes a esse estilo.

Além das vozes neurais padrão, Amazon Polly fornece um estilo de locutor que usa o sistema neural para gerar fala no estilo de um apresentador de TV ou de rádio. O estilo de locutor está disponível com as vozes de Matthew e Joanna em inglês dos EUA (en-US), a voz de Lupe em espanhol dos EUA (es-US) e a voz de Amy em inglês britânico (en-GB).

Para usar o estilo de locutor, primeiro escolha o mecanismo neural e, depois, use a sintaxe descrita nas etapas a seguir no texto de entrada.

Note

Para usar qualquer estilo de fala neural, você deve usar um dos `AWSRegions` que suportam vozes neurais. Essa opção não está disponível em todas as regiões. Para obter mais informações, consulte [Compatibilidade de recursos e regiões \(p. 90\)](#).

Para aplicar o estilo de locutor (console)

1. Abra o console do Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Verifique se você está usando um `AWSRegion` onde vozes neurais são compatíveis.
3. Na página Text-to-Speech (Texto em fala), em Engine (Mecanismo), escolha Neural.
4. Escolha o idioma e a voz que deseja usar.

Apenas Matthew e Joanna para inglês dos EUA (en-US), Lupe para espanhol dos EUA (es-US) e Amy para inglês britânico (en-GB) estão disponíveis na voz de locutor.

5. Ativar oSSML.
6. Adicione o texto de entrada à solicitação de conversão de texto em fala usando a sintaxe SSML no estilo de locutor.

```
<amazon:domain name="news">text</amazon:domain>
```

Por exemplo, você pode usar a tag de locutor da seguinte forma:

```
<speak>
<amazon:domain name="news">
From the Tuesday, April 16th, 1912 edition of The Guardian newspaper:

The maiden voyage of the White Star liner Titanic, the largest ship ever launched
ended in disaster.

The Titanic started her trip from Southampton for New York on Wednesday. Late on
Sunday night she struck an iceberg off the Grand Banks of Newfoundland. By
wireless telegraphy she sent out signals of distress, and several liners were
near enough to catch and respond to the call.
</amazon:domain>
</speak>
```

7. Selecione Escuta.

Como aplicar o estilo de locutor (CLI)

1. Na solicitação de API, inclua o parâmetro do mecanismo com o valor `neural`:

```
--engine neural
```

2. Adicione o texto de entrada à solicitação de API usando a sintaxe SSML no estilo de locutor.

```
<amazon:domain name="news">text</amazon:domain>
```

Por exemplo, você pode usar a tag de locutor da seguinte forma:

```
<speak>
<amazon:domain name="news">
From the Tuesday, April 16th, 1912 edition of The Guardian newspaper:

The maiden voyage of the White Star liner Titanic, the largest ship ever launched
ended in disaster.

The Titanic started her trip from Southampton for New York on Wednesday. Late on
Sunday night she struck an iceberg off the Grand Banks of Newfoundland. By
wireless telegraphy she sent out signals of distress, and several liners were
near enough to catch and respond to the call.
</amazon:domain>
</speak>
```

Para obter mais informações sobre SSML, consulte [Tags SSML compatíveis \(p. 109\)](#).

Marcas de fala

Marcas de fala são metadados que descrevem a fala que você sintetiza, como onde uma frase ou palavra começa e termina no fluxo de áudio. Quando você solicita marcas de fala para seu texto, o Amazon Polly retorna esses metadados em vez de fala sintetizada. Ao usar marcas de fala em conjunto com um fluxo de áudio de fala sintetizada, você pode fornecer às suas aplicações uma experiência visual melhor.

Por exemplo, combinar os metadados com o fluxo de áudio do seu texto pode permitir que você sincronize a fala com uma animação facial (sincronia labial) ou destaque palavras escritas à medida que são faladas.

As marcas de fala estão disponíveis ao usar formatos neurais ou padrão de conversão de texto em fala.

Tópicos

- [Tipos de marca de fala](#) (p. 96)
- [Uso de marcas de fala](#) (p. 97)
- [Solicitação de marcas de fala \(Console\)](#) (p. 100)

Tipos de marca de fala

Solicite marcas de fala usando a opção [SpeechMarkTypes](#) para os comandos [SynthesizeSpeech](#) ou [StartSpeechSynthesisTask](#). Você especifica os elementos de metadados que deseja retornar do texto de entrada. Você pode solicitar até quatro tipos de metadados, mas deve especificar pelo menos um por solicitação. Nenhuma saída de áudio é gerada com a solicitação.

NoAWS CLI, por exemplo:

```
--speech-mark-types='["sentence", "word", "viseme", "ssml"]'
```

O Amazon Polly gera marcas de fala usando os seguintes elementos:

- sentence – Indica um elemento de frase no texto de entrada.
- word – Indica um elemento de palavra no texto.
- viseme— Descreve os movimentos faciais e orais correspondentes a cada fonema falado. Para obter mais informações, consulte [Visemas e Amazon Polly](#) (p. 96).
- SSML— Descreve um <mark> elemento do texto de entrada SSML. Para obter mais informações, consulte [Geração de fala a partir de documentos SSML](#) (p. 102).

Visemas e Amazon Polly

Um visema representa a posição da face e da boca ao dizer uma palavra. Ele é o equivalente visual a um fonema, que é a unidade acústica básica da qual uma palavra é formada. Visemas são os blocos básicos da construção visual da fala.

Cada idioma tem um conjunto de visemas correspondentes a seus fonemas específicos. Em um idioma, cada fonema tem um visema correspondente que representa a forma que a boca deve tomar para formar o som. No entanto, nem todos os visemas podem ser mapeados a um fonema em particular, pois diversos fonemas parecem ser iguais quando falados, ainda que soem diferentes. Por exemplo, em inglês, as palavras "pet" e "bet" são acusticamente diferentes. No entanto, quando observado visualmente (sem som), eles são exatamente iguais.

A tabela a seguir lista mostra uma lista parcial dos fonemas do Alfabeto Fonético Internacional (AFI) e os símbolos do Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala), bem como os visemas correspondentes para vozes do idioma inglês (EUA).

Para obter a tabela completa e tabelas de todos os idiomas disponíveis, consulte [Tabelas de fonemas e visemas para os idiomas compatíveis \(p. 19\)](#).

AFI	X-SAMPA	Descrição	Exemplo	Visema
Consoantes				
b	b	Oclusiva bilabial sonora	bed	p
d	d	Oclusiva alveolar sonora	dig	t
dʒ	dZ	Africada pós-alveolar sonora	jump	S
ð	D	Fricativa dental sonora	then	T
f	f	Fricativa labiodental surda	five	f
g	g	Oclusiva velar sonora	game	k
h	h	Fricativa glotal surda	house	k
...

Uso de marcas de fala

Solicitação de marcas de fala

Para solicitar marcas de fala para o texto para entrada, use o comando `synthesize-speech`. Além do texto de entrada, os seguintes elementos são necessários para retornar esses metadados:

- `output-format`

O Amazon Polly é compatível apenas com o formato JSON ao retornar marcas de fala.

```
--output-format json
```

Se você usar um formato de saída não compatível, o Amazon Polly lançará uma exceção.

- `voice-id`

Para garantir que os metadados correspondam ao fluxo de áudio associado, especifique a mesma voz usada para gerar o fluxo de áudio de fala sintetizado. As vozes disponíveis não têm velocidades de fala idênticas. Se você usar uma voz diferente daquela usado para gerar a fala, os metadados não corresponderão ao fluxo de áudio.

```
--voice-id Joanna
```

- `speech-mark-types`

Especifique os tipos de marcas de fala desejados. Você pode solicitar qualquer um ou todos os tipos de marca de fala, mas deve especificar pelo menos um tipo.

```
--speech-mark-types='["sentence", "word", "viseme", "ssml"]'
```

- **text-type**

Texto sem formatação é o padrão para texto de entrada no Amazon Polly, de modo que você deve usar `otext-type ssml` se você deseja retornar marcas de fala SSML.

- **outfile**

Especifique o arquivo de saída ao qual os metadados são gravados.

```
MaryLamb.txt
```

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (\) no final de cada linha por um circunflexo (^) e use aspas completas (") ao redor do texto de entrada por aspas simples (') para marcas internas.

```
aws polly synthesize-speech \  
  --output-format json \  
  --voice-id Voice ID \  
  --text 'Input text' \  
  --speech-mark-types='["sentence", "word", "viseme"]' \  
  outfile
```

Saída de marca de fala

O Amazon Polly retorna objetos de marca de fala em um fluxo JSON delimitado por linha. Um objeto de marca de fala contém os seguintes campos:

- **time** – a data e a hora em milissegundos desde o início do fluxo de áudio correspondente
- **type**— o tipo de marca de fala (sentence, word, viseme ou ssml)
- **start**— o deslocamento em bytes (não caracteres) do início do objeto no texto de entrada (não incluindo marcas viseme)
- **END**— o deslocamento em bytes (não caracteres) do fim do objeto no texto de entrada (não incluindo marcas do viseme)
- **value** – isso varia de acordo com o tipo de marca de fala
 - SSML: tag <mark> de SSML
 - viseme: o nome do visema
 - word ou sentence: uma substring do texto de entrada, conforme delimitado pelos campos de início e fim

Por exemplo, o Amazon Polly gera o seguinte objeto de marca da fala do texto “Mary had a little lamb”:

```
{"time":373,"type":"word","start":5,"end":8,"value":"had"}
```

A palavra descrita ("had") começa 373 milissegundos após o fluxo de áudio começar, e começa no byte 5 e termina no byte 8 do texto de entrada.

Note

Esses metadados são para a voice-id Joanna. Se você usar outra voz com o mesmo texto de entrada, os metadados do texto podem ser diferentes.

Exemplos de marca de fala

Os exemplos a seguir de solicitações de marca de fala mostram como fazer solicitações comuns e o resultado que elas geram.

Exemplo 1: Marcas de fala sem SSML

O exemplo a seguir mostra como são os metadados solicitados na sua tela para a frase simples: "Mary had a little lamb". Para simplificar, não incluímos marcas de fala SSML neste exemplo.

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (\) no final de cada linha por um circunflexo (^) e use aspas completas (") ao redor do texto de entrada por aspas simples (') para marcas internas.

```
aws polly synthesize-speech \
  --output-format json \
  --voice-id Joanna \
  --text 'Mary had a little lamb.' \
  --speech-mark-types='["viseme", "word", "sentence"]' \
  MaryLamb.txt
```

Ao fazer essa solicitação, o Amazon Polly retorna o seguinte no arquivo .txt:

```
{ "time":0, "type": "sentence", "start":0, "end":23, "value": "Mary had a little lamb." }
{ "time":6, "type": "word", "start":0, "end":4, "value": "Mary" }
{ "time":6, "type": "viseme", "value": "p" }
{ "time":73, "type": "viseme", "value": "E" }
{ "time":180, "type": "viseme", "value": "r" }
{ "time":292, "type": "viseme", "value": "i" }
{ "time":373, "type": "word", "start":5, "end":8, "value": "had" }
{ "time":373, "type": "viseme", "value": "k" }
{ "time":460, "type": "viseme", "value": "a" }
{ "time":521, "type": "viseme", "value": "t" }
{ "time":604, "type": "word", "start":9, "end":10, "value": "a" }
{ "time":604, "type": "viseme", "value": "@" }
{ "time":643, "type": "word", "start":11, "end":17, "value": "little" }
{ "time":643, "type": "viseme", "value": "t" }
{ "time":739, "type": "viseme", "value": "i" }
{ "time":769, "type": "viseme", "value": "t" }
{ "time":799, "type": "viseme", "value": "t" }
{ "time":882, "type": "word", "start":18, "end":22, "value": "lamb" }
{ "time":882, "type": "viseme", "value": "t" }
{ "time":964, "type": "viseme", "value": "a" }
{ "time":1082, "type": "viseme", "value": "p" }
```

Nessa saída, cada parte do texto é dividida em termos de marcas de fala:

- A frase "Mary had a little lamb".
- Cada palavra no texto: "Mary", "had", "a", "little" e "lamb".
- O visema de cada som no fluxo de áudio correspondente: "p", "E", "r", "i", e assim por diante. Para obter mais informações sobre visemas, consulte [Visemas e Amazon Polly \(p. 96\)](#).

Exemplo 2: Marcas de fala com SSML

O processo de gerar marcas de fala com base em texto aprimorado para SSML é semelhante ao processo quando o SSML não estiver presente. Use o comando `synthesize-speech` e especifique o texto aprimorado por SSML e o tipo de marcas de fala desejado, conforme mostrado no exemplo a seguir. Para deixar o exemplo mais fácil de ler, nós não incluímos marcas de fala do visema, mas elas também podem ser incluídas.

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (`\`) no final de cada linha por um circunflexo (`^`) e use aspas completas (`"`) ao redor do texto de entrada por aspas simples (`'`) para marcas internas.

```
aws polly synthesize-speech \
  --output-format json \
  --voice-id Joanna \
  --text-type ssml \
  --text '<speak><prosody volume="+20dB">Mary had <break time="300ms"/>a little <mark
name="animal"/>lamb</prosody></speak>' \
  --speech-mark-types='["sentence", "word", "ssml"]' \
  output.txt
```

Ao fazer essa solicitação, o Amazon Polly retorna o seguinte no arquivo `.txt`:

```
{ "time": 0, "type": "sentence", "start": 31, "end": 95, "value": "Mary had <break time=\\"300ms\\"/>a
  little <mark name=\\"animal\\"/>lamb" }
{ "time": 6, "type": "word", "start": 31, "end": 35, "value": "Mary" }
{ "time": 325, "type": "word", "start": 36, "end": 39, "value": "had" }
{ "time": 897, "type": "word", "start": 40, "end": 61, "value": "<break time=\\"300ms\\"/>" }
{ "time": 1291, "type": "word", "start": 61, "end": 62, "value": "a" }
{ "time": 1373, "type": "word", "start": 63, "end": 69, "value": "little" }
{ "time": 1635, "type": "ssml", "start": 70, "end": 91, "value": "animal" }
{ "time": 1635, "type": "word", "start": 91, "end": 95, "value": "lamb" }
```

Solicitação de marcas de fala (Console)

Você pode usar o console para solicitar marcas de fala do Amazon Polly. Em seguida, visualize os metadados ou salve-os em um arquivo.

Como gerar marcas de fala (console)

1. Faça login no AWS Management Console e abra o console do Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Selecione a guia Text-to-Speech.
3. Ativar o SSML para usar o SSML.
4. Digite ou cole o texto na caixa de entrada.
5. Para idioma, escolha o idioma para seu texto.
6. Para Voice, escolha a voz que você deseja usar para o texto.
7. Para alterar a pronúncia de texto, expanda Configurações adicionais, ative o Personalizar a pronúncia, e para Aplicar léxicos, escolha o léxicos desejado.
8. Para verificar se a fala está em sua forma final, escolha Ouça.
9. Ativar o Configurações de formato de arquivo de fala.

Note

Baixar formatos MP3, OGG ou PCM não gera marcas de fala.

10. para o Formato do arquivo, escolha Marcas de fala.
11. para o Tipos de marca de fala, escolha os tipos de marcas de fala para gerar. A opção para escolher SSML Os metadados estão disponíveis apenas quando o SSML está ligado. Para obter mais informações sobre como usar SSML com o Amazon Polly, consulte [Geração de fala a partir de documentos SSML \(p. 102\)](#).
12. Escolha Download.

Geração de fala a partir de documentos SSML

Você pode usar o Amazon Polly para gerar fala a partir de texto sem formatação ou de documentos marcados com o Speech Synthesis Markup Language (SSML). Usar um texto aprimorado por SSML permite exercer controle adicional sobre como o Amazon Polly gera fala com base no texto fornecido.

Por exemplo, você pode incluir uma pausa longa dentro do texto ou alterar a velocidade ou o tom da fala. Outras opções incluem:

- enfatizar palavras ou frases específicas
- usar a pronúncia fonética
- incluir sons de respiração
- sussurrar
- usar o estilo de locutor.

Para obter detalhes completos sobre as tags SSML compatíveis com o Amazon Polly e como usá-las, consulte [Tags SSML compatíveis \(p. 109\)](#)

Ao usar SSML, há vários caracteres reservados que exigem tratamento especial. Isso ocorre porque o SSML usa esses caracteres como parte de seu código. Para usá-los, você usa uma entidade específica para escapá-los. Para obter mais informações, consulte [Caracteres reservados no SSML \(p. 102\)](#)

O Amazon Polly fornece esses tipos de controle com um subconjunto de tags SSML definidas pelo [Speech Synthesis Markup Language \(SSML\) versão 1.1, Recomendação do W3C](#).

Você pode usar o SSML no console do Amazon Polly ou com a AWS CLI. Os tópicos a seguir mostram como você pode usar o SSML para gerar a fala e controlar a saída de modo que ele atenda perfeitamente às suas necessidades.

Tópicos

- [Caracteres reservados no SSML \(p. 102\)](#)
- [Usar SSML \(Console\) \(p. 104\)](#)
- [Uso de SSML \(AWS CLI\) \(p. 105\)](#)
- [Tags SSML compatíveis \(p. 109\)](#)

Caracteres reservados no SSML

Existem cinco caracteres predefinidos que normalmente não podem ser usados em uma instrução SSML. Essas entidades são reservadas pela especificação da linguagem. Esses caracteres são

Nome da entidade	Código de escape
<	<
>	>
"	"
'	'

Nome do caractere de escape	Código
&	E comercial
'	apóstrofo ou aspas simples
<	menor que
>	maior que

Como o SSML usa esses caracteres como parte de seu código, para usar esses símbolos no SSML, você deve escapar o caractere ao usá-lo. Use o código de escape em vez do caractere real para que ele seja exibido corretamente enquanto cria um documento SSML válido. Por exemplo, a frase a seguir

```
We're using the lawyer at Peabody & Chambers, attorneys-at-law.
```

seria renderizada em SSML como

```
<say>  
We&apos;re using the lawyer at Peabody &amp; Chambers, attorneys-at-law.  
</say>
```

Nesse caso, os caracteres especiais apóstrofo e E comercial precisam de escape para que o documento SSML permaneça válido.

Para os símbolos &, <, e > os códigos de escape são sempre necessários quando você usa SSML. Além disso, ao usar o apóstrofo/aspas simples (') como apóstrofo, você também deve usar o código de escape.

No entanto, quando você usa aspas duplas (") ou apóstrofo/aspas simples (') como aspas, a opção por usar ou não o código de escape dependerá do contexto.

Aspas duplas

- Deve ser escapado quando em um valor de atributo delimitado por aspas duplas. Por exemplo, no seguinteAWS CLIcódigo

```
--text "Pete &quot;Maverick&quot; Mitchell"
```

- Não precisa de escape quando estiver no contexto textual. Por exemplo, no seguinte

```
He said, "Turn right at the corner."
```

- Não precisa de escape quando em um valor de atributo delimitado por aspas simples. Por exemplo, no seguinteAWS CLIcódigo

```
--text 'Pete "Maverick" Mitchell'
```

Aspas simples

- Precisa de escape quando usado como apóstrofo. Por exemplo, no seguinte

```
We&apos;ve got to leave quickly.
```

- Não precisa de escape quando estiver no contexto textual. Por exemplo, no seguinte

```
"And then I said, 'Don't quote me.'"
```

- Não precisa de escape quando em um atributo de código delimitado por aspas duplas. Por exemplo, no seguinte

```
--text "Pete 'Maverick' Mitchell"
```

Usar SSML (Console)

Com as tags do SSML, você pode personalizar e controlar diversos aspectos da fala, como pronúncia, volume, tom e velocidade. No AWS Management Console, o texto aprimorado por SSML que você deseja converter em áudio será inserido na guia SSML da página Texto para fala. Embora o texto inserido sem formatação conte com configurações padrão para o idioma e a voz escolhidos, o texto aprimorado com SSML informa o Amazon Polly não apenas o que você deseja dizer, mas como deseja dizer. Exceto para as tags SSML adicionadas, o Amazon Polly sintetiza texto aprimorados por SSML da mesma maneira que ele sintetiza texto sem formatação. Consulte [Exercício 2: Sintetização de fala com entrada de texto sem formatação \(console\) \(p. 6\)](#) para obter mais informações.

Ao usar SSML, coloque todo o texto em um `< speak >` Tag para permitir que o Amazon Polly saiba que você está usando SSML. Por exemplo:

```
< speak >Hi! My name is Joanna. I will read any text you type here.</ speak >
```

Em seguida, use tags SSML específicas no texto dentro das tags `< speak >` para personalizar a maneira como você deseja que o texto soe. Você pode adicionar uma pausa, alterar o ritmo da fala, diminuir ou aumentar o volume da voz ou adicionar muitas outras personalizações para que o texto soe da forma adequada para você. Para obter uma lista completa das tags SSML que você pode usar, consulte [Tags SSML compatíveis \(p. 109\)](#).

No exemplo a seguir, você usa uma tag SSML para instruir o Amazon Polly a substituir o "World Wide Web Consortium" pelo "W3C" quando se refere a um parágrafo curto. Você também pode usar tags para inserir uma pausa e sussurrar uma palavra. Compare os resultados deste exercício com os do [Aplicação dos léxicos usando o console \(sintetizar fala\) \(p. 133\)](#).

Para obter mais informações sobre o SSML por meio de exemplos, consulte [Tags SSML compatíveis \(p. 109\)](#).

Como sintetizar a fala com base no texto aprimorado por SSML (console)

1. Faça login no AWS Management Console e abra o console do Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Selecione a guia Text-to-Speech (Texto para fala) se ela ainda não estiver sendo exibida.
3. Ativar o SSML.
4. Digite ou cole o texto a seguir na caixa de texto:

```
< speak >
```

```
He was caught up in the game.<break time="1s"/> In the middle of the
10/3/2014 <sub alias="World Wide Web Consortium">W3C</sub> meeting,
he shouted, "Nice job!" quite loudly. When his boss stared at him, he repeated
<amazon:effect name="whispered">"Nice job,"</amazon:effect> in a
whisper.
</speak>
```

As tags SSML informam o Amazon Polly como renderizar o texto:

- `<break time="1s"/>`O comando informa o Amazon Polly sobre a pausa de 1 segundo entre as duas primeiras frases.
- `_{W3C}`O comando informa o Amazon Polly sobre a substituição de “World Wide Web Consortium” pelo acrônimo “W3C”.
- `<amazon:effect name="whispered">Nice job</amazon:effect>`O comando informa o Amazon Polly sobre o sussurro da segunda instância de “Bom trabalho”.

Note

Ao usar a AWS CLI, você coloca o texto de entrada entre aspas para diferenciá-lo do código ao redor. O console do Amazon Polly não mostra o código, para que você não coloque um texto de entrada entre aspas quando usá-lo.

5. para oldioma, escolhaInglês dos EUAe, em seguida, escolha uma voz.
6. Para ouvir a fala, selecioneOuça..
7. Para salvar o arquivo de fala, selecioneBaixar. Se você deseja salvá-lo em um formato diferente, expanda oConfigurações adicionais, ative oConfigurações de formato de arquivo de falae escolha o formato desejado e, em seguida, escolhaBaixar.

Uso de SSML (AWS CLI)

Você pode usar a AWS CLI para sintetizar o texto de entrada SSML. Os exemplos a seguir mostram como realizar tarefas comuns usando a AWS CLI.

Tópicos

- [Como usar o SSML com o comando Synthesize-Speech \(p. 105\)](#)
- [Sintetizar um documento aprimorado por SSML \(p. 106\)](#)
- [Como usar o SSML para tarefas comuns do Amazon Polly \(p. 107\)](#)

Como usar o SSML com o comando Synthesize-Speech

Este exemplo mostra como usar o comando `synthesize-speech` com uma string SSML. Quando você usa o comando `synthesize-speech`, normalmente fornece estas informações:

- O texto de entrada (obrigatório)
- Tags de abertura e fechamento (obrigatórias)
- O formato da saída
- Uma voz

Neste exemplo, você especifica uma string de texto simples entre aspas juntamente com as tags `<speak></speak>` de abertura e fechamento necessárias.

Important

Embora você não use aspas ao redor do texto de entrada no console do Amazon Polly, é necessário usá-las na AWS CLI. Também é importante que você diferencie as aspas ao redor do texto de entrada e as aspas necessárias para tags individuais.

Por exemplo, você pode usar aspas padrão (") ao redor do texto de entrada e aspas simples (') em tags internas, ou ao contrário. Essas opções funcionam para Unix, Linux e macOS. No entanto, no Windows, é necessário colocar o texto de entrada em aspas padrão e usar aspas simples para as tags.

Em todos os sistemas operacionais, você pode usar aspas padrão (") ao redor do texto de entrada e aspas simples (') em tags internas. Por exemplo:

```
--text "<speak>Hello <break time='300ms' /> World</speak>"
```

Nos sistemas operacionais Unix, Linux e macOS, você também pode usar essas opções ao contrário, ou seja, aspas simples (') ao redor do texto de entrada e aspas padrão (") em tags internas:

```
--text '<speak>Hello <break time="300ms" /> World</speak>'
```

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (\) no final de cada linha por um circunflexo (^) e use aspas completas (") ao redor do texto de entrada por aspas simples (') para tags internas.

```
aws polly synthesize-speech \  
--text-type ssm1 \  
--text '<speak>Hello world</speak>' \  
--output-format mp3 \  
--voice-id Joanna \  
speech.mp3
```

Para ouvir a fala sintetizada, reproduza o arquivo `speech.mp3` resultante usando qualquer reprodutor de áudio.

Sintetizar um documento aprimorado por SSML

Em textos de entrada mais longos, convém salvar o conteúdo SSML em um arquivo e simplesmente especificar o nome do arquivo no comando `synthesize-speech`. Por exemplo, você pode salvar os comandos a seguir em um arquivo chamado `example.xml`:

```
<?xml version="1.0"?>  
<speak version="1.1"  
  xmlns="http://www.w3.org/2001/10/synthesis"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:schemaLocation="http://www.w3.org/2001/10/synthesis http://www.w3.org/TR/speech-synthesis11/synthesis.xsd"  
  xml:lang="en-US">Hello World</speak>
```

O atributo `xml:lang` especifica `en-US` (inglês EUA) como o idioma do texto de entrada. Para obter informações sobre como o idioma do texto de entrada e o idioma da voz selecionada afetam a operação `SynthesizeSpeech`, consulte [Como melhorar a pronúncia de palavras estrangeiras \(p. 109\)](#).

Como executar um arquivo aprimorado por SSML

1. Salve o SSML em um arquivo (por exemplo, `example.xml`).

2. Execute o seguinte comando `synthesize-speech` no caminho onde o arquivo XML está armazenado e especifique o arquivo SSML como entrada, substituindo `file:\\example.xml` pelo texto de entrada. Como esse comando direciona para um arquivo em vez de conter o texto de entrada em si, não são necessárias aspas.

Note

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (`\`) no final de cada linha por um circunflexo (`^`).

```
aws polly synthesize-speech \  
--text-type ssml \  
--text file://example.xml \  
--output-format mp3 \  
--voice-id Joanna \  
speech.mp3
```

3. Para ouvir a fala sintetizada, reproduza o arquivo `speech.mp3` resultante usando qualquer reprodutor de áudio.

Como usar o SSML para tarefas comuns do Amazon Polly

Os exemplos a seguir mostram como usar tags SSML para concluir tarefas comuns do Amazon Polly. Para obter mais tags SSML, consulte [Tags SSML compatíveis \(p. 109\)](#).

Para testar os exemplos a seguir, use o comando `synthesize-speech` com o texto aprimorado por SSML adequado:

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (`\`) no final de cada linha por um circunflexo (`^`) e use aspas completas (`"`) ao redor do texto de entrada por aspas simples (`'`) para tags internas.

```
aws polly synthesize-speech \  
--text-type ssml \  
--text '<Speak>Hello <break time="300ms"/> World</Speak>' \  
--output-format mp3 \  
--voice-id Joanna \  
speech.mp3
```

Adicionar uma pausa

Para adicionar uma pausa entre as palavras, use o elemento `<break>`. O seguinte comando SSML `synthesize-speech` usa o elemento `<break>` para adicionar um atraso de 300 milissegundos entre as palavras "Hello" e "World".

```
<Speak>  
  Hello <break time="300ms"/> World.  
</Speak>
```

Como controlar volume, tom e velocidade

Use o elemento `<prosody>` para controlar o tom, a taxa e o volume da fala.

- O comando `synthesize-speech` a seguir usa o elemento `<prosody>` para controlar o volume:

```
<say-as>
  <prosody volume="+20dB">Hello world</prosody>
</say-as>
```

- O comando `synthesize-speech` a seguir usa o elemento `<prosody>` para controlar o tom:

```
<say-as>
  <prosody pitch="x-high">Hello world.</prosody>
</say-as>
```

- O comando `synthesize-speech` a seguir usa o elemento `<prosody>` para especificar a taxa da fala (velocidade da fala):

```
<say-as>
  <prosody rate="x-fast">Hello world.</prosody>
</say-as>
```

- Você pode especificar vários atributos em um elemento `<prosody>`, como mostrado nos exemplos a seguir:

```
<say-as>
  <prosody volume="x-loud" pitch="x-high" rate="x-fast">Hello world.</prosody>
</say-as>
```

Whispering

Para sussurrar palavras, use o elemento `<amazon:effect name="whispered">`. No exemplo a seguir, o `<amazon:effect name="whispered">` elemento instrui o Amazon Polly a sussurrar "little lamb":

```
<say-as>
  Mary has a <amazon:effect name="whispered">little lamb.</amazon:effect>
</say-as>
```

Para aprimorar esse efeito, use o elemento `<prosody>` para diminuir levemente a velocidade da fala sussurrada.

Enfatizar palavras

Para destacar uma palavra ou frase, use o elemento `<emphasis>`.

```
<say-as>
  <emphasis level="strong">Hello</emphasis> world how are you?
</say-as>
```

Como especificar a forma como certas palavras são ditas

Para fornecer informações sobre o tipo de texto a ser dito, use o elemento `<say-as>`.

Por exemplo, no SSML a seguir, o `<say-as>` indica que o texto 4/6 deve ser interpretado como uma data. O atributo `interpret-as="date" format="dm"` indica que ele deve ser falado como uma data com o formato mês/dia.

Você também pode usar o `<say-as>` elemento para instruir o Amazon Polly a dizer números como frações, números de telefone, unidades de medida e muito mais.

```
<say-as interpret-as="date" format="md" >4/6</say-as>
```

A fala resultante é "Today is June 4th" (Hoje é 4 de junho). A tag `<say-as>` descreve como o texto deve ser interpretado ao fornecer contexto adicional com o atributo `interpret-as`.

Reproduza o arquivo `speech.mp3` resultante para verificar a precisão da fala sintetizada.

Para obter mais informações sobre esse elemento, consulte [Controlar como tipos especiais de palavras são ditas](#) (p. 119).

Como melhorar a pronúncia de palavras estrangeiras

O Amazon Polly pressupõe que o texto de entrada está no mesmo idioma da voz selecionada. Para melhorar a pronúncia de palavras estrangeiras no texto de entrada, na chamada `synthesize-speech`. Especifique o idioma de destino com o atributo `xml:lang`. Isso instrui o Amazon Polly a aplicar diferentes regras de pronúncia às palavras estrangeiras que você marcar.

Os exemplos a seguir mostram como usar diferentes combinações de idiomas no texto de entrada e como especificar vozes e a pronúncia de palavras estrangeiras. Para obter uma lista completa dos idiomas disponíveis, consulte [Idiomas compatíveis com o Amazon Polly](#) (p. 18).

No exemplo a seguir, a voz (Joanna) é uma voz do Inglês dos EUA. Por padrão, o Amazon Polly pressupõe que o texto de entrada está no mesmo idioma da voz (neste caso, Inglês dos EUA). Quando você usa o `xml:lang` Tag, o Amazon Polly interpreta o texto como espanhol e o texto é falado da maneira como a voz selecionada pronunciaria palavras em espanhol, de acordo com as regras de pronúncia do idioma estrangeiro. Sem essa tag, o texto seria falado usando as regras de pronúncia da voz selecionada.

```
<say-as>  
That restaurant is terrific. <lang xml:lang="es-ES">Mucho gusto.</lang>  
</say-as>
```

Como o idioma do texto de entrada é Inglês, o Amazon Polly mapeia os fonemas em espanhol para os fonemas em inglês mais próximos. Como resultado, Joanna fala o texto como um nativo dos EUA, que pronuncia as palavras corretamente em Espanhol, mas com um sotaque do Inglês dos EUA.

Note

Alguns idiomas são mais semelhantes do que outros e, portanto, algumas combinações de idiomas funcionam melhor que outras.

Tags SSML compatíveis

O Amazon Polly oferece suporte às seguintes tags SSML:

Ação	Tag SSML	Disponibilidade com vozes neurais
Adicionar uma pausa (p. 111)	<code><break></code>	Disponibilidade total
Enfatizar palavras (p. 111)	<code><emphasis></code>	Não disponível
Especificar outro idioma para palavras específicas (p. 112)	<code><lang></code>	Disponibilidade total

Ação	Tag SSML	Disponibilidade com vozes neurais
Colocar uma tag personalizada no texto (p. 113)	<mark>	Disponibilidade total
Adicionar uma pausa entre parágrafos (p. 113)	<p>	Disponibilidade total
Usar a pronúncia fonética (p. 114)	<phoneme>	Disponibilidade total
Controlar o volume, a velocidade e o tom da fala (p. 115)	<prosody>	Disponibilidade parcial
Configurar uma duração máxima para fala sintetizada (p. 116)	<prosody amazon:max-duration>	Não disponível
Adicionar uma pausa entre as frases (p. 119)	<s>	Disponibilidade total
Controlar como tipos especiais de palavras são ditas (p. 119)	<say-as>	Disponibilidade parcial
Identificar um texto aprimorado por SSML (p. 110)	<speak>	Disponibilidade total
Como pronunciar acrônimos e abreviaturas (p. 122)	<sub>	Disponibilidade total
Melhorar a pronúncia especificando partes da fala (p. 122)	<w>	Disponibilidade total
Adicionar o som da respiração (p. 123)	<amazon:auto-breaths>	Não disponível
Estilo de locutor (p. 126)	<amazon:domain name="news">	Selecionar somente vozes neurais
Adicionar compactação de intervalo dinâmico (p. 126)	<amazon:effect name="drc">	Disponibilidade total
Falar suavemente (p. 128)	<amazon:effect phonation="soft">	Não disponível
Controlar o timbre (p. 128)	<amazon:effect vocal-tract-length>	Não disponível
Sussurrar (p. 129)	<amazon: effect name="whispered">	Não disponível

Se você usar tags SSML incompatíveis no formato neural ou padrão, receberá um erro.

Identificar um texto aprimorado por SSML

<speak>

O <speak>tag é o elemento raiz de todos os textos SSML do Amazon Polly. Todos os textos aprimorados por SSML devem estar entre tags <speak>.

```
<speak>Mary had a little lamb.</speak>
```

Essa tag é compatível com os formatos TTS padrão e neural.

Adicionar uma pausa

`<break>`

Essa tag é compatível com os formatos TTS padrão e neural.

Para adicionar uma pausa ao texto, use a tag `<break>`. Você pode definir uma pausa com base na intensidade (equivalente à pausa após uma vírgula, uma frase ou um parágrafo), ou configurá-la para um determinado período em segundos ou milissegundos. Se você não especificar um atributo para determinar o tamanho da pausa, o Amazon Polly usa o padrão, que é `<break strength="medium">`, que adiciona o tamanho de uma pausa após uma vírgula.

Valores de atributo `strength`:

- `none`: nenhuma pausa. Use `none` para remover uma pausa comum, como a que ocorre depois de um ponto final.
- `x-weak`: tem a mesma função de `none`; nenhuma pausa.
- `weak`: define uma pausa com a mesma duração da pausa após uma vírgula.
- `medium`: tem a mesma função de `weak`.
- `strong`: define uma pausa com a mesma duração da pausa após uma frase.
- `x-strong`: define uma pausa com a mesma duração da pausa após um parágrafo.

Valores de atributo `time`:

- `[number]s`: a duração da pausa, em segundos. A duração máxima é de 10s.
- `[number]ms`: a duração da pausa em milissegundos. A duração máxima é de 10000ms.

Por exemplo:

```
<say>
  Mary had a little lamb <break time="3s"/>Whose fleece was white as snow.
</say>
```

Se você não usar um atributo com a tag `break`, o resultado variará dependendo do texto:

- Se não houver mais nenhuma pontuação ao lado da tag `break`, ela criará uma `<break strength="medium">` (pausa de duração de vírgula).
- Se a tag estiver ao lado de uma vírgula, ela atualizará a tag para uma `<break strength="strong">` (pausa de duração de frase).
- Se a tag estiver ao lado de um ponto final, ela atualizará a tag para uma `<break strength="x-strong">` (pausa de duração de parágrafo).

Enfatizar palavras

`<emphasis>`

Essa tag é compatível somente com o formato TTS padrão.

Para enfatizar palavras, use a tag `<emphasis>`. Quando há ênfase de palavras, a taxa e o volume da fala mudam. Uma ênfase maior significa que o texto será falado pelo Amazon Polly em um volume mais alto e velocidade mais lenta. Uma ênfase menor significa que a fala será mais tranquila e mais lenta. Para especificar o grau de ênfase, use o atributo `level`.

Valores de atributo `level`:

- **Strong**: aumenta o volume e desacelera a taxa da fala para que ela soe mais alta e mais lentamente.
- **Moderate**: aumenta o volume e diminui a taxa da fala, mas menos do que **strong**. **Moderate** é o padrão.
- **Reduced**: diminui o volume e aumenta a taxa da fala. A fala é mais suave e mais rápida.

Note

A taxa e o volume normais de uma voz ficam entre os níveis `moderate` e `reduced`.

Por exemplo:

```
< speak >
  I already told you I < emphasis level="strong">really like</ emphasis > that person.
</ speak >
```

Especificar outro idioma para palavras específicas

`<lang>`

Essa tag é compatível com os formatos TTS padrão e neural.

Especifique outro idioma para uma determinada palavra, frase ou frase com a tag `<lang>`. Frases e palavras em idioma estrangeiro geralmente são ditas com melhor sonoridade quando são inseridas dentro de um par de tags `<lang>`. Para especificar o idioma, use o atributo `xml:lang`. Para obter uma lista completa dos idiomas disponíveis, consulte [Idiomas compatíveis com o Amazon Polly \(p. 18\)](#).

A menos que você aplique a tag `<lang>`, todas as palavras no texto de entrada serão faladas no idioma da voz especificada no `voice-id`. Se você aplicar a tag `<lang>`, as palavras serão faladas no idioma selecionado.

Por exemplo, se o `voice-id` é Joanna (que fala Inglês dos EUA), o Amazon Polly falará a seguinte frase na voz da Joanna, sem sotaque Francês:

```
< speak >
  Je ne parle pas français.
</ speak >
```

Se você usar a voz de Joanna com o `<lang>` Tag, Amazon Polly falará a frase na voz da Joanna em francês com sotaque americano:

```
< speak >
  < lang xml:lang="fr-FR">Je ne parle pas français.</ lang >.
</ speak >
```

Como Joanna não é uma voz nativa do Francês, a pronúncia se baseia no idioma nativo dela, que é o Inglês dos EUA. Por exemplo, apesar da pronúncia perfeita em Francês ter uma vibrante uvular /

R/ na palavra francês, a voz em Inglês dos EUA da Joanna pronuncia esse fonema como o som /r/ correspondente.

Se você usar `ovoice-id` De Giorgio, que fala italiano, com o seguinte texto, Amazon Polly falará a frase na voz de Giorgio com uma pronúncia italiana:

```
< speak>
  Mi piace Bruce Springsteen.
</ speak>
```

Se você usar a mesma voz com o seguinte `<lang>` Tag, Amazon Polly pronunciará Bruce Springsteen em Inglês com sotaque Italiano:

```
< speak>
  Mi piace < lang xml: lang="en-US"> Bruce Springsteen. </ lang>
</ speak>
```

Essa tag também pode ser usada como substituta da opção [DefaultLangCode](#) opcional ao sintetizar a fala. No entanto, isso requer que você formate o texto usando SSML.

Colocar uma tag personalizada no texto

`<mark>`

Essa tag é compatível com os formatos TTS padrão e neural.

Para colocar uma tag personalizada no texto, use a `<mark>` tag. O Amazon Polly não realiza nenhuma ação na tag, mas retorna a localização dela nos metadados do SSML. Essa tag pode ter qualquer nome que você desejar, desde que ela mantenha o seguinte formato:

```
< mark name="tag_name" />
```

Por exemplo, se o nome da tag for "animal" e o texto de entrada for:

```
< speak>
  Mary had a little < mark name="animal" /> lamb.
</ speak>
```

O Amazon Polly pode retornar os seguintes metadados SSML:

```
{ "time": 767, "type": "ssml", "start": 25, "end": 46, "value": "animal" }
```

Adicionar uma pausa entre parágrafos

`<p>`

Essa tag é compatível com os formatos TTS padrão e neural.

Para adicionar uma pausa entre parágrafos no texto, use a tag `<p>`. O uso dessa tag fornece uma pausa maior do que aquelas geralmente colocadas por falantes nativos em vírgulas ou final de frases. Use a tag `<p>` para incluir o parágrafo:

```
< speak>
  < p>This is the first paragraph. There should be a pause after this text is spoken.< /
p>
  < p>This is the second paragraph.< /p>
< /speak>
```

Isso é equivalente a especificar uma pausa usando `< break strength="x-strong"/>`.

Usar a pronúncia fonética

`< phoneme>`

Essa tag é compatível com os formatos TTS padrão e neural.

Para que o Amazon Polly use a pronúncia fonética em um texto específico, use a `< phoneme>` tag.

Dois atributos são necessários com a tag `< phoneme>`. Eles indicam o alfabeto fonético que o Amazon Polly usa e os símbolos fonéticos da pronúncia corrigida:

- `alphabet`
 - `ipa` – indica que o sistema do Alfabeto Fonético Internacional (AFI) será usado.
 - `x-sampa` – indica que o sistema Extended Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético estendido de métodos de avaliação da fala) será usado.
- `ph`
 - Especifica os símbolos fonéticos da pronúncia. Para obter mais informações, consulte [Tabelas de fonemas e visemas para os idiomas compatíveis \(p. 19\)](#)

Com a `< phoneme>`, o Amazon Polly usa a pronúncia especificada pela tag `ph` ao invés da pronúncia padrão associada ao idioma da voz selecionada.

Por exemplo, a palavra "pecan" pode ser pronunciada de duas formas. No exemplo a seguir, é atribuída uma pronúncia personalizada, diferente em cada linha, à palavra "pecan". O Amazon Polly pronuncia pecan conforme especificado no `ph` atributo, em vez de usar a pronúncia padrão.

Alfabeto Fonético Internacional (AFI)

```
< speak>
  You say, < phoneme alphabet="ipa" ph="p##k##n">pecan< /phoneme>.
  I say, < phoneme alphabet="ipa" ph="#pi.kæn">pecan< /phoneme>.
< /speak>
```

Extended Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fonético de métodos de avaliação da fala).

```
< speak>
  You say, < phoneme alphabet='x-sampa' ph='pI"kA:n'>pecan< /phoneme>.
  I say, < phoneme alphabet='x-sampa' ph='"pi.k{n'>pecan< /phoneme>.
< /speak>
```

Além disso, o chinês mandarim usa Pinyin para pronúncia fonética .

Pinyin

```
< speak>
  ## < phoneme alphabet="x-amazon-pinyin" ph="bo2">#< /phoneme>#
```



```
## <phoneme alphabet="x-amazon-pinyin" ph="bao2">#</phoneme>#  
</speak>
```

Controlar o volume, a velocidade e o tom da fala

<prosody>

Os atributos da tag Prosody são totalmente compatíveis com as vozes TTS padrão. Vozes neurais suportam `ovolumeerate`, mas não oferecem suporte a `opitchAttribute`.

Para controlar o volume, a velocidade ou o tom da voz selecionada, use a tag `prosody`.

O volume, a taxa e o tom da fala dependem da voz selecionada. Além das diferenças entre vozes em idiomas diferentes, há diferenças entre vozes individuais falando o mesmo idioma. Por isso, embora os atributos sejam semelhantes em todos os idiomas, há variações nítidas entre idiomas e nenhum valor absoluto estará disponível.

A tag `prosody` tem três atributos, e cada um deles têm vários valores disponíveis para definir o atributo. Todos os atributos usam a mesma sintaxe:

```
<prosody attribute="value"></prosody>
```

- **volume**
 - `default`: redefine o volume como o nível padrão da voz atual.
 - `silent`, `x-soft`, `soft`, `medium`, `loud`, `x-loud`: define o volume como um valor predefinido para a voz atual.
 - `+ndB`, `-ndB`: altera o volume com relação ao nível atual. Um valor de `+0dB` significa nenhuma alteração, `+6dB` significa aproximadamente o dobro do volume atual e `-6dB` significa aproximadamente metade do volume atual.

Por exemplo, você pode definir o volume para um trecho das seguintes maneiras:

```
<speak>  
    Sometimes it can be useful to <prosody volume="loud">increase the volume  
    for a specific speech.</prosody>  
</speak>
```

Ou você pode definir assim:

```
<speak>  
    And sometimes a lower volume <prosody volume="-6dB">is a more effective way of  
    interacting with your audience.</prosody>  
</speak>
```

- **rate**
 - `x-slow`, `slow`, `medium`, `fast`, `x-fast`: define o tom para um valor predefinido da voz selecionada.
 - `n%`: uma alteração de porcentagem não negativa na taxa da fala. Por exemplo, um valor de `100%` significa nenhuma alteração na taxa da fala, `200%` significa que a taxa é o dobro da taxa padrão e `50%` que taxa é metade da taxa padrão. Esse valor varia de 20 a 200%.

Por exemplo, você pode definir a taxa da fala para um trecho das seguintes maneiras:

```
<speak>  
    For dramatic purposes, you might wish to <prosody rate="slow">slow up the speaking
```

```
rate of your text.</prosody>
</speak>
```

Ou você pode definir assim:

```
<speak>
  Although in some cases, it might help your audience to <prosody rate="85%">slow
  the speaking rate slightly to aid in comprehension.</prosody>
</speak>
```

- **pitch**
 - **default**: redefine o tom como o nível padrão da voz atual.
 - **x-low, low, medium, high, x-high**: Define o tom do valor predefinido para a voz atual.
 - **+n%ou-n%**: Ajusta o tom por uma porcentagem relativa. Por exemplo, um valor de +0% significa nenhuma alteração no tom da linha de base, +5% fornece um tom de linha de base um pouco maior e -5% resulta em um tom de linha de base um pouco menor.

Por exemplo, você pode definir o tom para um trecho das seguintes maneiras:

```
<speak>
  Do you like synthesized speech <prosody pitch="high">with a pitch that is higher
  than normal?</prosody>
</speak>
```

Ou você pode definir assim:

```
<speak>
  Or do you prefer your speech <prosody pitch="-10%">with a somewhat lower pitch?</
prosody>
</speak>
```

A tag `<prosody>` deve conter pelo menos um atributo, mas pode incluir mais atributos na mesma tag.

```
<speak>
  Each morning when I wake up, <prosody volume="loud" rate="x-slow">I speak
  quite slowly and deliberately until I have my coffee.</prosody>
</speak>
```

Ela também pode ser combinada com tags aninhadas da seguinte forma:

```
<speak>
  <prosody rate="85%">Sometimes combining attributes <prosody pitch="-10%">can
  change the impression your audience has of a voice</prosody> as well.</prosody>
</speak>
```

Configurar uma duração máxima para fala sintetizada

`<prosody amazon:max-duration>`

No momento, essa tag é compatível somente com o formato TTS padrão.

Para controlar quanto tempo você deseja que dure uma fala quando ela for sintetizada, use a tag `<prosody>` com o atributo `amazon:max-duration`.

A duração da fala sintetizada varia levemente, dependendo da voz selecionada. Isso pode dificultar a correspondência da fala sintetizada com recursos visuais ou outras atividades que exigem uma sincronização precisa. Esse problema é ampliado em aplicativos de tradução, pois o tempo usado para dizer certas frases pode variar amplamente entre diferentes idiomas.

A tag `<prosody amazon:max-duration>` corresponde a fala sintetizada à quantidade de tempo que você deseja que ela leve (a duração).

Essa tag usa a seguinte sintaxe:

```
<prosody amazon:max-duration="time duration">
```

Com a tag `<prosody amazon:max-duration>`, é possível especificar a duração em segundos ou milissegundos:

- `ns`: a duração máxima em segundos
- `ms`: a duração máxima em milissegundos

Por exemplo, o seguinte texto falado tem uma duração máxima de 2 segundos:

```
<say>  
  <prosody amazon:max-duration="2s">  
    Human speech is a powerful way to communicate.  
  </prosody>  
</say>
```

Texto colocado dentro da tag não excede a duração especificada. Se a voz ou o idioma escolhido normalmente levam mais tempo do que aquela duração, o Amazon Polly acelera a fala para que ele se encaixe na duração especificada.

Se a duração especificada leva mais tempo do que o necessário para ler o texto em uma velocidade normal, o Amazon Polly lê a fala normalmente. Ele não diminui a velocidade da fala ou adiciona silêncio, portanto o áudio resultante fica mais curto do que o solicitado.

Note

O Amazon Polly aumenta a velocidade não mais do que 5 vezes a velocidade normal. Se o texto for falado mais rápido do que isso, normalmente não fará sentido. Se uma fala não couber na duração especificada, mesmo quando acelerada ao máximo, o áudio será acelerado mas durará mais do que a duração especificada.

É possível incluir uma única frase ou várias frases em uma tag `<prosody amazon:max-duration>`, e também é possível usar várias tags `<prosody amazon:max-duration>` em seu texto.

Por exemplo:

```
<say>  
  <prosody amazon:max-duration="2400ms">  
    Human speech is a powerful way to communicate.  
  </prosody>  
  <break strength="strong"/>  
  <prosody amazon:max-duration="5100ms">  
    Even a simple 'Hello' can convey a lot of information depending on the pitch,  
    intonation, and tempo.  
  </prosody>  
  <break strength="strong"/>  
  <prosody amazon:max-duration="8900ms">  
    We naturally understand this information, which is why speech is ideal for creating  
    applications where
```

```
        a screen isn't practical or possible, or simply isn't convenient.  
    </prosody>  
</speak>
```

Como usar o <prosody amazon:max-duration>tag pode aumentar a latência quando o Amazon Polly retornar a fala sintetizada. O grau de latência depende da passagem e da duração. Recomendamos usar um texto composto de passagens relativamente curtas.

Limitações

Há limitações em como você usa a tag <prosody amazon:max-duration> e em como ela funciona com outras tags SSML:

- O texto dentro de uma tag <prosody amazon:max-duration> não pode ter mais de 1500 caracteres.
- Não é possível aninhar tags <prosody amazon:max-duration>. Se você colocar um <prosody amazon:max-duration>Tag dentro de outra, o Amazon Polly ignora a tag interna.

No seguinte exemplo, a tag <prosody amazon:max-duration="5s"> é ignorada:

```
<speak>  
  <prosody amazon:max-duration="16s">  
    Human speech is a powerful way to communicate.  
  
    <prosody amazon:max-duration="5s">  
      Even a simple 'Hello' can convey a lot of information depending on the  
pitch, intonation, and tempo.  
    </prosody>  
  
    We naturally understand this information, which is why speech is ideal for  
creating applications where a screen isn't practical or possible, or simply isn't  
convenient.  
  </prosody>  
</speak>
```

- Não é possível usar as tags <prosody> com o atributo rate dentro de uma tag <prosody amazon:max-duration>. Isso ocorre porque as duas afetam a velocidade de fala do texto.

No exemplo a seguir, o Amazon Polly ignora o <prosody rate="2">tag:

```
<speak>  
  <prosody amazon:max-duration="7500ms">  
    Human speech is a powerful way to communicate.  
  
    <prosody rate="2">  
      Even a simple 'Hello' can convey a lot of information depending on the  
pitch, intonation, and tempo.  
    </prosody>  
  </prosody>  
</speak>
```

Pausas e max-duration

Ao usar a tag max-duration, você ainda poderá inserir pausas dentro do texto. No entanto, o Amazon Polly inclui a duração da pausa ao calcular a duração máxima da fala. Além disso, o Amazon Polly preserva as pausas breves que ocorrem onde vírgulas e pontos são colocados em uma passagem e inclui na duração máxima.

Por exemplo, no seguinte bloco, a pausa de 600 milissegundos e as pausas causadas pelas vírgulas e pontos ocorrem dentro da fala de 8 segundos:

```
<say-as>
  <prosody amazon:max-duration="8s">
    Human speech is a powerful way to communicate.
    <break time="600ms"/>
    Even a simple 'Hello' can convey a lot of information depending on the pitch,
    intonation, and tempo.
  </prosody>
</say-as>
```

Adicionar uma pausa entre as frases

<s>

Essa tag é compatível com os formatos TTS padrão e neural.

Para adicionar uma pausa entre linhas ou frases no texto, use a tag <s>. Usar essa tag gera o mesmo resultado destas ações:

- Finalizar uma frase com um ponto (.)
- Especificar uma pausa com <break strength="strong"/>

Ao contrário da tag <break>, a tag <s> inclui a frase. Isso é útil ao sintetizar falas organizadas em linhas, em vez de frases, como poesias.

No exemplo a seguir, a tag <s> cria uma curta pausa após a primeira e a segunda frase. A frase final não tem a tag <s>, mas também tem uma pausa curta depois, pois contém um ponto final.

```
<say-as>
  <s>Mary had a little lamb</s>
  <s>Whose fleece was white as snow</s>
  And everywhere that Mary went, the lamb was sure to go.
</say-as>
```

Controlar como tipos especiais de palavras são ditas

<say-as>

Exceto para o recurso "caracteres" ou "soletrar", essa tag é compatível com os formatos TTS padrão e neural. Se esse código SSML for encontrado pelo Amazon Polly em tempo de execução, a frase afetada será sintetizada usando a voz padrão relacionada. Observe, no entanto, que essa frase ainda será cobrada como se usasse a voz neural.

Usar a <say-as>tag com a taginterpret-asPara instruir o Amazon Polly a dizer determinados caracteres, palavras e números. Isso permite que você forneça contexto adicional para eliminar qualquer ambiguidade sobre como o Amazon Polly deve renderizar o texto.

A tag say-as usa um atributo, <interpret-as>, que usa um número de possíveis valores disponíveis. Todos usam a mesma sintaxe:

```
<say-as interpret-as="value">[text to be interpreted]</say-as>
```

Os valores a seguir estão disponíveis com o `interpret-as`:

- `charactersouspell-out`: Soletrará cada letra do texto, como em a-b-c.

Note

No momento, essa opção não é compatível com o uso de vozes neurais. Se esse código SSML for encontrado pelo Amazon Polly em tempo de execução, a frase afetada será sintetizada usando a voz padrão relacionada. Observe, no entanto, que essa frase ainda será cobrada como se usasse a voz neural.

- `cardinalounumber`: Interpreta o texto numérico como um número cardinal, como em 1.234.
- `ordinal`: interpreta o texto numérico como um número ordinal, por exemplo, 1.234º.
- `digits`: soletra cada dígito individualmente, como em 1-2-3-4.
- `fraction`: interpreta o texto numérico como uma fração. Isso funciona tanto em frações comuns, como 3/20, quanto em mistas, como 2 ½. Veja a seguir mais informações.
- `unit`: interpreta um texto numérico como medida. O valor deve ser um número ou uma fração que, por sua vez, é seguida por uma unidade sem espaço no meio, como em 1/2inch, ou por apenas uma unidade, como em 1meter.
- `date`: interpreta o texto como uma data. O formato de data deve ser especificado com o atributo do formato. Veja a seguir mais informações.
- `time`: interpreta o texto numérico como uma duração, em minutos e segundos, como em 1'21".
- `address`: interpreta o texto como parte de um endereço.
- `expletive`: aumenta o som do conteúdo incluído na tag.
- `telephone`: interpreta o texto numérico como um número de telefone de 7 ou 10 dígitos, como em 2025551212. Você também pode usar esse valor para lidar com ramais telefônicos, como em 2025551212x345. Veja a seguir mais informações.

Note

Atualmente, a opção `telephone` não está disponível para todos os idiomas. No entanto, ela está disponível para as vozes que falam variantes do idioma inglês (en-AU, en-GB, en-IN, en-US e en-GB-WLS), variantes do idioma espanhol (es-ES, es-MX e es-US), variantes do idioma francês (fr-FR e fr-CA) e variantes do português (pt-BR e pt-PT), bem como alemão (de-DE), italiano (it-IT), japonês (ja-JP) e russo (ru-RU). Também deve ser observado que, em alguns casos, idiomas como o árabe (arb) manipulam automaticamente o número definido como um número de telefone e, portanto, não implementam realmente a tag SSML `telephone`.

Frações

O Amazon Polly interpreta valores dentro de `say-as` que tem `interpret-as="fraction"` como frações comuns. Veja a seguir a sintaxe de frações.

- Fração

Sintaxe: *número cardinal/número cardinal*, como 2/9.

Por exemplo: `<say-as interpret-as="fraction">2/9</say-as>` é pronunciado "dois nonos."

- Número misto não negativo

Sintaxe: *número cardinal + número cardinal/número cardinal*, como 3+1/2.

Por exemplo: `<say-as interpret-as="fraction">3+1/2</say-as>` é pronunciado "três e meio".

Note

Deve haver um+Entre "3" e "1/2". O Amazon Polly não oferece suporte a um número misto sem o+, como "3 1/2".

Datas

Quando `interpret-as` for definido como `date`, você também precisará indicar o formato da data.

Neste caso, usa-se a seguinte sintaxe:

```
<say-as interpret-as="date" format="format">[date]</say-as>
```

Por exemplo:

```
<say-as>  
  I was born on <say-as interpret-as="date" format="mdy">12-31-1900</say-as>.  
</say-as>
```

Os formatos a seguir podem ser usados com o atributo `date`.

- `mdy`: mês-dia-ano.
- `dmy`: dia-mês-ano.
- `ymd`: ano-mês-dia.
- `md`: mês-dia.
- `dm`: dia-mês.
- `ym`: ano-mês.
- `my`: mês-ano.
- `d`: dia.
- `m`: Mês.
- `y`: Ano.
- `yyymmdd`: ano-mês-dia. Se você usar esse formato, o Amazon Polly poderá ignorar partes da data ao usar pontos de interrogação.

Por exemplo, o Amazon Polly renderiza o seguinte exemplo como “22 de setembro”:

```
<say-as interpret-as="date">???0922</say-as>
```

Format não é necessários.

Telefone

O Amazon Polly tenta interpretar o texto fornecido corretamente com base na formatação do texto, mesmo sem a opção `<say-as>` tag. Por exemplo, se o texto incluir “202-555-1212”, o Amazon Polly interpretará como um número de telefone de 10 dígitos e dirá cada dígito individualmente, com uma breve pausa para cada traço. Nesse caso, você não precisa usar o `<say-as interpret-as="telephone">`. No entanto, se você fornecer o texto “2025551212” e quiser que o Amazon Polly o diga como número de telefone, será necessário especificar o texto “2025551212”. `<say-as interpret-as="telephone">`.

A lógica para a interpretação de cada elemento é específica de cada idioma. Por exemplo, o Inglês dos EUA e o Inglês do Reino Unido se diferenciam na pronúncia dos números de telefone (no Inglês do Reino Unido, as sequências do mesmo dígito são agrupadas, como “double five” [dois números cinco] ou “triple four” [três números quatro]). Para ver a diferença, teste o exemplo a seguir com uma voz dos EUA e com uma voz do Reino Unido:

```
<say-as>  
  Richard's number is <say-as interpret-as="telephone">2122241555</say-as>  
</say-as>
```

Como pronunciar acrônimos e abreviaturas

`<sub>`

Essa tag é compatível com os formatos TTS padrão e neural.

Use a tag `<sub>` com o atributo `alias` para substituir uma palavra (ou pronúncia) diferente do texto selecionado, como um acrônimo ou abreviação.

Neste caso, usa-se sintaxe:

```
<sub alias="new word">abbreviation</sub>
```

No exemplo a seguir, o nome "Mercúrio" é substituído pelo símbolo químico do elemento para tornar o conteúdo do áudio mais claro.

```
<say-as>  
  My favorite chemical element is <sub alias="Mercury">Hg</sub>, because it looks so  
  shiny.  
</say-as>
```

Melhorar a pronúncia especificando partes da fala

`<w>`

Essa tag é compatível com os formatos TTS padrão e neural.

Você pode usar a tag `<w>` para personalizar a pronúncia de palavras especificando a parte da fala da palavra ou significado alternativo. Isso é feito usando o atributo `role`.

Essa tag usa a seguinte sintaxe:

```
<w role="attribute">text</w>
```

Os valores a seguir podem ser usados para o atributo `role`:

Para especificar a parte da fala:

- `amazon:VB`: interpreta a palavra como verbo (presente simples).
- `amazon:VBD`: interpreta a palavra como verbo passado.
- `amazon:DT`: interpreta a palavra como determinante.
- `amazon:IN`: interpreta a palavra como uma proposição.
- `amazon:JJ`: interpreta a palavra como adjetivo.
- `amazon:NN`: interpreta a palavra como substantivo.

Por exemplo, dependendo da parte da fala, a pronúncia do inglês dos EUA da palavra "read" varia de acordo com a tag:

```
<say-as>  
  The word <say-as interpret-as="characters">read</say-as> may be interpreted  
  as either the present simple form <w role="amazon:VB">read</w>, or the past
```



```
participle form <w role="amazon:VBD">read</w>.  
</speak>
```

Para especificar um significado específico:

- `amazon:DEFAULT`: usa o sentido padrão da palavra.
- `amazon:SENSE_1`: usa o sentido não padrão da palavra, quando presente. Por exemplo, o termo "bass" é pronunciado de forma diferente dependendo de seu significado. O significado padrão é a parte mais baixa da escala musical. O significado alternativo é uma espécie de peixe de água doce também chamado de "bass", mas com pronúncia diferente. O uso de `<w role="amazon:SENSE_1">bass</w>` renderiza a pronúncia não padrão (peixe de água fresca) para o texto de áudio.

Essa diferença na pronúncia e no significado poderá ser ouvida se você sintetizar o seguinte:

```
<speak>  
  Depending on your meaning, the word <say-as interpret-as="characters">bass</say-as>  
  may be interpreted as either a musical element: bass, or as its alternative meaning,  
  a freshwater fish <w role="amazon:SENSE_1">bass</w>.  
</speak>
```

Note

Alguns idiomas podem ter uma seleção diferente de partes da fala com suporte.

Adicionar o som da respiração

`<amazon:breath>` e `<amazon:auto-breaths>`

Essa tag é compatível somente com o formato TTS padrão.

Um fala natural inclui corretamente palavras faladas e sons de respiração. Ao adicionar sons de respiração à fala sintetizada, você pode torná-la mais natural. As tags `<amazon:breath>` e `<amazon:auto-breaths>` fornecem respirações. Você tem as seguintes opções:

- Modo manual: defina o local, o comprimento e o volume de um som de respiração no texto
- Modo automatizado: O Amazon Polly insere automaticamente sons de respiração na saída de fala
- Modo misto: você e o Amazon Polly adicionam sons de respiração

Modo manual

No modo manual, coloque a tag `<amazon:breath/>` no texto de entrada em que você deseja colocar uma respiração. Você pode personalizar o comprimento e volume de respirações com os atributos `duration` e `volume` respectivamente:

- `duration`: controla o comprimento da respiração. Os valores válidos são: `default`, `x-short`, `short`, `medium`, `long`, `x-long`. O valor padrão é `medium`.
- `volume`: controla o volume de sons de respiração. Os valores válidos são: `default`, `x-soft`, `soft`, `medium`, `loud`, `x-loud`. O valor padrão é `medium`.

Note

O comprimento e o volume exatos de cada valor de atributo depende da voz do Amazon Polly específica usada.

Para definir um som de respiração usando os valores padrão, use `<amazon:breath/>` sem atributos.

Por exemplo, para usar atributos para definir a duração e o volume de uma respiração como média, você deve definir os atributos da seguinte maneira:

```
<speak>
  Sometimes you want to insert only <amazon:breath duration="medium" volume="x-loud"/>a
  single breath.
</speak>
```

Para usar os valores padrão, você pode simplesmente usar a tag:

```
<speak>
  Sometimes you need <amazon:breath/>to insert one or more average breathes
  <amazon:breath/> so that the
  text sounds correct.
</speak>
```

Você pode adicionar sons de respiração individuais em um trecho da seguinte forma:

```
<speak>
  <amazon:breath duration="long" volume="x-loud"/> <prosody rate="120%"> <prosody
  volume="loud">
    Wow! <amazon:breath duration="long" volume="loud"/> </prosody> That was quite fast
  <amazon:breath
    duration="medium" volume="x-loud"/>. I almost beat my personal best time on this
  track. </prosody>
</speak>
```

Modo automatizado

No modo automatizado, você usa o `<amazon:auto-breaths>` Para instruir o Amazon Polly a criar automaticamente ruídos de respiração em intervalos apropriados. Você pode definir a frequência dos intervalos, o volume e a duração. Coloque a tag `</amazon:auto-breaths>` no início do texto em que você deseja aplicar a respiração automatizada e feche a tag no final.

Note

Ao contrário da tag do modo manual, `<amazon:breath/>`, a tag `<amazon:auto-breaths>` requer uma tag de fechamento (`</amazon:auto-breaths>`).

Você pode usar os seguintes atributos opcionais com a tag `<amazon:auto-breaths>`:

- **volume:** controla o volume de sons de respiração. Os valores válidos são: `default,x-soft,soft,medium,loud,x-loud`. O valor padrão é `medium`.
- **frequency:** controla a frequência com que sons respiratórios ocorrem no texto. Os valores válidos são: `default,x-low,low,medium,high,x-high`. O valor padrão é `medium`.
- **duration:** controla o comprimento da respiração. Os valores válidos são: `default,x-short,short,medium,long,x-long`. O valor padrão é `medium`.

Por padrão, a frequência de sons de respiração depende do texto de entrada. No entanto, muitas vezes, sons de respiração ocorrem após vírgulas e pontos.

Os exemplos a seguir mostram como usar a tag `<amazon:auto-breaths>`. Para decidir quais opções usar para o seu conteúdo, copie os exemplos aplicáveis para o console do Amazon Polly e escute as diferenças.

- Usando o modo automatizado sem parâmetros opcionais.

```
<speak>
  <amazon:auto-breaths>Amazon Polly is a service that turns text into lifelike
speech,
  allowing you to create applications that talk and build entirely new categories of
speech-
  enabled products. Amazon Polly is a text-to-speech service that uses advanced deep
learning
  technologies to synthesize speech that sounds like a human voice. With dozens of
lifelike
  voices across a variety of languages, you can select the ideal voice and build
speech-
  enabled applications that work in many different countries.</amazon:auto-breaths>
</speak>
```

- Usando o modo automatizado com controle de volume. O parâmetros não especificados (duration e frequency) são definidos como os valores padrão (medium).

```
<speak>
  <amazon:auto-breaths volume="x-soft">Amazon Polly is a service that turns text into
lifelike
  speech, allowing you to create applications that talk and build entirely new
categories of
  speech-enabled products. Amazon Polly is a text-to-speech service, that uses
advanced deep
  learning technologies to synthesize speech that sounds like a human voice. With
dozens of
  lifelike voices across a variety of languages, you can select the ideal voice and
build speech-
  enabled applications that work in many different countries.</amazon:auto-breaths>
</speak>
```

- Usando o modo automatizado com controle de frequência. O parâmetros não especificados (duration e volume) são definidos como os valores padrão (medium).

```
<speak>
  <amazon:auto-breaths frequency="x-low">Amazon Polly is a service that turns text
into lifelike
  speech, allowing you to create applications that talk and build entirely new
categories of
  speech-enabled products. Amazon Polly is a text-to-speech service, that uses
advanced deep
  learning technologies to synthesize speech that sounds like a human voice. With
dozens of
  lifelike voices across a variety of languages, you can select the ideal voice and
build speech-
  enabled applications that work in many different countries.</amazon:auto-breaths>
</speak>
```

- Usando o modo automatizado com parâmetros opcionais. Para o não especificadoDuration, o Amazon Polly usa o valor padrão (medium).

```
<speak>
  <amazon:auto-breaths volume="x-loud" frequency="x-low">Amazon Polly is a service
that turns
  text into lifelike speech, allowing you to create applications that talk and build
entirely new
  categories of speech-enabled products. Amazon Polly is a text-to-speech service,
that uses
  advanced deep learning technologies to synthesize speech that sounds like a human
voice. With
  dozens of lifelike voices across a variety of languages, you can select the ideal
voice and build
```

```
speech-enabled applications that work in many different countries.</amazon:auto-breaths>
</speak>
```

Estilo de locutor

<amazon:domain name="news">

O estilo de locutor está disponível somente para as vozes de Matthew ou Joanna, que estão disponíveis somente em inglês dos EUA (en-US), Lupe, em espanhol dos EUA (es-US) e Amy em inglês britânico (en-GB). Ele só é compatível ao usar o formato Neural.

Para usar o estilo de locutor, use tags SSML e a seguinte sintaxe:

```
<amazon:domain name="news">text</amazon:domain>
```

Por exemplo, use o estilo de locutor com a voz da Amy da seguinte forma:

```
<speak>
<amazon:domain name="news">
From the Tuesday, April 16th, 1912 edition of The Guardian newspaper:

The maiden voyage of the White Star liner Titanic, the largest ship ever launched, has
ended in disaster.

The Titanic started her trip from Southampton for New York on Wednesday. Late on Sunday
night she struck
an iceberg off the Grand Banks of Newfoundland. By wireless telegraphy she sent out signals
of distress,
and several liners were near enough to catch and respond to the call.
</amazon:domain>
</speak>
```

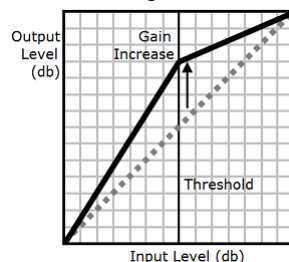
Adicionar compactação de intervalo dinâmico

<amazon:effect name="drc">

Essa tag é compatível com os formatos TTS padrão e neural.

Dependendo do texto, do idioma e da voz usados em um arquivo de áudio, os sons variam de baixo a alto. Os sons ambientais, como o som de um veículo em movimento, muitas vezes podem ocultar os sons mais baixos, o que dificulta ouvir a faixa de áudio com clareza. Para aumentar o volume de determinados sons no seu arquivo de áudio, use a tag de compactação de alcance dinâmico (drc).

A tag drc define um limite de "intensidade" de médio alcance para seu áudio e aumenta o volume (o ganho) dos sons em torno desse limite. Ele aplica o maior aumento de ganho mais próximo do limiar, e o aumento de ganho é diminuído muito além do limiar.



Isso facilita ouvir os sons de meio alcance em um ambiente com barulho, o que aumenta a clareza do arquivo de áudio todo.

A tag `drc` é um parâmetro booleano (presente ou ausente). Ela usa a sintaxe: `<amazon:effect name="drc">` e é fechada com `</amazon:effect>`.

Você pode usar `odrcTag` com qualquer voz ou idioma compatível com o Amazon Polly. Você pode aplicá-la a uma seção inteira da gravação ou em apenas algumas palavras. Por exemplo:

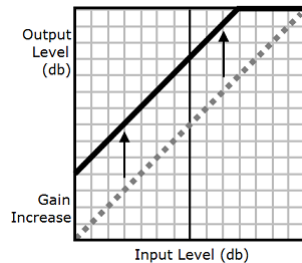
```
<speak>
  Some audio is difficult to hear in a moving vehicle, but <amazon:effect name="drc">
    this audio
  is less difficult to hear in a moving vehicle.</amazon:effect>
</speak>
```

Note

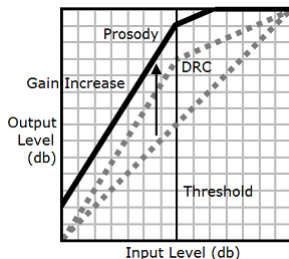
Quando você usa "drc" na sintaxe, há distinção entre letras maiúsculas e minúsculas. `amazon:effect`

Usar `drc` com a tag `prosody volume`

Assim como mostra o gráfico a seguir, a tag `prosody volume` aumenta uniformemente o volume de um arquivo de áudio inteiro a partir do nível original (linha pontilhada) até um nível ajustado (linha contínua). Para aumentar ainda mais o volume de certas partes do arquivo, use a tag `drc` com a tag `prosody volume`. Combinar tags não afeta as configurações da tag `prosody volume`.



Quando você usa `odrc` e `prosody volume` em conjunto, o Amazon Polly aplica o `odrc` primeiro, aumentando os sons de alcance médio (aqueles próximos ao limite). Em seguida, ele aplica a tag `prosody volume` e aumenta ainda mais o volume da faixa de áudio toda de forma uniforme.



Para usar as tags em conjunto, aninhe uma dentro da outra. Por exemplo:

```
<speak>
  <prosody volume="loud">This text needs to be understandable and loud. <amazon:effect
    name="drc">
    This text also needs to be more understandable in a moving car.</amazon:effect></
  prosody>
</speak>
```

Neste texto, a tag `prosody volume` aumenta o volume do trecho inteiro para "alto". A tag `drc` aprimora o volume de valores de alcance médio na segunda frase.

Note

Ao usar as tags `drc` e `prosody volume` em conjunto, use as práticas de XML padrão para agrupar tags.

Falar suavemente

```
<amazon:effect phonation="soft">
```

No momento, essa tag é compatível somente com o formato TTS padrão.

Para especificar que o texto de entrada deve ser falado em uma voz mais suave do que o normal, use a tag `<amazon:effect phonation="soft">`.

Neste caso, usa-se sintaxe:

```
<amazon:effect phonation="soft">text</amazon:effect>
```

Por exemplo, você pode usar essa tag com a voz de Matthew da seguinte forma:

```
<speak>
  This is Matthew speaking in my normal voice. <amazon:effect phonation="soft">This
  is Matthew speaking in my softer voice.</amazon:effect>
</speak>
```

Controlar o timbre

```
<amazon:effect vocal-tract-length>
```

No momento, essa tag é compatível somente com o formato TTS padrão.

Timbre é a qualidade tonal de uma voz que ajuda a distinguir a diferença entre as vozes, mesmo quando elas têm o mesmo tom e intensidade. Uma das características fisiológicas mais importantes que contribui com o timbre da fala timbre é o comprimento do trato vocal. O trato vocal é uma cavidade de ar que se estende desde o topo das pregas vocais até a borda dos lábios.

Para controlar o timbre da saída de fala no Amazon Polly, use o `vocal-tract-length` tag. Esta tag tem o efeito de alterar o comprimento do trato vocal do orador, que soa como uma alteração no tamanho do orador. Ao aumentar o `vocal-tract-length`, o orador soa fisicamente maior. Ao diminuí-lo, o orador soa menor. Você pode usar essa tag com qualquer uma das vozes do portfólio de texto para fala do Amazon Polly.

Para alterar o timbre, use os seguintes valores:

- `+n%` ou `-n%`: Ajusta comprimento do trato vocal por uma alteração percentual na voz atual. Por exemplo, `+4%` ou `-2%`. Os valores válidos variam de `+100%` a `-50%`. Valores fora deste intervalo são cortados. Por exemplo, `sons +111%`, como `+100%` e `-60%` soam como `-50%`.
- `n%`: altera o comprimento do trato vocal a uma porcentagem absoluta do comprimento do trato da voz atual. Por exemplo, `110%` ou `75%`. Um valor absoluto de `110%` é equivalente a um valor relativo de `+10%`. Um valor absoluto de `100%` é o mesmo que o valor padrão da voz atual.

O exemplo a seguir mostra como alterar o comprimento do trato vocal para alterar o timbre:

```
<speak>
  This is my original voice, without any modifications. <amazon:effect vocal-tract-
length="+15%">
  Now, imagine that I am much bigger. </amazon:effect> <amazon:effect vocal-tract-
length="-15%">
  Or, perhaps you prefer my voice when I'm very small. </amazon:effect> You can also
  control the
  timbre of my voice by making minor adjustments. <amazon:effect vocal-tract-
length="+10%">
  For example, by making me sound just a little bigger. </amazon:effect><amazon:effect
  vocal-tract-length="-10%"> Or, making me sound only somewhat smaller. </
amazon:effect>
</speak>
```

Combinar várias tags

Você pode combinar a `vocal-tract-length` Tag com qualquer outra tag SSML compatível com o Amazon Polly. Visto que o timbre (comprimento do trato vocal) e a altura estão intimamente conectados, você pode ter os melhores resultados usando ambas as tags `vocal-tract-length` e `<prosody pitch>`. Para produzir a voz mais realista, recomendamos que você use diferentes porcentagens de alterações para as duas tags. Experimente com várias combinações para obter os resultados desejados.

O exemplo a seguir mostra como combinar tags.

```
<speak>
  The pitch and timbre of a person's voice are connected in human speech.
  <amazon:effect vocal-tract-length="-15%"> If you are going to reduce the vocal tract
length,
  </amazon:effect><amazon:effect vocal-tract-length="-15%"> <prosody pitch="+20%"> you
  might consider increasing the pitch, too. </prosody></amazon:effect>
  <amazon:effect vocal-tract-length="+15%"> If you choose to lengthen the vocal tract,
  </amazon:effect> <amazon:effect vocal-tract-length="+15%"> <prosody pitch="-10%">
  you might also want to lower the pitch. </prosody></amazon:effect>
</speak>
```

Sussurrar

`<amazon:effect name="whispered">`

No momento, essa tag é compatível somente com o formato TTS padrão.

Essa tag indica que o texto de entrada deve ser falado em uma voz sussurrada, em vez da fala normal. Isso pode ser usado com qualquer uma das vozes do portfólio de texto para fala do Amazon Polly.

Neste caso, usa-se a seguinte sintaxe:

```
<amazon:effect name="whispered">text</amazon:effect>
```

Por exemplo:

```
<speak>
  <amazon:effect name="whispered">If you make any noise, </amazon:effect>
  she said, <amazon:effect name="whispered">they will hear us.</amazon:effect>
</speak>
```

Neste caso, a fala sintetizada dita pelo personagem será sussurrada, mas a frase "she said" será dita na fala normal sintetizada da voz do Amazon Polly selecionada.

É possível aprimorar o efeito "sussurro" diminuindo a velocidade da prosódia em até 10%, dependendo do efeito desejado.

Por exemplo:

```
< speak>
  When any voice is made to whisper, < amazon:effect name="whispered">
    < prosody rate="-10%">the sound is slower and quieter than normal speech
  < /prosody>< /amazon:effect>
< /speak>
```

Ao gerar marcas de fala para uma voz sussurrada, o fluxo de áudio também deverá incluir a voz sussurrada para garantir que as marcas de fala correspondam ao fluxo de áudio.

Gerenciamento de léxicos

Léxicos de pronúncia permitem que você personalize a pronúncia de palavras. O Amazon Polly fornece operações de API que você pode usar para armazenar léxicos em uma AWS região. Esses léxicos são específicos para essa região em particular. Você pode usar um ou mais dos léxicos dessa região quando sintetizar o texto usando a operação `SynthesizeSpeech`. Isso aplica o léxico específico ao texto de entrada antes de a síntese começar. Para obter mais informações, consulte [SynthesizeSpeech \(p. 235\)](#).

Note

Esses léxicos devem estar em conformidade com a recomendação PLS (Pronunciation Lexicon Specification, Especificação do léxico de pronúncia), do W3C. Para obter mais informações, consulte [Pronunciation Lexicon Specification \(PLS\) Version 1.0](#) no site do W3C.

Veja a seguir exemplos de maneiras de usar léxicos com mecanismos de síntese de fala:

- As palavras comuns são, às vezes, estilizadas com números tomando o lugar de letras, como em "g3t sm4rt" ("get smart", ou "fique esperto"). Seres humanos conseguem ler essas palavras corretamente. No entanto, um mecanismo de texto para fala (TTS) lê o texto literalmente, pronunciando a palavra exatamente como está escrita. É nesse ponto que você pode utilizar léxicos para personalizar a fala sintetizada usando o Amazon Polly. Neste exemplo, você pode especificar um alias (get smart) para a palavra "g3t sm4rt" no léxico.
- Seu texto pode incluir um acrônimo, como W3C. Você pode usar um léxico para definir um alias para a palavra W3C, de forma que ela seja lida no formato completo e expandido (World Wide Web Consortium).

Os léxicos oferecem a você mais controle sobre como o Amazon Polly pronuncia palavras incomuns no idioma selecionado. Por exemplo, você pode especificar a pronúncia usando um alfabeto fonético. Para obter mais informações, consulte [Pronunciation Lexicon Specification \(PLS\) Version 1.0](#) no site do W3C.

Tópicos

- [Aplicação de vários léxicos \(p. 131\)](#)
- [Gerenciamento de léxicos usando o Amazon Polly Console \(p. 132\)](#)
- [Gerenciamento de léxicos usando a AWS CLI \(p. 135\)](#)

Aplicação de vários léxicos

Você pode aplicar até cinco léxicos ao texto. Se o mesmo grafema aparecer em mais de um léxico aplicável ao seu texto, a ordem na qual eles são aplicados pode fazer a diferença na fala resultante. Por exemplo, no texto a seguir: "Hello, my name is Bob." e dois lexemas em diferentes léxicos nos quais ambos usem o grafema Bob.

LexA

```
<lexeme>
  <grapheme>Bob</grapheme>
  <alias>Robert</alias>
</lexeme>
```

LexB

```
<lexeme>
  <grapheme>Bob</grapheme>
  <alias>Bobby</alias>
</lexeme>
```

Se os léxicos estiverem listados na ordem LexA e depois LexB, a fala sintetizada será "Hello, my name is Robert". Se estiverem listados na ordem LexB e depois LexA, a fala sintetizada será "Hello, my name is Bobby".

Example – Aplicação do LexA antes do LexB

```
aws polly synthesize-speech \
--lexicon-names LexA LexB \
--output-format mp3 \
--text 'Hello, my name is Bob' \
--voice-id Justin \
bobAB.mp3
```

Resultado da fala: "Hello, my name is Robert."

Example – Aplicação do LexB antes do LexA

```
aws polly synthesize-speech \
--lexicon-names LexB LexA \
--output-format mp3 \
--text 'Hello, my name is Bob' \
--voice-id Justin \
bobBA.mp3
```

Resultado da fala: "Hello, my name is Bobby."

Para obter informações sobre a aplicação de léxicos usando o console do Amazon Polly, consulte [Aplicação dos léxicos usando o console \(sintetizar fala\)](#) (p. 133).

Gerenciamento de léxicos usando o Amazon Polly Console

Você pode usar o console do Amazon Polly para fazer o upload, download, aplicar, filtrar e excluir léxicos. Os procedimentos a seguir demonstram cada um desses processos.

Upload dos léxicos usando o console

Para usar um léxico de pronúncia, é preciso primeiro fazer o upload. Há dois locais no console do qual você pode fazer upload de um léxico: a guia Text-to-Speech e a guia Lexicons.

Os processos a seguir descrevem como adicionar léxicos que você pode usar para personalizar como palavras e frases incomuns ao idioma escolhido são pronunciadas.

Para adicionar um léxico pela guia Léxicos

1. Faça login noAWS Management Consolee abra o console do Amazon Polly em<https://console.aws.amazon.com/polly/>.

2. Escolha a guia Lexicons.
3. SelecioneFazer upload de léxico.
4. Forneça um nome para o léxico e, em seguida, useEscolha um arquivo léxicoPara encontrar o léxico para carregar. Você só pode carregar arquivos PLS com extensões.pls ou .xml.
5. SelecioneFazer upload de léxico. Se um léxico com o mesmo nome (seja arquivo .pls ou .xml) já existir, o upload do léxico substituirá o léxico existente.

Para adicionar um léxico pela guia Texto para fala

1. Faça login noAWS Management Consolee abra o console do Amazon Polly em<https://console.aws.amazon.com/polly/>.
2. Selecione a guia Text-to-Speech.
3. AmpliarConfigurações adicionaisAtive oPersonalizar a pronúnciaE, depois, escolhaFazer upload de léxico.
4. Forneça um nome para o léxico e, em seguida, useEscolha um arquivo léxicoPara encontrar o léxico para carregar. Você só pode usar arquivos PLS com extensões.pls ou .xml.
5. SelecioneFazer upload de léxico. Se um léxico com o mesmo nome (seja arquivo .pls ou .xml) já existir, o upload do léxico substituirá o léxico existente.

Aplicação dos léxicos usando o console (sintetizar fala)

O procedimento a seguir demonstra como aplicar um léxico ao seu texto inserido aplicando o léxico `w3c.pls` para substituir "W3C" por "World Wide Web Consortium". Se você aplicar vários léxicos ao texto, eles serão aplicados de cima para baixo, com a primeira correspondência levando precedência sobre as correspondências posteriores. O léxico só será aplicado ao texto se o idioma especificado nele for o mesmo que o idioma escolhido.

Você pode aplicar um léxico a uma entrada em texto sem formatação ou SSML.

Example – Aplicação do léxico W3C.pls

Para criar o léxico do qual você precisará para este exercício, consulte [Uso da operação PutLexicon \(p. 135\)](#). Use um editor de texto sem formatação para criar o léxico W3C.pls exibido na parte superior do tópico. Lembre-se do local onde salvou o arquivo.

Para aplicar o léxico W3C.pls à sua entrada

Neste exemplo, apresentamos um léxico para substituir "World Wide Web Consortium" por "W3C". Compare os resultados deste exercício com os de [Usar SSML \(Console\) \(p. 104\)](#) para inglês norte-americano e outro idioma.

1. Faça login noAWS Management Consolee abra o console do Amazon Polly em<https://console.aws.amazon.com/polly/>.
2. Execute um destes procedimentos:
 - Desligue oSSMLE digite ou cole este texto na caixa de entrada de texto.

```
He was caught up in the game.  
In the middle of the 10/3/2014 W3C meeting  
he shouted, "Score!" quite loudly.
```

- Ativar oSSMLE digite ou cole este texto na caixa de entrada de texto.

```
< speak>He wasn't paying attention.< break time="1s"/>
In the middle of the 10/3/2014 W3C meeting
he shouted, "Score!" quite loudly.</ speak>
```

3. DoldiomaLista, escolhaInglês dos EUA, depois, escolha a voz que você deseja usar para este texto.
4. AmpliarConfigurações adicionaisE ative oPersonalize a pronúncia.
5. Na lista de léxicos, selecione W3C (English, US).

Se o léxico W3C (English, US) não estiver listado, escolha Upload lexicon e carregue-o; em seguida, escolha-o na lista. Para criar esse léxico, consulte [Uso da operação PutLexicon \(p. 135\)](#).
6. Para ouvir a fala imediatamente, selecioneOuça..
7. Para salvar a fala em um arquivo,
 - a. Escolha Download.
 - b. Para alterar para um formato de arquivo diferente, ativeConfigurações de formato de arquivo de falaSelecione o formato de arquivo desejado e, depois, escolhaBaixar.

Repita as etapas anteriores, mas escolha um idioma diferente e observe a diferença na saída.

Filtro da lista de léxicos usando o console

O procedimento a seguir descreve como filtrar a lista de léxicos de forma que somente os léxicos de determinado idioma sejam exibidos.

Para filtrar os léxicos listados por idioma

1. Faça login noAWS Management Consolee abra o console do Amazon Polly em<https://console.aws.amazon.com/polly/>.
2. Escolha a guia Lexicons.
3. SelecioneQualquer idioma.
4. Na lista de idiomas, selecione o idioma que você deseja filtrar.

A lista exibe somente os léxicos para o idioma escolhido.

Download dos léxicos usando o console

O processo a seguir descreve como fazer o download de um ou mais léxicos. Você pode adicionar, remover ou modificar entradas do léxico no arquivo e, em seguida, carregá-las novamente para manter seu léxico atualizado.

Para baixar um ou mais léxicos

1. Faça login noAWS Management Consolee abra o console do Amazon Polly em<https://console.aws.amazon.com/polly/>.
2. Escolha a guia Lexicons.
3. Escolha os léxicos que você deseja baixar.
 - a. Para baixar um único léxico, escolha o nome na lista.
 - b. Para baixar vários léxicos como um único arquivo compactado, marque a caixa de seleção ao lado de cada entrada na lista que você deseja baixar.

4. Escolha Download.
5. Abra a pasta para onde você deseja fazer o download do léxico.
6. Escolha Salvar.

Exclusão de um léxico usando o Console

Para excluir um léxico

O processo a seguir descreve como excluir um léxico. Depois de excluir o léxico, você deve adicioná-lo de volta antes de usá-lo novamente. Você pode excluir um ou mais léxicos ao mesmo tempo marcando as caixas de seleção ao lado de cada léxico.

1. Faça login noAWS Management Consolee abra o console do Amazon Polly em<https://console.aws.amazon.com/polly/>.
2. Escolha a guia Lexicons.
3. Escolha um ou mais léxicos que você deseja excluir da lista.
4. Escolha Delete.
5. Insira o texto de confirmação e escolhaExcluirPara remover o léxico da região ouCancelarpara ficar com ele.

Gerenciamento de léxicos usando a AWS CLI

Os tópicos a seguir abrangem os comandos da AWS CLI necessários para gerenciar os léxicos de pronúncia.

Tópicos

- [Uso da operação PutLexicon \(p. 135\)](#)
- [Usando a operação GetLexicon \(p. 140\)](#)
- [Uso das operações ListLexicons \(p. 140\)](#)
- [Uso da operação DeleteLexicon \(p. 141\)](#)

Uso da operação PutLexicon

Com o Amazon Polly, você pode usar[PutLexicon \(p. 227\)](#)para armazenar léxicos de pronúncia em umaAWS região da sua conta. Em seguida, você pode especificar um ou mais desses léxico armazenados na sua solicitação [SynthesizeSpeech \(p. 235\)](#) que você deseja aplicar antes de o serviço começar a sintetizar o texto. Para obter mais informações, consulte [Gerenciamento de léxicos \(p. 131\)](#).

Esta seção fornece exemplos de léxicos e instruções passo a passo para armazená-los e testá-los.

Note

Esses léxicos devem estar em conformidade com a recomendação PLS (Pronunciation Lexicon Specification, Especificação do léxico de pronúncia), do W3C. Para obter mais informações, consulte [Pronunciation Lexicon Specification \(PLS\) Version 1.0](#) no site do W3C.

Exemplo 1: Léxico com um léxico

Considere o léxico compatível com a PLS do W3C a seguir.

```
<?xml version="1.0" encoding="UTF-8"?>
<lexicon version="1.0"
  xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon
    http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"
  alphabet="ipa"
  xml:lang="en-US">
  <lexeme>
    <grapheme>W3C</grapheme>
    <alias>World Wide Web Consortium</alias>
  </lexeme>
</lexicon>
```

Observe o seguinte:

- Os dois atributos especificados no elemento `<lexicon>`:
 - O `xml:lang` atributo especifica o código do idioma `en-US` ao qual o léxico se aplica. O Amazon Polly pode usar esse léxico de exemplo se a voz que você especificar no `SynthesizeSpeech` chamada tem o mesmo código de idioma (`en-US`).

Note

Você pode usar a operação `DescribeVoices` para localizar o código de idioma associado a uma voz.

- O atributo `alphabet` especifica IPA, o que significa que o Alfabeto Fonético Internacional (AFI; IPA na sigla em inglês, que significa "International Phonetic Alphabet") é usado para pronúncias. O AFI é um dos alfabetos para escrever pronúncias. O Amazon Polly também é compatível com Speech Assessment Methods Phonetic Alphabet (X-SAMPA, Alfabeto fon)
- O elemento `<lexeme>` descreve o mapeamento entre `<grapheme>` (ou seja, uma representação textual da palavra) e `<alias>`.

Para testar esse léxico, faça o seguinte:

1. Salve o léxico como `example.pls`.
2. Execute o comando da AWS CLI `put-lexicon` para armazenar o léxico (com o nome `w3c`), na região `us-east-2`.

```
aws polly put-lexicon \
--name w3c \
--content file://example.pls
```

3. Execute o comando `synthesize-speech` para sintetizar o texto de amostra em um fluxo de áudio (`speech.mp3`) e especifique parâmetro `lexicon-name` opcional.

```
aws polly synthesize-speech \
--text 'W3C is a Consortium' \
--voice-id Joanna \
--output-format mp3 \
--lexicon-names="w3c" \
speech.mp3
```

4. Reproduza o `speech.mp3` resultante e observe que a palavra `W3C` no texto é substituída por `World Wide Web Consortium`.

O léxico do exemplo anterior usa um alias. O AFI mencionado no léxico não é usado. O léxico a seguir especifica uma pronúncia fonética usando o elemento <phoneme> com o AFI.

```
<?xml version="1.0" encoding="UTF-8"?>
<lexicon version="1.0"
  xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon
    http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"
  alphabet="ipa"
  xml:lang="en-US">
  <lexeme>
    <grapheme>pecan</grapheme>
    <phoneme>p#k##n</phoneme>
  </lexeme>
</lexicon>
```

Siga as mesmas etapas para testar esse léxico. Coloque um texto que tenha a palavra "pecan" (por exemplo, "Pecan pie is delicious").

Exemplo 2: Léxico com vários léxicos

Neste exemplo, o lexema que você especificar no léxico se aplica exclusivamente ao texto de entrada para a síntese. Considere o seguinte léxico:

```
<?xml version="1.0" encoding="UTF-8"?>
<lexicon version="1.0"
  xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon
    http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"
  alphabet="ipa" xml:lang="en-US">

  <lexeme>
    <grapheme>W3C</grapheme>
    <alias>World Wide Web Consortium</alias>
  </lexeme>
  <lexeme>
    <grapheme>W3C</grapheme>
    <alias>WWW Consortium</alias>
  </lexeme>
  <lexeme>
    <grapheme>Consortium</grapheme>
    <alias>Community</alias>
  </lexeme>
</lexicon>
```

O léxico especifica três lexemas, dois dos quais definem um alias para o grafema W3C, da seguinte forma:

- O primeiro elemento <lexeme> define um alias (World Wide Web Consortium).
- O segundo <lexeme> define um alias alternativo (WWW Consortium).

O Amazon Polly usa a primeira substituição para determinado grafema em um léxico.

O terceiro <lexeme> define uma substituição (Community) para a palavra Consortium.

Primeiro, vamos testar esse léxico. Suponha que você deseja sintetizar o texto de exemplo a seguir para um arquivo de áudio (speech.mp3) e especifica o léxico em uma chamada para SynthesizeSpeech.

```
The W3C is a Consortium
```

O `SynthesizeSpeech` primeiro aplica a léxico da seguinte forma:

- De acordo com o primeiro lexema, a palavra W3C é revisada como World Wide Web Consortium. O texto revisado aparece da seguinte forma:

```
The World Wide Web Consortium is a Consortium
```

- O alias definido no terceiro lexema se aplica somente à palavra Consortium, que foi parte do texto original, resultando no texto a seguir:

```
The World Wide Web Consortium is a Community.
```

Você pode testar isso usando a AWS CLI, da seguinte forma:

1. Salve o léxico como `example.pls`.
2. Execute o comando `put-lexicon` para armazenar o léxico com o nome `w3c` na região `us-east-2`.

```
aws polly put-lexicon \  
--name w3c \  
--content file://example.pls
```

3. Execute o comando `list-lexicons` para verificar se o léxico `w3c` está na lista de léxicos retornados.

```
aws polly list-lexicons
```

4. Execute o comando `synthesize-speech` para sintetizar o texto de amostra para um arquivo de áudio (`speech.mp3`) e especifique parâmetro `lexicon-name` opcional.

```
aws polly synthesize-speech \  
--text 'W3C is a Consortium' \  
--voice-id Joanna \  
--output-format mp3 \  
--lexicon-names="w3c" \  
speech.mp3
```

5. Reproduza o arquivo `speech.mp3` resultante para verificar se a fala sintetizada reflete as alterações no texto.

Exemplo 3: Como especificar vários léxicos

Em uma chamada para `SynthesizeSpeech`, você pode especificar vários léxicos. Neste caso, o primeiro léxico especificado (da esquerda para a direita) sobrepõe quaisquer léxicos anteriores.

Considere os dois léxicos a seguir. Observe que cada léxico descreve aliases diferentes para o mesmo grafema de W3C.

- Léxico 1: `w3c.pls`

```
<?xml version="1.0" encoding="UTF-8"?>  
<lexicon version="1.0"  
  xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon  
    http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"  
  alphabet="ipa" xml:lang="en-US">
```



```
<lexeme>
  <grapheme>W3C</grapheme>
  <alias>World Wide Web Consortium</alias>
</lexeme>
</lexicon>
```

- Léxico 2: w3cAlternate.pls

```
<?xml version="1.0" encoding="UTF-8"?>
<lexicon version="1.0"
  xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon
    http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"
  alphabet="ipa" xml:lang="en-US">

  <lexeme>
    <grapheme>W3C</grapheme>
    <alias>WWW Consortium</alias>
  </lexeme>
</lexicon>
```

Vamos supor que você armazene esses léxicos como w3c e w3cAlternate, respectivamente. Se você especificar léxicos na ordem (w3c seguido por w3cAlternate) em uma chamada SynthesizeSpeech, o alias de W3C definido no primeiro léxico terá precedência sobre o segundo. Para testar os léxicos, faça o seguinte:

1. Salve os léxicos localmente em arquivos chamados w3c.pls e w3cAlternate.pls.
2. Faça upload desses léxicos usando o comando da AWS CLI put-lexicon.
 - Faça upload do léxico w3c.pls e armazene-o como w3c.

```
aws polly put-lexicon \
--name w3c \
--content file://w3c.pls
```

- Faça upload do léxico w3cAlternate.pls no serviço como w3cAlternate.

```
aws polly put-lexicon \
--name w3cAlternate \
--content file://w3cAlternate.pls
```

3. Execute o comando synthesize-speech para sintetizar texto de exemplo em um fluxo de áudio (speech.mp3) e especifique os dois léxicos usando o parâmetro lexicon-name.

```
aws polly synthesize-speech \
--text 'PLS is a W3C recommendation' \
--voice-id Joanna \
--output-format mp3 \
--lexicon-names '["w3c", "w3cAlternative"]' \
speech.mp3
```

4. Teste o speech.mp3 resultante. Ele deve ler da seguinte forma:

```
PLS is a World Wide Web Consortium recommendation
```

Amostras de código adicionais para a API PutLexicon

- Amostra de Java: [PutLexicon](#) (p. 150)
- Amostra de Python (Boto3): [PutLexicon](#) (p. 156)

Usando a operação GetLexicon

O Amazon Polly fornece a [GetLexicon](#) (p. 218) Para recuperar o conteúdo de um léxico de pronúncia armazenado em sua conta em uma região específica.

O seguinte comando da AWS CLI `get-lexicon` recupera o conteúdo do léxico `example`.

```
aws polly get-lexicon \  
--name example
```

Se você ainda não tiver um léxico armazenado em sua conta, pode usar a operação `PutLexicon` para armazenar um. Para obter mais informações, consulte [Uso da operação PutLexicon](#) (p. 135).

Veja a seguir uma resposta de exemplo. Além do léxico de conteúdo, a resposta retorna os metadados, como o código de idioma ao qual o léxico se aplica, número de lexemas definido no léxico, o nome de recurso da Amazon (ARN) do recurso e o tamanho do léxico em bytes. O valor `LastModified` é um timestamp do Unix.

```
{  
  "Lexicon": {  
    "Content": "lexicon content in plain text PLS format",  
    "Name": "example"  
  },  
  "LexiconAttributes": {  
    "LanguageCode": "en-US",  
    "LastModified": 1474222543.989,  
    "Alphabet": "ipa",  
    "LexemesCount": 1,  
    "LexiconArn": "arn:aws:polly:us-east-2:account-id:lexicon/example",  
    "Size": 495  
  }  
}
```

Amostras de código adicionais para a API GetLexicon

- Amostra de Java: [GetLexicon](#) (p. 149)
- Amostra de Python (Boto3): [GetLexicon](#) (p. 155)

Uso das operações ListLexicons

O Amazon Polly fornece a [ListLexicons](#) (p. 222) A operação de API, que você pode usar para obter a lista de léxicos de pronúncia na sua conta em um `AWSRegião` : A chamada da AWS CLI a seguir lista os léxicos na sua conta na região `us-east-2`.

```
aws polly list-lexicons
```

A seguir está um exemplo de resposta, mostrando dois léxicos de nome `w3c` e `tomato`. Para cada léxico, a resposta retorna metadados, como o código do idioma para o qual o léxico se aplica, o número de

lexemas definido no léxico, o tamanho em bytes, e assim por diante. O código de idioma descreve um idioma e uma localidade aos quais se aplicam os lexemas definidos no léxico.

```
{
  "Lexicons": [
    {
      "Attributes": {
        "LanguageCode": "en-US",
        "LastModified": 1474222543.989,
        "Alphabet": "ipa",
        "LexemesCount": 1,
        "LexiconArn": "arn:aws:polly:aws-region:account-id:lexicon/w3c",
        "Size": 495
      },
      "Name": "w3c"
    },
    {
      "Attributes": {
        "LanguageCode": "en-US",
        "LastModified": 1473099290.858,
        "Alphabet": "ipa",
        "LexemesCount": 1,
        "LexiconArn": "arn:aws:polly:aws-region:account-id:lexicon/tomato",
        "Size": 645
      },
      "Name": "tomato"
    }
  ]
}
```

Amostras de código adicionais para a API ListLexicon

- Amostra de Java: [ListLexicons](#) (p. 149)
- Amostra de Python (Boto3): [ListLexicon](#) (p. 156)

Uso da operação DeleteLexicon

O Amazon Polly fornece a [DeleteLexicon](#) (p. 213) Operação da API para excluir um léxico de pronúncia de uma AWS Região na sua conta. A AWS CLI a seguir exclui o léxico especificado.

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (\) no final de cada linha por um circunflexo (^) e use aspas completas (") em torno do texto de entrada por aspas simples (') para tags internas.

```
aws polly delete-lexicon \
--name example
```

Amostras de código adicionais para a API DeleteLexicon

- Amostra de Java: [DeleteLexicon](#) (p. 147)
- Amostra de Python (Boto3): [DeleteLexicon](#) (p. 154)

Criar arquivos de áudio longos

Para criar arquivos TTS para grandes passagens de texto, use a síntese assíncrona. Isso usa três APIs `SpeechSynthesisTask`:

- `StartSpeechSynthesisTask`: começa uma nova tarefa de síntese.
- `GetSpeechSynthesisTask`: retorna detalhes sobre uma tarefa de síntese enviada anteriormente.
- `ListSpeechSynthesisTasks`: lista todas as tarefas de síntese enviadas.

A operação `SynthesizeSpeech` produz áudio em tempo quase real, com latência relativamente baixa na maioria dos casos. Para fazer isso, a operação pode sintetizar somente 3000 caracteres.

O recurso de síntese assíncrona do Amazon Polly supera o desafio de processar um documento de texto maior alterando a maneira pela qual o documento é sintetizado e retornado. Quando uma solicitação de síntese é feita por meio do envio de texto de entrada usando o comando `StartSpeechSynthesisTask`, o Amazon Polly coloca as solicitações em fila e as processa de maneira assíncrona em segundo plano, assim que os recursos do sistema ficam disponíveis. Em seguida, o Amazon Polly faz upload do fluxo de fala ou de marcas de fala diretamente para o bucket do Amazon Simple Storage Service (Amazon S3) (obrigatório), e notifica você sobre a disponibilidade do arquivo concluído por seu tópico do SNS (opcional).

Dessa forma, todas as funcionalidades, exceto o processamento em tempo quase real, estão disponíveis para textos de até 100.000 caracteres faturáveis (ou 200.000 caracteres no total).

Para sintetizar um documento usando esse método, é necessário ter um bucket do Amazon S3 gravável no qual o arquivo de áudio possa ser salvo. Você poderá ser notificado quando o áudio sintetizado estiver pronto, fornecendo um identificador opcional do tópico do SNS. Quando a tarefa de síntese for concluída, o Amazon Polly publica uma mensagem nesse tópico. Essa mensagem também pode conter informações de erro úteis em casos em que a tarefa de síntese não tenha obtido êxito. Para fazer isso, verifique se o usuário que está criando a tarefa de síntese também pode publicar no tópico do SNS. Consulte a [documentação do Amazon SNS](#) para obter mais informações sobre como criar e inscrever-se em um tópico do SNS.

Criptografia

É possível armazenar o arquivo de saída em um formato criptografado no bucket do S3, se desejar. Para fazer isso, habilite a [criptografia do bucket do Amazon S3](#), que usa uma das mais fortes cifras de bloco disponíveis, o Advanced Encryption Standard de 256 bits (AES-256).

Tópicos

- [Configurar a política do IAM para a síntese assíncrona](#) (p. 142)
- [Criar arquivos de áudio longos \(console\)](#) (p. 143)
- [Criar arquivos de áudio longos \(CLI\)](#) (p. 144)

Configurar a política do IAM para a síntese assíncrona

Para usar a funcionalidade de síntese assíncrona, será necessária uma política do IAM que permita o seguinte:

- uso das novas operações do Amazon Polly
- gravar no bucket do S3 de saída

- publicar no tópico do SNS de status [opcional]

A política a seguir concede apenas as permissões necessárias exigidas para a síntese assíncrona e podem ser anexada ao usuário do IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "polly:StartSpeechSynthesisTask",
        "polly:GetSpeechSynthesisTask",
        "polly:ListSpeechSynthesisTasks"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::bucket-name/*"
    },
    {
      "Effect": "Allow",
      "Action": "sns:Publish",
      "Resource": "arn:aws:sns:region:account:topic"
    }
  ]
}
```

Criar arquivos de áudio longos (console)

É possível usar o console do Amazon Polly para criar falas longas usando a síntese assíncrona com a mesma funcionalidade que você pode usar com o AWS CLI. Isso é feito usando a guia Text-to-Speech (Texto para fala) da mesma maneira que qualquer outra síntese.

A outra funcionalidade de síntese assíncrona também está disponível pelo console. A guia S3 synthesis tasks (Tarefas de síntese do S3) reflete a funcionalidade `ListSpeechSynthesisTasks`, exibindo todas as tarefas salvas no bucket do S3 e permitindo filtrá-las, se desejar. Clicar em uma única tarefa específica mostra seus detalhes, refletindo a funcionalidade `GetSpeechSynthesisTask`.

Para sintetizar um texto grande usando o console do Amazon Polly do

1. Faça login no AWS Management Console e abra o console Amazon Polly em <https://console.aws.amazon.com/polly/>.
2. Selecione a guia Text-to-Speech.
3. com SSML ativado ou desativado, digite ou passe o texto na caixa de entrada.
4. Selecione o idioma, a região e a voz para seu texto.
5. Selecione Salvar no S3.

Note

Ambos os arquivos ficam esmaecidos se o comprimento do texto estiver acima dos limites do `SynthesizeSpeech` operação.

6. O console abre um formulário para que você possa escolher onde armazenar o arquivo de saída.
 - a. Preencha o nome do bucket do Amazon S3 de destino.

- b. Se preferir, preencha o prefixo de chaves da saída.

Note

A bucket do S3 de saída deve ser gravável.

- c. Se você deseja ser notificado quando a tarefa de síntese for concluída, forneça um identificador opcional do tópico do SNS.

Note

O SNS deve estar aberto para publicação pelo usuário atual do console para usar essa opção. Para obter mais informações, consulte [Amazon Simple Notification Service \(SNS\)](#)

- d. Selecione Salvar no S3.

Para recuperar informações sobre suas tarefas de síntese de fala

1. No console, selecione a guia S3 Synthesis Tasks (Tarefas de síntese do S3).
2. As tarefas são exibidas por ordem de data. Para filtrar as tarefas, por status, escolha Todos os status, em seguida, escolha o status a ser usado.
3. Para visualizar os detalhes de uma tarefa específica, selecione o Task ID (ID da tarefa) vinculado.

Criar arquivos de áudio longos (CLI)

Amazon Polly síntese assíncrona a funcionalidade usa três `SpeechSynthesisTask` APIs para trabalhar com grandes quantidades de texto:

- `StartSpeechSynthesisTask`: começa uma nova tarefa de síntese.
- `GetSpeechSynthesisTask`: retorna detalhes sobre uma tarefa de síntese enviada anteriormente.
- `ListSpeechSynthesisTasks`: lista todas as tarefas de síntese enviadas.

Sintetizar grandes quantidades de texto (**`StartSpeechSynthesisTask`**)

Quando quiser criar um arquivo de áudio maior do que pode criar com a operação `SynthesizeSpeech` em tempo real, use a operação `StartSpeechSynthesisTask`. Além dos argumentos necessários para a operação `SynthesizeSpeech`, `StartSpeechSynthesisTask` também exige o nome de um bucket do Amazon S3. Outros dois argumentos opcionais também estão disponíveis: um prefixo de chaves para o arquivo de saída e o ARN de um tópico do SNS, se quiser receber notificações de status sobre a tarefa.

- `OutputS3BucketName`: o nome do bucket do Amazon S3 para o qual deve ser feito o upload da síntese. Esse bucket deve estar na mesma região que o serviço do Amazon Polly. Além disso, o usuário do IAM que está sendo usado para fazer a chamada deve ter acesso ao bucket. [Obrigatório]
- `OutputS3KeyPrefix`: prefixo de chaves para o arquivo de saída. Use este parâmetro se deseja salvar o arquivo de saída da fala em uma chave personalizada como um diretório em seu bucket. [Optional]
- `SnsTopicArn`: o ARN do tópico do SNS a ser usado se quiser receber notificações sobre o status da tarefa. Esse tópico do SNS deve estar na mesma região que o serviço do Amazon Polly. Além disso, o usuário do IAM que está sendo usado para fazer a chamada deve ter acesso ao tópico. [Optional]

Por exemplo, o exemplo a seguir pode ser usado para executar `start-speech-synthesis-task` AWS CLI na região Leste dos EUA (Ohio):

O exemplo da AWS CLI a seguir está formatado para Unix, Linux e macOS. Para Windows, substitua o caractere de continuação Unix de barra invertida (`\`) no final de cada linha por um circunflexo (`^`) e use aspas completas (`"`) em torno do texto de entrada por aspas simples (`'`) para marcas de interiores.

```
aws polly start-speech-synthesis-task \  
  --region us-east-2 \  
  --endpoint-url "https://polly.us-east-2.amazonaws.com/" \  
  --output-format mp3 \  
  --output-s3-bucket-name your-bucket-name \  
  --output-s3-key-prefix optional/prefix/path/file \  
  --voice-id Joanna \  
  --text file://text_file.txt
```

Isso resultará em uma resposta semelhante a esta:

```
"SynthesisTask":  
{  
  "OutputFormat": "mp3",  
  "OutputUri": "https://s3.us-east-2.amazonaws.com/your-bucket-name/optional/prefix/  
path/file.<task_id>.mp3",  
  "TextType": "text",  
  "CreationTime": [...],  
  "RequestCharacters": [...],  
  "TaskStatus": "scheduled",  
  "TaskId": [task_id],  
  "VoiceId": "Joanna"  
}
```

A operação `start-speech-synthesis-task` retorna diversos campos novos:

- `OutputUri`: o local do seu arquivo de saída da fala.
- `TaskId`: um identificador exclusivo para a tarefa de síntese de fala gerado pelo Amazon Polly.
- `CreationTime`: um time stamp de quando a tarefa foi inicialmente enviada.
- `RequestCharacters`: o número de caracteres faturáveis na tarefa.
- `TaskStatus`: fornece informações sobre o status da tarefa enviada.

Quando a tarefa for enviada, o status inicial será exibido como `scheduled`. Quando o Amazon Polly começar a processar a tarefa, o status será alterado para `inProgress`, depois, para `completed` ou `failed`. Se a tarefa falhar, uma mensagem de erro será retornada ao chamar a operação `GetSpeechSynthesisTask` ou `ListSpeechSynthesisTasks`.

Quando a tarefa for concluída, o arquivo da fala estará disponível no local especificado em `OutputUri`.

Recuperar informações sobre sua tarefa de síntese de fala

É possível obter informações sobre uma tarefa, como erros, status e assim por diante, usando a operação `GetSpeechSynthesisTask`. Para fazer isso, você precisará do `task-id` retornado pelo `StartSpeechSynthesisTask`.

Por exemplo, o exemplo a seguir pode ser usado para executar o `get-speech-synthesis-task` AWS CLI Comando da :

```
aws polly get-speech-synthesis-task \  
  --region us-east-2 \  
  --endpoint-url "https://polly.us-east-2.amazonaws.com/" \  
  --task-id task identifier
```

Também é possível listar todas as tarefas de síntese de fala que você executou na região atual usando a operação `ListSpeechSynthesisTasks`.

Por exemplo, o exemplo a seguir pode ser usado para executar `olist-speech-synthesis-tasks` AWS CLI. Comando da :

```
aws polly list-speech-synthesis-tasks \  
--region us-east-2 \  
--endpoint-url "https:// polly.us-east-2.amazonaws.com/"
```


Exemplos de código e de aplicativo

Esta seção apresenta amostras de código e aplicativos de exemplo que você pode usar para explorar o Amazon Polly.

Tópicos

- [Código de exemplo \(p. 147\)](#)
- [Aplicações de exemplo \(p. 158\)](#)

O **Código de exemplo** O tópico contém fragmentos de código organizados por linguagem de programação e separados em exemplos de diferentes funcionalidades do Amazon Polly. O **Aplicativo de exemplo** O tópico contém aplicativos organizados por linguagem de programação que podem ser usados de modo independente para explorar o Amazon Polly.

Antes de começar a usar esses exemplos, recomendamos que você primeiro leia [Como funciona o Amazon Polly \(p. 3\)](#) e siga as etapas descritas em [Conceitos básicos da Amazon Polly \(p. 4\)](#).

Código de exemplo

Este tópico contém exemplos de código para várias funcionalidades que podem ser usadas para explorar o Amazon Polly.

Código de exemplo por linguagem de programação

- [Exemplos de Java \(p. 147\)](#)
- [Exemplos do Python \(p. 153\)](#)

Exemplos de Java

O código a seguir mostra como usar os aplicativos baseados em Java para realizar várias tarefas com o Amazon Polly. Esses exemplos não são exemplos completos, mas podem ser incluídos em aplicativos Java maiores que usam o [AWS SDK for Java](#).

Trechos de código

- [DeleteLexicon \(p. 147\)](#)
- [DescribeVoices \(p. 148\)](#)
- [GetLexicon \(p. 149\)](#)
- [ListLexicons \(p. 149\)](#)
- [PutLexicon \(p. 150\)](#)
- [StartSpeechSynthesisTask \(p. 151\)](#)
- [Marcas de fala \(p. 152\)](#)
- [SynthesizeSpeech \(p. 153\)](#)

DeleteLexicon

O seguinte exemplo de código Java mostra como usar aplicativos baseados em Java para excluir um léxico específico armazenado em uma [AWS Região](#) : Um léxico que foi excluído não está disponível para síntese de voz, nem pode ser recuperado usando as APIs [GetLexicon](#) ou [ListLexicon](#).

Para obter mais informações sobre esta operação, consulte a referência da API [DeleteLexicon](#).

```
package com.amazonaws.polly.samples;

import com.amazonaws.services.polly.AmazonPolly;
import com.amazonaws.services.polly.AmazonPollyClientBuilder;
import com.amazonaws.services.polly.model.DeleteLexiconRequest;

public class DeleteLexiconSample {
    private String LEXICON_NAME = "SampleLexicon";

    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();

    public void deleteLexicon() {
        DeleteLexiconRequest deleteLexiconRequest = new
DeleteLexiconRequest().withName(LEXICON_NAME);

        try {
            client.deleteLexicon(deleteLexiconRequest);
        } catch (Exception e) {
            System.err.println("Exception caught: " + e);
        }
    }
}
```

DescribeVoices

A amostra de código Java a seguir mostra como usar aplicativos baseados em Java para produzir uma lista de vozes que estão disponíveis para uso ao solicitar uma síntese de fala. Opcionalmente, você pode especificar um código de idioma para filtrar as vozes disponíveis. Por exemplo, se você especificar en-US, a operação retorna uma lista de todas as vozes disponíveis em inglês dos EUA.

Para obter mais informações sobre esta operação, consulte a referência da API [DescribeVoices](#).

```
package com.amazonaws.polly.samples;

import com.amazonaws.services.polly.AmazonPolly;
import com.amazonaws.services.polly.AmazonPollyClientBuilder;
import com.amazonaws.services.polly.model.DescribeVoicesRequest;
import com.amazonaws.services.polly.model.DescribeVoicesResult;

public class DescribeVoicesSample {
    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();

    public void describeVoices() {
        DescribeVoicesRequest allVoicesRequest = new DescribeVoicesRequest();
        DescribeVoicesRequest enUsVoicesRequest = new
DescribeVoicesRequest().withLanguageCode("en-US");

        try {
            String nextToken;
            do {
                DescribeVoicesResult allVoicesResult =
client.describeVoices(allVoicesRequest);
                nextToken = allVoicesResult.getNextToken();
                allVoicesRequest.setNextToken(nextToken);

                System.out.println("All voices: " + allVoicesResult.getVoices());
            } while (nextToken != null);

            do {
                DescribeVoicesResult enUsVoicesResult =
client.describeVoices(enUsVoicesRequest);
                nextToken = enUsVoicesResult.getNextToken();
                enUsVoicesRequest.setNextToken(nextToken);
            } while (nextToken != null);
        }
    }
}
```

```
        System.out.println("en-US voices: " + enUsVoicesResult.getVoices());
    } while (nextToken != null);
} catch (Exception e) {
    System.err.println("Exception caught: " + e);
}
}
```

GetLexicon

O seguinte exemplo de código Java mostra como usar aplicativos baseados em Java para produzir o conteúdo de um léxico específico de pronúncia armazenado em uma AWS Região :

Para obter mais informações sobre esta operação, consulte a referência da API [GetLexicon](#).

```
package com.amazonaws.polly.samples;

import com.amazonaws.services.polly.AmazonPolly;
import com.amazonaws.services.polly.AmazonPollyClientBuilder;
import com.amazonaws.services.polly.model.GetLexiconRequest;
import com.amazonaws.services.polly.model.GetLexiconResult;

public class GetLexiconSample {
    private String LEXICON_NAME = "SampleLexicon";

    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();

    public void getLexicon() {
        GetLexiconRequest getLexiconRequest = new
        GetLexiconRequest().withName(LEXICON_NAME);

        try {
            GetLexiconResult getLexiconResult = client.getLexicon(getLexiconRequest);
            System.out.println("Lexicon: " + getLexiconResult.getLexicon());
        } catch (Exception e) {
            System.err.println("Exception caught: " + e);
        }
    }
}
```

ListLexicons

O seguinte exemplo de código Java mostra como usar aplicativos baseados em Java para produzir uma lista de léxicos de pronúncia armazenados em uma AWS Região :

Para obter mais informações sobre esta operação, consulte a referência da API [ListLexicons](#).

```
package com.amazonaws.polly.samples;

import com.amazonaws.services.polly.AmazonPolly;
import com.amazonaws.services.polly.AmazonPollyClientBuilder;
import com.amazonaws.services.polly.model.LexiconAttributes;
import com.amazonaws.services.polly.model.LexiconDescription;
import com.amazonaws.services.polly.model.ListLexiconsRequest;
import com.amazonaws.services.polly.model.ListLexiconsResult;

public class ListLexiconsSample {
    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();

    public void listLexicons() {
```

```
ListLexiconsRequest listLexiconsRequest = new ListLexiconsRequest();

try {
    String nextToken;
    do {
        ListLexiconsResult listLexiconsResult =
client.listLexicons(listLexiconsRequest);
        nextToken = listLexiconsResult.getNextToken();
        listLexiconsRequest.setNextToken(nextToken);

        for (LexiconDescription lexiconDescription :
listLexiconsResult.getLexicons()) {
            LexiconAttributes attributes = lexiconDescription.getAttributes();
            System.out.println("Name: " + lexiconDescription.getName()
                + ", Alphabet: " + attributes.getAlphabet()
                + ", LanguageCode: " + attributes.getLanguageCode()
                + ", LastModified: " + attributes.getLastModified()
                + ", LexemesCount: " + attributes.getLexemesCount()
                + ", LexiconArn: " + attributes.getLexiconArn()
                + ", Size: " + attributes.getSize());
        }
    } while (nextToken != null);
} catch (Exception e) {
    System.err.println("Exception caught: " + e);
}
}
```

PutLexicon

O seguinte exemplo de código Java mostra como usar aplicativos baseados em Java para armazenar um léxico de pronúncia em uma AWS Região :

Para obter mais informações sobre esta operação, consulte a referência da API [PutLexicon](#).

```
package com.amazonaws.polly.samples;

import com.amazonaws.services.polly.AmazonPolly;
import com.amazonaws.services.polly.AmazonPollyClientBuilder;
import com.amazonaws.services.polly.model.PutLexiconRequest;

public class PutLexiconSample {
    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();

    private String LEXICON_CONTENT = "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\" +
        "<lexicon version=\"1.0\" xmlns=\"http://www.w3.org/2005/01/pronunciation-lexicon\" xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" \" +
        \"xsi:schemaLocation=\"http://www.w3.org/2005/01/pronunciation-lexicon http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd\" \" +
        \"alphabet=\"ipa\" xml:lang=\"en-US\">\" +
        "<lexeme><grapheme>test1</grapheme><alias>test2</alias></lexeme>\" +
        "</lexicon>";
    private String LEXICON_NAME = "SampleLexicon";

    public void putLexicon() {
        PutLexiconRequest putLexiconRequest = new PutLexiconRequest()
            .withContent(LEXICON_CONTENT)
            .withName(LEXICON_NAME);

        try {
            client.putLexicon(putLexiconRequest);
        } catch (Exception e) {
            System.err.println("Exception caught: " + e);
        }
    }
}
```

```
}  
}
```

StartSpeechSynthesisTask

O seguinte exemplo de código Java mostra como usar aplicativos baseados em Java para sintetizar uma fala longa (de até 100.000 caracteres cobrados) e armazená-la diretamente em um bucket do Amazon S3.

Para obter mais informações, consulte a referência da API [StartSpeechSynthesisTask](#).

```
package com.amazonaws.parrot.service.tests.speech.task;  
  
import com.amazonaws.parrot.service.tests.AbstractParrotServiceTest;  
import com.amazonaws.services.polly.AmazonPolly;  
import com.amazonaws.services.polly.model.*;  
import org.awaitility.Duration;  
  
import java.util.concurrent.TimeUnit;  
  
import static org.awaitility.Awaitility.await;  
  
public class StartSpeechSynthesisTaskSample {  
  
    private static final int SYNTHESIS_TASK_TIMEOUT_SECONDS = 300;  
    private static final AmazonPolly AMAZON_POLLY_CLIENT =  
        AmazonPollyClientBuilder.defaultClient();  
    private static final String PLAIN_TEXT = "This is a sample text to be synthesized.";  
    private static final String OUTPUT_FORMAT_MP3 = OutputFormat.Mp3.toString();  
    private static final String OUTPUT_BUCKET = "synth-books-buckets";  
    private static final String SNS_TOPIC_ARN = "arn:aws:sns:eu-  
west-2:123456789012:synthesize-finish-topic";  
    private static final Duration SYNTHESIS_TASK_POLL_INTERVAL = Duration.FIVE_SECONDS;  
    private static final Duration SYNTHESIS_TASK_POLL_DELAY = Duration.TEN_SECONDS;  
  
    public static void main(String... args) {  
        StartSpeechSynthesisTaskRequest request = new StartSpeechSynthesisTaskRequest()  
            .withOutputFormat(OUTPUT_FORMAT_MP3)  
            .withText(PLAIN_TEXT)  
            .withTextType(TextType.Text)  
            .withVoiceId(VoiceId.Amy)  
            .withOutputS3BucketName(OUTPUT_BUCKET)  
            .withSnsTopicArn(SNS_TOPIC_ARN)  
            .withEngine("neural");  
  
        StartSpeechSynthesisTaskResult result =  
            AMAZON_POLLY_CLIENT.startSpeechSynthesisTask(request);  
        String taskId = result.getSynthesisTask().getTaskId();  
  
        await().with()  
            .pollInterval(SYNTHESIS_TASK_POLL_INTERVAL)  
            .pollDelay(SYNTHESIS_TASK_POLL_DELAY)  
            .atMost(SYNTHESIS_TASK_TIMEOUT_SECONDS, TimeUnit.SECONDS)  
            .until(  
                () ->  
                getSynthesisTaskStatus(taskId).equals(TaskStatus.Completed.toString())  
            );  
    }  
  
    private static SynthesisTask getSynthesisTask(String taskId) {  
        GetSpeechSynthesisTaskRequest getSpeechSynthesisTaskRequest = new  
            GetSpeechSynthesisTaskRequest()  
                .withTaskId(taskId);  
        GetSpeechSynthesisTaskResult result  
            =AMAZON_POLLY_CLIENT.getSpeechSynthesisTask(getSpeechSynthesisTaskRequest);
```

```
        return result.getSynthesisTask();
    }

    private static String getSynthesisTaskStatus(String taskId) {
        GetSpeechSynthesisTaskRequest getSpeechSynthesisTaskRequest = new
        GetSpeechSynthesisTaskRequest()
            .withTaskId(taskId);
        GetSpeechSynthesisTaskResult result
        =AMAZON_POLLY_CLIENT.getSpeechSynthesisTask(getSpeechSynthesisTaskRequest);
        return result.getSynthesisTask().getTaskStatus();
    }
}
```

Marcas de fala

A amostra de código a seguir mostra como usar aplicativos baseados em Java para sintetizar marcas de fala para texto digitado. Essa funcionalidade usa a API `SynthesizeSpeech`.

Para obter mais informações sobre essa funcionalidade, consulte [Marcas de fala](#) (p. 96).

Para obter mais informações sobre a API, consulte a referência da API [SynthesizeSpeech](#).

```
package com.amazonaws.polly.samples;

import com.amazonaws.services.polly.AmazonPolly;
import com.amazonaws.services.polly.AmazonPollyClientBuilder;
import com.amazonaws.services.polly.model.OutputFormat;
import com.amazonaws.services.polly.model.SpeechMarkType;
import com.amazonaws.services.polly.model.SynthesizeSpeechRequest;
import com.amazonaws.services.polly.model.SynthesizeSpeechResult;
import com.amazonaws.services.polly.model.VoiceId;

import java.io.File;
import java.io.FileOutputStream;
import java.io.InputStream;

public class SynthesizeSpeechMarksSample {
    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();

    public void synthesizeSpeechMarks() {
        String outputFileName = "/tmp/speechMarks.json";

        SynthesizeSpeechRequest synthesizeSpeechRequest = new SynthesizeSpeechRequest()
            .withOutputFormat(OutputFormat.Json)
            .withSpeechMarkTypes(SpeechMarkType.Viseme, SpeechMarkType.Word)
            .withVoiceId(VoiceId.Joanna)
            .withText("This is a sample text to be synthesized.");

        try (FileOutputStream outputStream = new FileOutputStream(new
        File(outputFileName))) {
            SynthesizeSpeechResult synthesizeSpeechResult =
            client.synthesizeSpeech(synthesizeSpeechRequest);
            byte[] buffer = new byte[2 * 1024];
            int readBytes;

            try (InputStream in = synthesizeSpeechResult.getAudioStream()){
                while ((readBytes = in.read(buffer)) > 0) {
                    outputStream.write(buffer, 0, readBytes);
                }
            }
        }
    }
}
```

```
        } catch (Exception e) {  
            System.err.println("Exception caught: " + e);  
        }  
    }  
}
```

SynthesizeSpeech

A amostra de código Java a seguir mostra como usar aplicativos baseados em Java para sintetizar fala com textos mais curtos para processamento em tempo quase real.

Para obter mais informações, consulte a referência da API [SynthesizeSpeech](#).

```
package com.amazonaws.polly.samples;  
  
import com.amazonaws.services.polly.AmazonPolly;  
import com.amazonaws.services.polly.AmazonPollyClientBuilder;  
import com.amazonaws.services.polly.model.OutputFormat;  
import com.amazonaws.services.polly.model.SynthesizeSpeechRequest;  
import com.amazonaws.services.polly.model.SynthesizeSpeechResult;  
import com.amazonaws.services.polly.model.VoiceId;  
  
import java.io.File;  
import java.io.FileOutputStream;  
import java.io.InputStream;  
  
public class SynthesizeSpeechSample {  
    AmazonPolly client = AmazonPollyClientBuilder.defaultClient();  
  
    public void synthesizeSpeech() {  
        String outputFileName = "/tmp/speech.mp3";  
  
        SynthesizeSpeechRequest synthesizeSpeechRequest = new SynthesizeSpeechRequest()  
            .withOutputFormat(OutputFormat.Mp3)  
            .withVoiceId(VoiceId.Joanna)  
            .withText("This is a sample text to be synthesized.")  
            .withEngine("neural");  
  
        try (FileOutputStream outputStream = new FileOutputStream(new  
File(outputFileName))) {  
            SynthesizeSpeechResult synthesizeSpeechResult =  
client.synthesizeSpeech(synthesizeSpeechRequest);  
            byte[] buffer = new byte[2 * 1024];  
            int readBytes;  
  
            try (InputStream in = synthesizeSpeechResult.getAudioStream()){  
                while ((readBytes = in.read(buffer)) > 0) {  
                    outputStream.write(buffer, 0, readBytes);  
                }  
            }  
        } catch (Exception e) {  
            System.err.println("Exception caught: " + e);  
        }  
    }  
}
```

Exemplos do Python

O código a seguir mostra como usar os aplicativos baseados em Python (boto3) para realizar várias tarefas com o Amazon Polly. Esses exemplos não se propõem a ser exemplos completos, mas podem ser incluídos em aplicativos Python maiores que usam o [AWS SDK for Python \(Boto\)](#).

Fragmentos de código

- [DeleteLexicon](#) (p. 154)
- [GetLexicon](#) (p. 155)
- [ListLexicon](#) (p. 156)
- [PutLexicon](#) (p. 156)
- [StartSpeechSynthesisTask](#) (p. 157)
- [SynthesizeSpeech](#) (p. 158)

DeleteLexicon

O exemplo de código Python a seguir usa o AWS SDK for Python (Boto) para excluir um léxico na região especificada no seu AWSConfiguração do . O exemplo exclui somente o léxico especificado. Ele pede que você confirme se deseja continuar antes de realmente excluir o léxico.

O exemplo de código a seguir usa as credenciais padrão armazenadas no AWSArquivo de configuração do SDK. Para obter informações sobre como criar o arquivo de configuração, consulte [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\)](#) (p. 7).

Para obter mais informações sobre esta operação, consulte a referência da API [DeleteLexicon](#).

```
from argparse import ArgumentParser
from sys import version_info

from boto3 import Session
from botocore.exceptions import BotoCoreError, ClientError

# Define and parse the command line arguments
cli = ArgumentParser(description="DeleteLexicon example")
cli.add_argument("name", type=str, metavar="LEXICON_NAME")
arguments = cli.parse_args()

# Create a client using the credentials and region defined in the adminuser
# section of the AWS credentials and configuration files
session = Session(profile_name="adminuser")
polly = session.client("polly")

# Request confirmation
prompt = input if version_info >= (3, 0) else raw_input
proceed = prompt((u"This will delete the \"{0}\" lexicon,"
                  " do you want to proceed? [y,n]: ").format(arguments.name))

if proceed in ("y", "Y"):
    print(u"Deleting {0}...".format(arguments.name))

    try:
        # Request deletion of a lexicon by name
        response = polly.delete_lexicon(Name=arguments.name)
    except (BotoCoreError, ClientError) as error:
        # The service returned an error, exit gracefully
        cli.error(error)

    print("Done.")
else:
    print("Cancelled.")
```


GetLexicon

O código Python a seguir usa o AWS SDK for Python (Boto) para recuperar todos os léxicos armazenados em uma AWS Região : O exemplo aceita um nome de léxico como parâmetro de linha de comando e obtém somente esse léxico, imprimindo o caminho tmp onde foi salvo localmente.

O exemplo de código a seguir usa as credenciais padrão armazenadas no AWS Arquivo de configuração do SDK. Para obter informações sobre como criar o arquivo de configuração, consulte [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\) \(p. 7\)](#).

Para obter mais informações sobre esta operação, consulte a referência da API [GetLexicon](#).

```
from argparse import ArgumentParser
from os import path
from tempfile import gettempdir

from boto3 import Session
from botocore.exceptions import BotoCoreError, ClientError

# Define and parse the command line arguments
cli = ArgumentParser(description="GetLexicon example")
cli.add_argument("name", type=str, metavar="LEXICON_NAME")
arguments = cli.parse_args()

# Create a client using the credentials and region defined in the adminuser
# section of the AWS credentials and configuration files
session = Session(profile_name="adminuser")
polly = session.client("polly")

print(u"Fetching {0}..." .format(arguments.name))

try:
    # Fetch lexicon by name
    response = polly.get_lexicon(Name=arguments.name)
except (BotoCoreError, ClientError) as error:
    # The service returned an error, exit gracefully
    cli.error(error)

# Get the lexicon data from the response
lexicon = response.get("Lexicon", {})

# Access the lexicon's content
if "Content" in lexicon:
    output = path.join(gettempdir(), u"%s.pls" % arguments.name)
    print(u"Saving to %s..." % output)

    try:
        # Save the lexicon contents to a local file
        with open(output, "w") as pls_file:
            pls_file.write(lexicon["Content"])
    except IOError as error:
        # Could not write to file, exit gracefully
        cli.error(error)
else:
    # The response didn't contain lexicon data, exit gracefully
    cli.error("Could not fetch lexicons contents")

print("Done.")
```

ListLexicon

O exemplo de código Python a seguir usa o AWS SDK for Python (Boto) para listar os léxicos na sua conta na região especificada no AWSConfiguração do . Para obter informações sobre como criar o arquivo de configuração, consulte [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\) \(p. 7\)](#).

Para obter mais informações sobre esta operação, consulte a referência da API [ListLexicons](#).

```
import sys

from boto3 import Session
from botocore.exceptions import BotoCoreError, ClientError

# Create a client using the credentials and region defined in the adminuser
# section of the AWS credentials and configuration files
session = Session(profile_name="adminuser")
polly = session.client("polly")

try:
    # Request the list of available lexicons
    response = polly.list_lexicons()
except (BotoCoreError, ClientError) as error:
    # The service returned an error, exit gracefully
    print(error)
    sys.exit(-1)

# Get the list of lexicons in the response
lexicons = response.get("Lexicons", [])
print("{0} lexicon(s) found".format(len(lexicons)))

# Output a formatted list of lexicons with some of the attributes
for lexicon in lexicons:
    print((u" - {Name} ({Attributes[LanguageCode]}), "
          "{Attributes[LexemesCount]} lexeme(s)").format(**lexicon))
```

PutLexicon

O seguinte exemplo de código mostra como usar os aplicativos baseados em Python (boto3) para armazenar um léxico de pronúncia em um AWSRegião :

Para obter mais informações sobre esta operação, consulte a referência da API [PutLexicon](#).

Observe o seguinte:

- Você precisa atualizar o código fornecendo um nome de arquivo de léxico local e um nome de léxico armazenado.
- O exemplo pressupõe que você tenha arquivos de léxico em um subdiretório chamado pls. Você precisa atualizar o caminho conforme apropriado.

O exemplo de código a seguir usa as credenciais padrão armazenadas no AWSArquivo de configuração do SDK. Para obter informações sobre como criar o arquivo de configuração, consulte [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\) \(p. 7\)](#).

Para obter mais informações sobre esta operação, consulte a referência da API [PutLexicon](#).

```
from argparse import ArgumentParser

from boto3 import Session
from botocore.exceptions import BotoCoreError, ClientError
```

```
# Define and parse the command line arguments
cli = ArgumentParser(description="PutLexicon example")
cli.add_argument("path", type=str, metavar="FILE_PATH")
cli.add_argument("-n", "--name", type=str, required=True,
                 metavar="LEXICON_NAME", dest="name")
arguments = cli.parse_args()

# Create a client using the credentials and region defined in the adminuser
# section of the AWS credentials and configuration files
session = Session(profile_name="adminuser")
polly = session.client("polly")

# Open the PLS lexicon file for reading
try:
    with open(arguments.path, "r") as lexicon_file:
        # Read the pls file contents
        lexicon_data = lexicon_file.read()

        # Store the PLS lexicon on the service.
        # If a lexicon with that name already exists,
        # its contents will be updated
        response = polly.put_lexicon(Name=arguments.name,
                                     Content=lexicon_data)
except (IOError, BotoCoreError, ClientError) as error:
    # Could not open/read the file or the service returned an error,
    # exit gracefully
    cli.error(error)

print(u"The \"{0}\" lexicon is now available for use.".format(arguments.name))
```

StartSpeechSynthesisTask

O exemplo de código Python a seguir usa o AWS SDK for Python (Boto) para listar os léxicos na sua conta na região especificada no AWSConfiguração do . Para obter informações sobre como criar o arquivo de configuração, consulte [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\)](#) (p. 7).

Para obter mais informações, consulte a referência da API [StartSpeechSynthesisTask](#).

```
import boto3
import time

polly_client = boto3.Session(
    aws_access_key_id='',
    aws_secret_access_key='',
    region_name='eu-west-2').client('polly')

response = polly_client.start_speech_synthesis_task(VoiceId='Joanna',
                                                    OutputS3BucketName='synth-books-buckets',
                                                    OutputS3KeyPrefix='key',
                                                    OutputFormat='mp3',
                                                    Text='This is a sample text to be synthesized.',
                                                    Engine='neural')

taskId = response['SynthesisTask']['TaskId']

print("Task id is {}".format(taskId))

task_status = polly_client.get_speech_synthesis_task(TaskId = taskId)

print(task_status)
```

SynthesizeSpeech

O exemplo de código Python a seguir usa o AWS SDK for Python (Boto) para listar os léxicos na sua conta na região especificada no AWSConfiguração do . Para obter informações sobre como criar o arquivo de configuração, consulte [Etapa 3,1: Configurar a AWS Command Line Interface \(AWS CLI\) \(p. 7\)](#).

Para obter mais informações sobre a API, consulte a referência da API [SynthesizeSpeech](#).

```
import boto3

polly_client = boto3.Session(
    aws_access_key_id=,
    aws_secret_access_key=,
    region_name='us-west-2').client('polly')

response = polly_client.synthesize_speech(VoiceId='Joanna',
    OutputFormat='mp3',
    Text = 'This is a sample text to be synthesized.',
    Engine = 'neural')

file = open('speech.mp3', 'wb')
file.write(response['AudioStream'].read())
file.close()
```

Aplicações de exemplo

Esta seção contém exemplos adicionais, na forma de aplicativos de exemplo que podem ser usados para explorar o Amazon Polly.

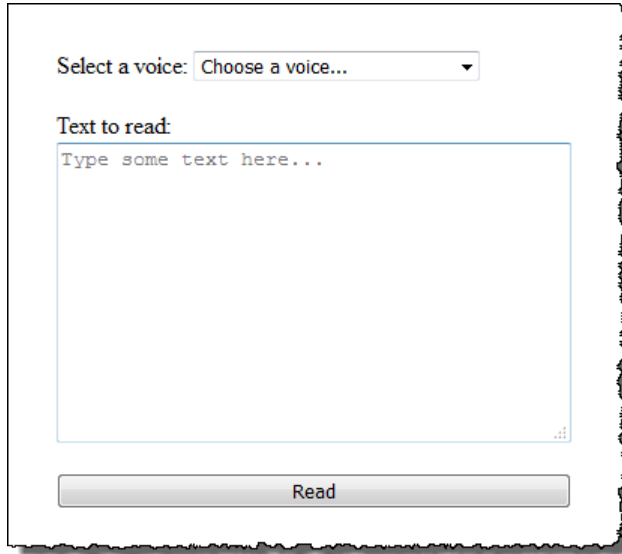
Aplicativos de exemplo por linguagem de programação

- [Exemplo Python \(cliente HTML5 e Python Server\) \(p. 158\)](#)
- [Exemplo de Java \(p. 168\)](#)
- [Exemplo de iOS \(p. 172\)](#)
- [Exemplo no Android \(p. 174\)](#)

Exemplo Python (cliente HTML5 e Python Server)

Esta aplicação de exemplo é formada pelo seguinte:

- Um servidor HTTP 1.1 usando a codificação de transferência HTTP (consulte a codificação de transferência em parte ("chunked") (consulte [Codificação de transferência em parte](#))
- Uma interface de usuário HTML5 simples que interage com o servidor HTTP 1.1 (exibido abaixo):



O objetivo deste exemplo é mostrar como usar o Amazon Polly para fazer o fluxo de fala a partir de uma aplicação HTML5 baseada no navegador. Consumir o fluxo de áudio produzido pelo Amazon Polly à medida que o texto é sintetizado é a abordagem recomendada para casos de uso em que a agilidade é um fator importante (por exemplo, sistemas de diálogo, leitores de tela, etc.).

Para executar essa aplicação de exemplo, é necessário o seguinte:

- Navegador da Web compatível com os padrões HTML5 e EcmaScript5 (por exemplo, Chrome 23.0 ou superior, Firefox 21.0 ou superior, Internet Explorer 9.0 ou superior)
- Python versão superior a 3.0

Para testar a aplicação

1. Salve o código do servidor como `server.py`. Para o código, consulte [Exemplo Python: Código do Python \(server.py\)](#) (p. 163).
2. Salve o código do cliente HTML5 como `index.html`. Para o código, consulte [Exemplo Python: Interface do usuário do HTML5 \(index.html\)](#) (p. 160).
3. Execute o comando a seguir a partir do caminho onde você salvou o `server.py` para iniciar a aplicação (em alguns sistemas, você pode precisar usar `python3` em vez de `python` ao executar o comando).

```
$ python server.py
```

Depois que a aplicação é iniciada, é exibido um URL no terminal.

4. Abra o URL mostrado no terminal em um navegador da web.

Você pode passar o endereço e a porta para o servidor de aplicações usar como parâmetro para `server.py`. Para obter mais informações, execute `python server.py -h`.

5. Para ouvir a fala, selecione uma voz na lista, digite um texto e, em seguida, escolha Read. A fala começa a ser reproduzida assim que o Amazon Polly transfere o primeiro bloco utilizável de dados de áudio.
6. Para interromper o servidor Python quando terminar de testar o aplicativo, pressione Ctrl+C no terminal onde o servidor está em execução.

Note

O servidor cria um cliente Boto3 usando o AWS SDK for Python (Boto). O cliente usa as credenciais armazenadas no AWS no seu computador para assinar e autenticar solicitações para o Amazon Polly. Para obter mais informações sobre como criar o AWS arquivo de configuração e armazenar credenciais, consulte [Configurar o AWS Command Line Interface](#) no AWS Command Line Interface Guia do usuário do.

Exemplo Python: Interface do usuário do HTML5 (index.html)

Esta seção fornece o código para o cliente HTML5 descrito em [Exemplo Python \(cliente HTML5 e Python Server\)](#) (p. 158).

```
<html>

<head>
  <title>Text-to-Speech Example Application</title>
  <script>
    /*
     * This sample code requires a web browser with support for both the
     * HTML5 and ECMAScript 5 standards; the following is a non-comprehensive
     * list of compliant browsers and their minimum version:
     *
     * - Chrome 23.0+
     * - Firefox 21.0+
     * - Internet Explorer 9.0+
     * - Edge 12.0+
     * - Opera 15.0+
     * - Safari 6.1+
     * - Android (stock web browser) 4.4+
     * - Chrome for Android 51.0+
     * - Firefox for Android 48.0+
     * - Opera Mobile 37.0+
     * - iOS (Safari Mobile and Chrome) 3.2+
     * - Internet Explorer Mobile 10.0+
     * - Blackberry Browser 10.0+
     */

    // Mapping of the OutputFormat parameter of the SynthesizeSpeech API
    // and the audio format strings understood by the browser
    var AUDIO_FORMATS = {
      'ogg_vorbis': 'audio/ogg',
      'mp3': 'audio/mpeg',
      'pcm': 'audio/wave; codecs=1'
    };

    /**
     * Handles fetching JSON over HTTP
     */
    function fetchJSON(method, url, onSuccess, onError) {
      var request = new XMLHttpRequest();
      request.open(method, url, true);
      request.onload = function () {
        // If loading is complete
        if (request.readyState === 4) {
          // if the request was successful
          if (request.status === 200) {
            var data;

            // Parse the JSON in the response
            try {
              data = JSON.parse(request.responseText);
            } catch (error) {
```

```
        onError(request.status, error.toString());
    }

    onSuccess(data);
} else {
    onError(request.status, request.responseText)
}
}

};

request.send();
}

/**
 * Returns a list of audio formats supported by the browser
 */
function getSupportedAudioFormats(player) {
    return Object.keys(AUDIO_FORMATS)
        .filter(function (format) {
            var supported = player.canPlayType(AUDIO_FORMATS[format]);
            return supported === 'probably' || supported === 'maybe';
        });
}

// Initialize the application when the DOM is loaded and ready to be
// manipulated
document.addEventListener("DOMContentLoaded", function () {
    var input = document.getElementById('input'),
        voiceMenu = document.getElementById('voice'),
        text = document.getElementById('text'),
        player = document.getElementById('player'),
        submit = document.getElementById('submit'),
        supportedFormats = getSupportedAudioFormats(player);

    // Display a message and don't allow submitting the form if the
    // browser doesn't support any of the available audio formats
    if (supportedFormats.length === 0) {
        submit.disabled = true;
        alert('The web browser in use does not support any of the' +
            ' available audio formats. Please try with a different' +
            ' one.');
```

```
    }

    // Play the audio stream when the form is submitted successfully
    input.addEventListener('submit', function (event) {
        // Validate the fields in the form, display a message if
        // unexpected values are encountered
        if (voiceMenu.selectedIndex <= 0 || text.value.length === 0) {
            alert('Please fill in all the fields.');
```

```
        } else {
            var selectedVoice = voiceMenu
                .options[voiceMenu.selectedIndex]
                .value;

            // Point the player to the streaming server
            player.src = '/read?voiceId=' +
                encodeURIComponent(selectedVoice) +
                '&text=' + encodeURIComponent(text.value) +
                '&outputFormat=' + supportedFormats[0];
            player.play();
        }

        // Stop the form from submitting,
        // Submitting the form is allowed only if the browser doesn't
        // support Javascript to ensure functionality in such a case
        event.preventDefault();
    });
});
```

```
    });

    // Load the list of available voices and display them in a menu
    fetchJSON('GET', '/voices',
    // If the request succeeds
    function (voices) {
        var container = document.createDocumentFragment();

        // Build the list of options for the menu
        voices.forEach(function (voice) {
            var option = document.createElement('option');
            option.value = voice['Id'];
            option.innerHTML = voice['Name'] + ' (' +
                voice['Gender'] + ', ' +
                voice['LanguageName'] + ')';
            container.appendChild(option);
        });

        // Add the options to the menu and enable the form field
        voiceMenu.appendChild(container);
        voiceMenu.disabled = false;
    },
    // If the request fails
    function (status, response) {
        // Display a message in case loading data from the server
        // fails
        alert(status + ' - ' + response);
    });
});

</script>
<style>
    #input {
        min-width: 100px;
        max-width: 600px;
        margin: 0 auto;
        padding: 50px;
    }

    #input div {
        margin-bottom: 20px;
    }

    #text {
        width: 100%;
        height: 200px;
        display: block;
    }

    #submit {
        width: 100%;
    }
</style>
</head>

<body>
    <form id="input" method="GET" action="/read">
        <div>
            <label for="voice">Select a voice:</label>
            <select id="voice" name="voiceId" disabled>
                <option value="">Choose a voice...</option>
            </select>
        </div>
        <div>
            <label for="text">Text to read:</label>
            <textarea id="text" maxlength="1000" minlength="1" name="text">
```



```
placeholder="Type some text here..."></textarea>
</div>
<input type="submit" value="Read" id="submit" />
</form>
<audio id="player"></audio>
</body>
</html>
```

Exemplo Python: Código do Python (server.py)

Esta seção fornece o código para o servidor Python descrito em [Exemplo Python \(cliente HTML5 e Python Server\)](#) (p. 158).

```
"""
Example Python 2.7+/3.3+ Application

This application consists of a HTTP 1.1 server using the HTTP chunked transfer
coding (https://tools.ietf.org/html/rfc2616#section-3.6.1) and a minimal HTML5
user interface that interacts with it.

The goal of this example is to start streaming the speech to the client (the
HTML5 web UI) as soon as the first consumable chunk of speech is returned in
order to start playing the audio as soon as possible.
For use cases where low latency and responsiveness are strong requirements,
this is the recommended approach.

The service documentation contains examples for non-streaming use cases where
waiting for the speech synthesis to complete and fetching the whole audio stream
at once are an option.

To test the application, run 'python server.py' and then open the URL
displayed in the terminal in a web browser (see index.html for a list of
supported browsers). The address and port for the server can be passed as
parameters to server.py. For more information, run: 'python server.py -h'
"""
from argparse import ArgumentParser
from collections import namedtuple
from contextlib import closing
from io import BytesIO
from json import dumps as json_encode
import os
import sys

if sys.version_info >= (3, 0):
    from http.server import BaseHTTPRequestHandler, HTTPServer
    from socketserver import ThreadingMixIn
    from urllib.parse import parse_qs
else:
    from BaseHTTPServer import BaseHTTPRequestHandler, HTTPServer
    from SocketServer import ThreadingMixIn
    from urlparse import parse_qs

from boto3 import Session
from botocore.exceptions import BotoCoreError, ClientError

ResponseStatus = namedtuple("HTTPStatus",
                           ["code", "message"])

ResponseData = namedtuple("ResponseData",
                          ["status", "content_type", "data_stream"])

# Mapping the output format used in the client to the content type for the
# response
```

```
AUDIO_FORMATS = {"ogg_vorbis": "audio/ogg",
                 "mp3": "audio/mpeg",
                 "pcm": "audio/wave; codecs=1"}
CHUNK_SIZE = 1024
HTTP_STATUS = {"OK": ResponseStatus(code=200, message="OK"),
               "BAD_REQUEST": ResponseStatus(code=400, message="Bad request"),
               "NOT_FOUND": ResponseStatus(code=404, message="Not found"),
               "INTERNAL_SERVER_ERROR": ResponseStatus(code=500, message="Internal server
error")}]
PROTOCOL = "http"
ROUTE_INDEX = "/index.html"
ROUTE_VOICES = "/voices"
ROUTE_READ = "/read"

# Create a client using the credentials and region defined in the adminuser
# section of the AWS credentials and configuration files
session = Session(profile_name="adminuser")
polly = session.client("polly")

class HTTPStatusError(Exception):
    """Exception wrapping a value from http.server.HTTPStatus"""

    def __init__(self, status, description=None):
        """
        Constructs an error instance from a tuple of
        (code, message, description), see http.server.HTTPStatus
        """
        super(HTTPStatusError, self).__init__()
        self.code = status.code
        self.message = status.message
        self.explain = description

class ThreadedHTTPServer(ThreadingMixIn, HTTPServer):
    """An HTTP Server that handle each request in a new thread"""
    daemon_threads = True

class ChunkedHTTPRequestHandler(BaseHTTPRequestHandler):
    """HTTP 1.1 Chunked encoding request handler"""
    # Use HTTP 1.1 as 1.0 doesn't support chunked encoding
    protocol_version = "HTTP/1.1"

    def query_get(self, queryData, key, default=""):
        """Helper for getting values from a pre-parsed query string"""
        return queryData.get(key, [default])[0]

    def do_GET(self):
        """Handles GET requests"""

        # Extract values from the query string
        path, _, query_string = self.path.partition('?')
        query = parse_qs(query_string)

        response = None

        print(u"[START]: Received GET for %s with query: %s" % (path, query))

        try:
            # Handle the possible request paths
            if path == ROUTE_INDEX:
                response = self.route_index(path, query)
            elif path == ROUTE_VOICES:
                response = self.route_voices(path, query)
```

```
elif path == ROUTE_READ:
    response = self.route_read(path, query)
else:
    response = self.route_not_found(path, query)

self.send_headers(response.status, response.content_type)
self.stream_data(response.data_stream)

except HTTPStatusError as err:
    # Respond with an error and log debug
    # information
    if sys.version_info >= (3, 0):
        self.send_error(err.code, err.message, err.explain)
    else:
        self.send_error(err.code, err.message)

    self.log_error(u"%s %s %s - [%d] %s", self.client_address[0],
                   self.command, self.path, err.code, err.explain)

print("[END]")

def route_not_found(self, path, query):
    """Handles routing for unexpected paths"""
    raise HTTPStatusError(HTTP_STATUS["NOT_FOUND"], "Page not found")

def route_index(self, path, query):
    """Handles routing for the application's entry point"""
    try:
        return ResponseData(status=HTTP_STATUS["OK"], content_type="text_html",
                             # Open a binary stream for reading the index
                             # HTML file
                             data_stream=open(os.path.join(sys.path[0],
                                                            path[1:]), "rb"))
    except IOError as err:
        # Couldn't open the stream
        raise HTTPStatusError(HTTP_STATUS["INTERNAL_SERVER_ERROR"],
                              str(err))

def route_voices(self, path, query):
    """Handles routing for listing available voices"""
    params = {}
    voices = []

    while True:
        try:
            # Request list of available voices, if a continuation token
            # was returned by the previous call then use it to continue
            # listing
            response = polly.describe_voices(**params)
        except (BotoCoreError, ClientError) as err:
            # The service returned an error
            raise HTTPStatusError(HTTP_STATUS["INTERNAL_SERVER_ERROR"],
                                  str(err))

        # Collect all the voices
        voices.extend(response.get("Voices", []))

        # If a continuation token was returned continue, stop iterating
        # otherwise
        if "NextToken" in response:
            params = {"NextToken": response["NextToken"]}
        else:
            break

    json_data = json_encode(voices)
    bytes_data = bytes(json_data, "utf-8") if sys.version_info >= (3, 0) \
```

```
        else bytes(json_data)

    return ResponseData(status=HTTP_STATUS["OK"],
                        content_type="application/json",
                        # Create a binary stream for the JSON data
                        data_stream=BytesIO(bytes_data))

def route_read(self, path, query):
    """Handles routing for reading text (speech synthesis)"""
    # Get the parameters from the query string
    text = self.query_get(query, "text")
    voiceId = self.query_get(query, "voiceId")
    outputFormat = self.query_get(query, "outputFormat")

    # Validate the parameters, set error flag in case of unexpected
    # values
    if len(text) == 0 or len(voiceId) == 0 or \
        outputFormat not in AUDIO_FORMATS:
        raise HTTPStatusError(HTTP_STATUS["BAD_REQUEST"],
                              "Wrong parameters")
    else:
        try:
            # Request speech synthesis
            response = polly.synthesize_speech(Text=text,
                                              VoiceId=voiceId,
                                              OutputFormat=outputFormat,
                                              Engine="neural")

        except (BotoCoreError, ClientError) as err:
            # The service returned an error
            raise HTTPStatusError(HTTP_STATUS["INTERNAL_SERVER_ERROR"],
                                  str(err))

        return ResponseData(status=HTTP_STATUS["OK"],
                            content_type=AUDIO_FORMATS[outputFormat],
                            # Access the audio stream in the response
                            data_stream=response.get("AudioStream"))

def send_headers(self, status, content_type):
    """Send out the group of headers for a successful request"""
    # Send HTTP headers
    self.send_response(status.code, status.message)
    self.send_header('Content-type', content_type)
    self.send_header('Transfer-Encoding', 'chunked')
    self.send_header('Connection', 'close')
    self.end_headers()

def stream_data(self, stream):
    """Consumes a stream in chunks to produce the response's output"""
    print("Streaming started...")

    if stream:
        # Note: Closing the stream is important as the service throttles on
        # the number of parallel connections. Here we are using
        # contextlib.closing to ensure the close method of the stream object
        # will be called automatically at the end of the with statement's
        # scope.
        with closing(stream) as managed_stream:
            # Push out the stream's content in chunks
            while True:
                data = managed_stream.read(CHUNK_SIZE)
                self.wfile.write(b"%X\r\n%s\r\n" % (len(data), data))

            # If there's no more data to read, stop streaming
            if not data:
                break
```

```
        # Ensure any buffered output has been transmitted and close the
        # stream
        self.wfile.flush()

        print("Streaming completed.")
    else:
        # The stream passed in is empty
        self.wfile.write(b"0\r\n\r\n")
        print("Nothing to stream.")

# Define and parse the command line arguments
cli = ArgumentParser(description='Example Python Application')
cli.add_argument(
    "-p", "--port", type=int, metavar="PORT", dest="port", default=8000)
cli.add_argument(
    "--host", type=str, metavar="HOST", dest="host", default="localhost")
arguments = cli.parse_args()

# If the module is invoked directly, initialize the application
if __name__ == '__main__':
    # Create and configure the HTTP server instance
    server = ThreadedHTTPServer((arguments.host, arguments.port),
                               ChunkedHTTPRequestHandler)
    print("Starting server, use <Ctrl-C> to stop...")
    print(u"Open {0}://{1}:{2}{3} in a web browser.".format(PROTOCOL,
                                                         arguments.host,
                                                         arguments.port,
                                                         ROUTE_INDEX))

    try:
        # Listen for requests indefinitely
        server.serve_forever()
    except KeyboardInterrupt:
        # A request to terminate has been received, stop the server
        print("\nShutting down...")
        server.socket.close()
```

Exemplo de Java

Este exemplo mostra como usar o Amazon Polly para fazer o fluxo de fala a partir de uma aplicação baseada em Java. O exemplo usa o [AWS SDK for Java](#) para ler o texto especificado usando uma voz selecionada entre uma lista.

O código mostrado abrange as principais tarefas, mas faz apenas a verificação mínima de erros. Se o Amazon Polly encontrar um erro, a aplicação será encerrada.

Para executar essa aplicação de exemplo, é necessário o seguinte:

- Java Development Kit (JDK) do Java 8
- [AWS SDK for Java](#)
- [Apache Maven](#)

Para testar a aplicação

1. Certifique-se de que a variável de ambiente JAVA_HOME está definida para o JDK.

Por exemplo, se você instalou o JDK 1.8.0_121 no Windows em C:\Program Files\Java\jdk1.8.0_121, deve digitar o seguinte no prompt de comando:

```
set JAVA_HOME="C:\Program Files\Java\jdk1.8.0_121"
```

Se você instalou o JDK 1.8.0_121 em Linux no /usr/lib/jvm/java8-openjdk-amd64, você deve digitar o seguinte no prompt de comando:

```
export JAVA_HOME=/usr/lib/jvm/java8-openjdk-amd64
```

2. Defina as variáveis do ambiente Maven para executar o Maven a partir da linha de comando.

Por exemplo, se você instalou Maven 3.3.9 no Windows em C:\Program Files\apache-maven-3.3.9, deve digitar o seguinte:

```
set M2_HOME="C:\Program Files\apache-maven-3.3.9"  
set M2=%M2_HOME%\bin  
set PATH=%M2%;%PATH%
```

Se você instalou o Maven 3.3.9 no Linux no /home/ec2-user/opt/apache-maven-3.3.9, deve digitar o seguinte:

```
export M2_HOME=/home/ec2-user/opt/apache-maven-3.3.9  
export M2=$M2_HOME/bin  
export PATH=$M2:$PATH
```

3. Crie um novo diretório chamado polly-java-demo.
4. No diretório polly-java-demo, crie um novo arquivo chamado pom.xml e cole o seguinte código dentro:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"  
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
          xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/  
maven-4.0.0.xsd">  
  <modelVersion>4.0.0</modelVersion>  
  <groupId>com.amazonaws.polly</groupId>  
  <artifactId>java-demo</artifactId>
```

```
<version>0.0.1-SNAPSHOT</version>

<dependencies>
<!-- https://mvnrepository.com/artifact/com.amazonaws/aws-java-sdk-polly -->
<dependency>
<groupId>com.amazonaws</groupId>
<artifactId>aws-java-sdk-polly</artifactId>
<version>1.11.77</version>
</dependency>
<!-- https://mvnrepository.com/artifact/com.googlecode.soundlibs/jlayer -->
<dependency>
<groupId>com.googlecode.soundlibs</groupId>
<artifactId>jlayer</artifactId>
<version>1.0.1-1</version>
</dependency>
</dependencies>
<build>
<plugins>
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>exec-maven-plugin</artifactId>
<version>1.2.1</version>
<executions>
<execution>
<goals>
<goal>java</goal>
</goals>
</execution>
</executions>
<configuration>
<mainClass>com.amazonaws.demos.polly.PollyDemo</mainClass>
</configuration>
</plugin>
</plugins>
</build>
</project>
```

5. Crie um novo diretório chamado polly em src/main/java/com/amazonaws/demos.
6. No diretório polly, crie um novo arquivo de código-fonte Java chamado PollyDemo.java e cole o seguinte código:

```
package com.amazonaws.demos.polly;

import java.io.IOException;
import java.io.InputStream;

import com.amazonaws.ClientConfiguration;
import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.polly.AmazonPollyClient;
import com.amazonaws.services.polly.model.DescribeVoicesRequest;
import com.amazonaws.services.polly.model.DescribeVoicesResult;
import com.amazonaws.services.polly.model.OutputFormat;
import com.amazonaws.services.polly.model.SynthesizeSpeechRequest;
import com.amazonaws.services.polly.model.SynthesizeSpeechResult;
import com.amazonaws.services.polly.model.Voice;

import javax.sound.sampled.AudioFormat;
import javax.sound.sampled.AudioInputStream;
import javax.sound.sampled.AudioSystem;
import javax.sound.sampled.DataLine;
import javax.sound.sampled.LineUnavailableException;
import javax.sound.sampled.SourceDataLine;
import javax.sound.sampled.TargetDataLine;
import javax.sound.sampled.UnsupportedAudioFileException;
```

```
public class PollyDemo {

    private final AmazonPollyClient polly;
    private final Voice voice;
    private static final String SAMPLE = "Congratulations. You have successfully built
this working demo
of Amazon Polly in Java. Have fun building voice enabled apps with Amazon Polly
(that's me!), and always
look at the AWS website for tips and tricks on using Amazon Polly and other great
services from AWS";

    public PollyDemo(Region region) {
        // create an Amazon Polly client in a specific region
        polly = new AmazonPollyClient(new DefaultAWSCredentialsProviderChain(),
        new ClientConfiguration());
        polly.setRegion(region);
        // Create describe voices request.
        DescribeVoicesRequest describeVoicesRequest = new DescribeVoicesRequest();

        // Synchronously ask Amazon Polly to describe available TTS voices.
        DescribeVoicesResult describeVoicesResult =
        polly.describeVoices(describeVoicesRequest);
        voice = describeVoicesResult.getVoices().get(0);
    }

    public InputStream synthesize(String text, OutputFormat format) throws IOException {
        SynthesizeSpeechRequest synthReq =
        new SynthesizeSpeechRequest().withText(text).withVoiceId(voice.getId())
        .withOutputFormat(format).withEngine("neural");
        SynthesizeSpeechResult synthRes = polly.synthesizeSpeech(synthReq);

        return synthRes.getAudioStream();
    }

    public static void main(String args[]) throws Exception {
        //create the test class
        PollyDemo helloWorld = new PollyDemo(Region.getRegion(Regions.US_EAST_1));
        //get the audio stream
        InputStream speechStream = helloWorld.synthesize(SAMPLE, OutputFormat.Mp3);

        //create an MP3 player
        AdvancedPlayer player = new AdvancedPlayer(speechStream,
        javax.sound.sampled.AudioSystem.getAudioDevice());

        player.setPlayBackListener(new PlaybackListener() {
            @Override
            public void playbackStarted(PlaybackEvent evt) {
                System.out.println("Playback started");
                System.out.println(SAMPLE);
            }

            @Override
            public void playbackFinished(PlaybackEvent evt) {
                System.out.println("Playback finished");
            }
        });

        // play it!
        player.play();
    }
}
```

7. Volte para o diretório `polly-java-demo` para limpar, compilar e executar a demonstração:


```
mvn clean compile exec:java
```

Exemplo de iOS

O exemplo a seguir usa o SDK do iOS para o Amazon Polly para ler o texto especificado usando uma voz selecionada entre uma lista de vozes.

O código exibido aqui abrange as principais tarefas, mas não lida com erros. Para obter o código completo, consulte [AWS Mobile SDK for iOS Demo do Amazon Polly](#).

Initialize

```
// Region of Amazon Polly.
let AwsRegion = AWSRegionType.usEast1

// Cognito pool ID. Pool needs to be unauthenticated pool with
// Amazon Polly permissions.
let CognitoIdentityPoolId = "YourCognitoIdentityPoolId"

// Initialize the Amazon Cognito credentials provider.
let credentialProvider = AWSCognitoCredentialsProvider(regionType: AwsRegion,
    identityPoolId: CognitoIdentityPoolId)

// Create an audio player
var audioPlayer = AVPlayer()
```

Obtenha a lista de vozes disponíveis

```
// Use the configuration as default
AWSServiceManager.default().defaultServiceConfiguration = configuration

// Get all the voices (no parameters specified in input) from Amazon Polly
// This creates an async task.
let task = AWSPolly.default().describeVoices(AWSPollyDescribeVoicesInput())

// When the request is done, asynchronously do the following block
// (we ignore all the errors, but in a real-world scenario they need
// to be handled)
task.continue(successBlock: { (awsTask: AWSTask) -> Any? in
    // awsTask.result is an instance of AWSPollyDescribeVoicesOutput in
    // case of the "describeVoices" method
    let voices = (awsTask.result! as AWSPollyDescribeVoicesOutput).voices

    return nil
})
```

Sintetize a fala

```
// First, Amazon Polly requires an input, which we need to prepare.
// Again, we ignore the errors, however this should be handled in
// real applications. Here we are using the URL Builder Request,
// since in order to make the synthesis quicker we will pass the
// presigned URL to the system audio player.
let input = AWSPollySynthesizeSpeechURLBuilderRequest()

// Text to synthesize
input.text = "Sample text"

// We expect the output in MP3 format
input.outputFormat = AWSPollyOutputFormat.mp3
```

```
// Choose the voice ID
input.voiceId = AWSPollyVoiceId.joanna

// Create an task to synthesize speech using the given synthesis input
let builder = AWSPollySynthesizeSpeechURLBuilder.default().getPreSignedURL(input)

// Request the URL for synthesis result
builder.continueOnSuccessWith(block: { (awsTask: AWSTask<NSURL>) -> Any? in
    // The result of getPreSignedURL task is NSURL.
    // Again, we ignore the errors in the example.
    let url = awsTask.result!

    // Try playing the data using the system AVAudioPlayer
    self.audioPlayer.replaceCurrentItem(with: AVPlayerItem(url: url as URL))
    self.audioPlayer.play()

    return nil
})
```

Exemplo no Android

O exemplo a seguir usa o SDK do Android para o Amazon Polly para ler o texto especificado usando uma voz selecionada entre uma lista de vozes.

O código exibido aqui abrange as principais tarefas, mas não lida com erros. Para obter o código completo, consulte o [AWS Mobile SDK for Android Demo do Amazon Polly](#).

Initialize

```
// Cognito pool ID. Pool needs to be unauthenticated pool with
// Amazon Polly permissions.
String COGNITO_POOL_ID = "YourCognitoIdentityPoolId";

// Region of Amazon Polly.
Regions MY_REGION = Regions.US_EAST_1;

// Initialize the Amazon Cognito credentials provider.
CognitoCachingCredentialsProvider credentialsProvider = new
    CognitoCachingCredentialsProvider(
        getApplicationContext(),
        COGNITO_POOL_ID,
        MY_REGION
    );

// Create a client that supports generation of presigned URLs.
AmazonPollyPresigningClient client = new AmazonPollyPresigningClient(credentialsProvider);
```

Obtenha a lista de vozes disponíveis

```
// Create describe voices request.
DescribeVoicesRequest describeVoicesRequest = new DescribeVoicesRequest();

// Synchronously ask Amazon Polly to describe available TTS voices.
DescribeVoicesResult describeVoicesResult = client.describeVoices(describeVoicesRequest);
List<Voice> voices = describeVoicesResult.getVoices();
```

Obtenha o URL para fluxo de áudio

```
// Create speech synthesis request.
SynthesizeSpeechPresignRequest synthesizeSpeechPresignRequest =
    new SynthesizeSpeechPresignRequest()
        // Set the text to synthesize.
        .withText("Hello world!")
        // Select voice for synthesis.
        .withVoiceId(voices.get(0).getId()) // "Joanna"
        // Set format to MP3.
        .withOutputFormat(OutputFormat.Mp3);

// Get the presigned URL for synthesized speech audio stream.
URL presignedSynthesizeSpeechUrl =
    client.getPresignedSynthesizeSpeechUrl(synthesizeSpeechPresignRequest);
```

Reproduzir fala sintetizada

```
// Use MediaPlayer: https://developer.android.com/guide/topics/media/mediaplayer.html
```

```
// Create a media player to play the synthesized audio stream.
MediaPlayer mediaPlayer = new MediaPlayer();
mediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);

try {
    // Set media player's data source to previously obtained URL.
    mediaPlayer.setDataSource(presignedSynthesizeSpeechUrl.toString());
} catch (IOException e) {
    Log.e(TAG, "Unable to set data source for the media player! " + e.getMessage());
}

// Prepare the MediaPlayer asynchronously (since the data source is a network stream).
mediaPlayer.prepareAsync();

// Set the callback to start the MediaPlayer when it's prepared.
mediaPlayer.setOnPreparedListener(new MediaPlayer.OnPreparedListener() {
    @Override
    public void onPrepared(MediaPlayer mp) {
        mp.start();
    }
});

// Set the callback to release the MediaPlayer after playback is completed.
mediaPlayer.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {
    @Override
    public void onCompletion(MediaPlayer mp) {
        mp.release();
    }
});
```

Aplicativos compatíveis com SAPI do Amazon Polly para Windows

Com o plug-in Amazon Polly para Windows, o Amazon Polly fornece um recurso de fala muito expandido para aplicativos Windows compatíveis com Speech Application Programming Interface (SAPI). A SAPI é uma API do Microsoft Windows que permite que aplicativos de desktop gerem fala.

O Windows fornece uma voz masculina e uma feminina que podem ser usadas com a API SAPI. Com o plug-in Amazon Polly para Windows, o aplicativo Windows se torna umAWSCiente, permitindo que você use todas as vozes do Amazon Polly, incluindo vozes padrão e neurais e o estilo de fala do locutor. O plug-in Amazon Polly para Windows oferece suporte apenas a aplicativos SAPI de 64 bits.

O Amazon Polly para Windows oferece suporte a todos os recursos do Amazon Polly, como Speech Synthesis Markup Language (SSML). Ao usar SSML com o plug-in, a funcionalidade padrão está disponível e os limites padrão são aplicáveis. Para obter mais informações sobre como usar SSML com o Amazon Polly, consulte [Geração de fala a partir de documentos SSML \(p. 102\)](#). Para obter mais informações sobre limites, consulte [Cotas no Amazon Polly \(p. 189\)](#).

Tópicos

- [Instalação e configuração do Amazon Polly para Windows \(SAPI\) \(p. 176\)](#)
- [Usando o Amazon Polly em aplicativos \(p. 178\)](#)

Instalação e configuração do Amazon Polly para Windows (SAPI)

Para instalar e configurar o plug-in Amazon Polly para Windows (SAPI), você precisa de umaAWSConta. Se você não tiver uma conta, consulte [Etapa 1.1: Cadastre-se no AWS \(p. 4\)](#).

Tópicos

- [Criar um usuário do IAM para oAWSCiente \(p. 176\)](#)
- [Instalar aAWS CLIpara Windows \(p. 177\)](#)
- [Criar um perfil para oAWSCiente \(p. 177\)](#)
- [Instalar o plug-in Amazon Polly para Windows \(p. 177\)](#)

Criar um usuário do IAM para oAWSCiente

Antes de conectar oAWSpara o seuAWSConta, você precisa criar um usuário do Identity and Access Management (IAM) para o cliente e anexar uma política de permissões a esse usuário. UmUsuário do IAMé uma pessoa ou aplicativo em umaAWSConta que pode fazer chamadas de API paraAWSProdutos .

OAWSO cliente usa o serviço Amazon Polly, portanto, ele precisa doAmazonPollyReadOnlyAccesspolítica de permissões.

Para criar um usuário do IAM

1. Faça login noAWS Management Consolee abra [oConsole do IAM](#).
2. Selecione Users (Usuários).

3. Escolha Add User (Adicionar usuário).
4. Em User name (Nome do usuário), insira `polly-windows-user`.
5. para oTipo de acesso, escolhaAcesso programáticoe, depois, escolhaPróximo: Permissions
6. Selecione Attach existing policies directly (Anexar políticas existentes diretamente).
7. Na caixa de pesquisa, insira `polly`.
8. Escolha AmazonPollyReadOnlyAccess.
9. Selecione Next (Próximo): Tags.
10. Selecione Next (Próximo): Análise.
11. Escolha Create User.
12. Registre o ID de chave de acesso e a chave de acesso secreta. Você precisa dessas informações para configurar oAWSCliente.

Important

Esta é a única vez em que você poderá acessar essas chaves. Por isso, lembre-se de tomar nota delas.

Instalar aAWS CLIpara Windows

OAWS Command Line Interface(AWS CLI) é uma ferramenta de código aberto que permite interagir com oAWSServiços que usam comandos do prompt de comando do Windows. Para instalar oAWS CLI, consulte[Instalar aAWS CLI no Windows](#)noAWS Command Line InterfaceGuia do usuário.

Criar um perfil para oAWSCliente

O plug-in Amazon Polly para Windows requer umAWSPerfil chamado`polly-windows`. Esse perfil garante que o mecanismo Amazon Polly use a conta correta.

Para criar o`polly-windows`Perfil doAWSCliente

1. No Windows, abra um prompt de comando.
2. No prompt de comando, execute o comando `aws configure --profile polly-windows`.
3. Quando for solicitado oAWSAccess Key IDeAWSSecret Access Key (Chave de acesso secreta), insira os valores que você salvou quando criou o usuário do IAM.
4. para oRegião padrão, insira o nome doAWSRegião em letras minúsculas.

Note

Nem todos os recursos e vozes do Amazon Polly estão disponíveis em todos osAWSRegiões. Para mais informações, consulte [Compatibilidade de recursos e regiões \(p. 90\)](#).

5. Para Default output format (Formato de saída padrão), pressione Enter.
6. No prompt de comando, verifique o perfil executando o comando `aws --profile polly-windows polly describe-voices`. Se você tiver criado o perfil com êxito, oAWS CLlexibe uma lista com os nomes disponíveis do Amazon Polly.

Instalar o plug-in Amazon Polly para Windows

Instalar o plug-in Amazon Polly para Windows baixando-o do Amazon Simple Storage Service (Amazon S3) e executando o instalador. Depois de instalar o plug-in, você pode usar vozes do Amazon Polly em todos os aplicativos Windows que usem a API SAPI.

Note

Se você estiver planejando usar uma voz neural no aplicativo SAPI, certifique-se de que o cliente esteja configurado para uma [AWS Região](#) que suporta NTTS antes de instalar o plugin Amazon Polly para Windows. Para mais informações, consulte [Compatibilidade de recursos e regiões](#) (p. 90).

Para instalar o plug-in Amazon Polly para Windows

1. No Windows, faça download do [plug-in](#).
2. Descompacte a pasta.
3. Executar `AmazonPollyForWindowsSetup.exe`.
4. No Control Panel (Painel de controle), procure `speech settings`.
5. UnderText-to-Speech, paraVoz, escolha uma voz Amazon Polly.
6. Escolha Preview Voice (Demonstrar voz).

Usando o Amazon Polly em aplicativos

Depois que você instalar o plug-in Amazon Polly para Windows, as vozes do Amazon Polly podem ser acessadas por qualquer aplicativo Windows que implemente o SAPI do Windows. Para usar uma voz, escolha-a na lista de vozes do Amazon Polly no aplicativo. Para obter uma lista completa de vozes disponíveis, consulte [Vozes disponíveis](#) (p. 12).

Para obter um exemplo de como usar vozes do Amazon Polly em um aplicativo Windows (Adobe Captivate), consulte [Usando o Amazon Polly em aplicativos Windows](#) no AWS Blog Machine Learning

Para usar vozes do Amazon Polly em um aplicativo Windows, você pode usar o PollyPlayer. PollyPlayerO é um aplicativo simples de sintetização de voz do Windows que é instalado quando você instala o plug-in Amazon Polly para Windows.

Para usar o PollyPlayer

1. No menu Iniciar do Windows, emAmazon Polly para Windows, escolhaPollyPlayer.
2. No aplicativo PollyPlayer, escolha uma voz e insira o texto que você deseja ouvir.



PollyPlayer

Pick a Voice:

Amazon Polly - Arabic - Zeina (Standard)
Amazon Polly - Australian English - Nicole (Standard)
Amazon Polly - Australian English - Russell (Standard)
Amazon Polly - Brazilian Portuguese - Ricardo (Standard)
Amazon Polly - Brazilian Portuguese - Vitoria (Standard)
Amazon Polly - British English - Amy (Neural)
Amazon Polly - British English - Amy (Standard)
Amazon Polly - British English - Brian (Neural)
Amazon Polly - British English - Brian (Standard)
Amazon Polly - British English - Emma (Neural)
Amazon Polly - British English - Emma (Standard)
Amazon Polly - Canadian French - Chantal (Standard)
Amazon Polly - Castilian Spanish - Conchita (Standard)
Amazon Polly - Castilian Spanish - Enrique (Standard)
Amazon Polly - Castilian Spanish - Lucia (Standard)
Amazon Polly - Chinese Mandarin - Zhiyu (Standard)
Amazon Polly - Danish - Mads (Standard)
Amazon Polly - Danish - Naja (Standard)
Amazon Polly - Dutch - Lotte (Standard)
Amazon Polly - Dutch - Ruben (Standard)
Amazon Polly - French - Celine (Standard)
Amazon Polly - French - Lea (Standard)
Amazon Polly - French - Mathieu (Standard)
Amazon Polly - German - Hans (Standard)

3. Para ouvir o texto, escolha Say It (Diga).

OAWSPlug-in do WordPress

Com o aAWSPlug-in do WordPress, você pode configurar vários serviços AWS, incluindo o Amazon Polly. Com o Amazon Polly, é possível fornecer gravações de áudio de seu conteúdo aos visitantes de seu site do WordPress. Use o plug-in para criar arquivos de áudio em qualquer uma das vozes e idiomas compatíveis com o Amazon Polly. Seus visitantes podem transmitir o áudio à vontade usando reprodutores de áudio em linha e aplicativos móveis.

É possível configurar o plug-in para fazer o seguinte:

- Criar gravações de áudio automaticamente para um novo conteúdo no momento da publicação, ou criar gravações para publicações individuais
- Criar gravações de áudio de seu conteúdo arquivado
- Usar o feed RSS do Amazon Pollycast para disponibilizar conteúdo de áudio em formato de podcast

Tópicos

- [Pré-requisitos de instalação \(p. 181\)](#)
- [Instalar e configurar o plug-in \(p. 183\)](#)
- [Personalizar a página do WordPress \(p. 185\)](#)
- [Podcasts com o Amazon Pollycast \(p. 187\)](#)
- [Como armazenar arquivos de áudio \(p. 188\)](#)

Pré-requisitos de instalação

Para usar o aAWS para WordPress plugin, você precisa de uma AWS, uma conta AWS Identity and Access Management (IAM) e um site do WordPress.

Tópicos

- [Criar uma conta da AWS \(p. 181\)](#)
- [Criação de um usuário do IAM \(p. 182\)](#)
- [Criar um site WordPress \(p. 183\)](#)

Criar uma conta da AWS

Se já tiver uma conta da AWS, ignore essa seção. Caso contrário, crie uma.

Para criar uma conta da AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga as instruções online.

Parte do procedimento de cadastro envolve uma chamada telefônica e a digitação de um código de verificação usando o teclado do telefone.

Criação de um usuário do IAM

Para usar aAWSPlug-in do WordPress do, você deve criar umUsuário do IAMpara o plug-in. Um usuário do IAM é uma pessoa ou aplicação em uma conta da AWS que tem permissão para fazer chamadas de API para produtos da AWS.

Note

Se você não usar o WordPress.com, mas tiver um site do WordPress auto-hospedado no Amazon EC2, poderá usar uma função do IAM em vez de um usuário do IAM. Para obter mais informações, consulte [Funções do IAM para o Amazon EC2](#) no Guia do usuário do Amazon EC2.

O procedimento a seguir contém as etapas para criar um IAMPolítica doe, em seguida, anexe-o ao usuário do IAM. Uma política do IAM é um documento que define as permissões que se aplicam a um usuário.

Para criar um usuário do IAM

1. Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação, selecione Policies (Políticas). Selecione Create policy (Criar política).
3. Selecione JSON.
4. Exclua tudo da caixa de texto de política e cole ou digite a seguinte política JSON:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "acm:DeleteCertificate",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:RequestedRegion": "us-east-1"
        }
      }
    }
  ]
}
```

5. Escolha Review policy (Revisar política).
6. Na página Review Policy (Revisar política), faça o seguinte:
 - a. Em Name (Nome), insira **AWSForWordPressDeleteCert**.
 - b. Selecione Create policy (Criar política).
7. No painel de navegação, escolha Usuários. Selecione Add user (Adicionar usuário).
8. Na página Set user details (Definir detalhes do usuário), faça o seguinte:
 - a. Em User name (Nome do usuário), digite **AWSForWordPressPlugin**.
 - b. Em Access type (Tipo de acesso), escolha Programmatic access (Acesso programático).
 - c. Selecione Next (Próximo): Permissions
9. Na página Set permissions (Definir permissões), faça o seguinte:
 - a. Selecione Attach existing policies directly.
 - b. Na caixa de pesquisa, digite **WordPress** e marque as caixas de seleção ao lado de AWSForWordPressPolicy e AWSForWordPressDeleteCert. Certifique-se de selecionar as caixas de seleção para ambas as políticas do WordPress.

Note

O `AWSForWordPressPolicy` é um `AWS` que concede ao usuário permissão para usar todos os recursos incluídos no `AWSPlug-in` do WordPress. Quando novos recursos forem adicionados ao plugin, a AWS atualizará essa política para incluir as permissões necessárias para usar os novos recursos.

- c. Selecione Next (Próximo): Tags.
10. Selecione Next (Próximo): Análise.
11. Escolha Criar usuário.
12. Escolha Download .csv (Fazer download de .csv) para salvar as credenciais do usuário (ID de chave de acesso e chave de acesso secreta) no computador. Você precisa delas para configurar o `AWSPlug-in` do WordPress.

Important

Esta é a única vez que você poderá salvar a chave de acesso secreta do usuário; portanto, salve-a agora.

Criar um site WordPress

Se você já tiver um site do WordPress, poderá avançar para [Instalar e configurar o plug-in \(p. 183\)](#).

Se você não tem um site WordPress, você pode criar um usando [WordPress.com](#). Para usar o `AWSPlug-in` do WordPress, você precisa de um plano de negócios ou de eCommerce do WordPress.com.

Você também pode instalar o software WordPress em seu próprio servidor Web, usando o Amazon Lightsail, o Amazon EC2 ou outra plataforma de hospedagem na Web. Hospedar seu próprio site WordPress envolve mais etapas do que usar o WordPress.com e requer a capacidade de configurar e gerenciar um servidor web, um load balancer, registros DNS e certificados de servidor web.

Independentemente de como você configurar seu site WordPress, você precisará do seguinte para poder usar o `AWSPlug-in` do WordPress:

- Seu site deve ter seu próprio nome de domínio. Um nome de domínio, também conhecido como endereço web ou URL (localizador uniforme de recursos), é o endereço que os visitantes usam para acessar seu site. Por exemplo, o nome de domínio da Amazon é `amazon.com`. Neste tópico, usamos `example.com` como nome de domínio de exemplo genérico, mas você precisa de um nome de domínio personalizado para seu site.
- Seu site deve funcionar usando HTTPS. Esta é uma prática recomendada de segurança e o plugin presume que seu site funciona usando HTTPS. Para verificar, acesse o endereço do seu site usando HTTPS (por exemplo, `https://example.com`) e verifique se seu site é exibido corretamente.

Quando seu site tiver um nome de domínio e trabalhar usando HTTPS, vá para a seção a seguir.

Instalar e configurar o plug-in

Antes de instalar o plugin, certifique-se de completar os [pré-requisitos \(p. 181\)](#).

Para instalar o plugin

1. Faça login no painel de administração do seu site WordPress, também conhecido como WP Admin.
2. Escolha Plugins.

3. Selecione Add New (Adicionar novo).
4. Na caixa de pesquisa, insira **AWS for WordPress**.
5. Encontre oAWSPara o WordPressPlug-in do . Escolha Install Now (Instalar agora) e selecione Activate (Ativar).

Configurar o plug-in

Depois de instalar oAWSPara o WordPress, configure-o para habilitar podcasting, locais de armazenamento alternativos e outras opções.

Note

No procedimento a seguir, os nomes de campo e comando podem diferir levemente dos nomes usados no WordPress.

Como configurar o plug-in

1. Na página WordPress Admin (Administração do WordPress, escolhaSettings (Configurações).
2. Configure o plug-in usando as seguintes opções:
 - AWSChave de acesso eAWSChave secreta—EstesAWSAs credenciais do plug-in permitem que o use o Amazon Polly e o Amazon Simple Storage Service (Amazon S3). Digite oAWSChaves de acesso e secreta criadas anteriormente. Se estiver hospedando seu site do WordPress no Amazon EC2, use uma função do IAM do em vez de credenciais. Nesse caso, deixe esses dois campos em branco.
 - Sample Rate—Taxa de exemplo dos arquivos de áudio que serão gerados, em Hz. Taxas de amostragem mais altas produzem áudio de melhor qualidade.
 - Nome da voz—Voz do Amazon Polly para usar no arquivo de áudio.

Note

Lembre-se de que a escolha da voz é determinada pelo idioma que você está usando. Para obter uma lista completa de vozes disponíveis, consulte [Vozes disponíveis \(p. 12\)](#).

- Posição do jogador—Onde posicionar o reprodutor de áudio no site. Você pode colocá-lo antes ou depois da publicação ou não o usar. Se quiser disponibilizar seus arquivos como podcasts usando o Amazon Pollycast, não exiba o reprodutor de áudio.
 - Novo post default—Especifica se o Amazon Polly deve criar um arquivo de áudio automaticamente para todas as novas publicações. Escolha essa opção se desejar que o Amazon Polly crie um arquivo de áudio para cada nova publicação.
 - Reprodução Automática—Especifica se o reprodutor de áudio deve começar a reproduzir o áudio automaticamente em uma publicação quando um usuário acessá-la.
 - Armazenar áudio no Amazon S3—Se você deseja armazenar arquivos de áudio em um bucket do S3Em vez de armazená-los no servidor web, escolha esta opção. O Amazon Polly cria o bucket para você. Para obter mais informações e a definição de preço, consulte[Amazon S3](#).
 - Amazon CloudFront Nome de domínio (CDN)—Se desejar transmitir seus arquivos de áudio com o Amazon CloudFront, forneça o nome do seu domínio do CloudFront. O plug-in usa o domínio para fazer fluxo de áudio. Se você ainda não tem um domínio, crie um no Amazon CloudFront.
 - iTunes category— a categoria do seu podcast. Escolher uma categoria permite que os usuários de podcast encontrem mais facilmente seu podcast no catálogo de podcasts.
 - iTunes explícita—Especifica se o recurso de podcasting do Amazon Pollycast deve ser ativado com a configuração explícita do iTunes.
 - Atualizar em massa todas as postagens—Se você quiser converter todas as suas publicações anteriores do WordPress para usar oAWSPara o WordPressplugin, escolha esta opção.
3. Selecione Save changes (Salvar alterações).

Personalizar a página do WordPress

Você pode usar várias opções para personalizar o conteúdo do WordPress para poder funcionar de forma mais eficaz com o plug-in do Amazon Polly do WordPress do:

- [Ajuste as configurações de plug-in para ajustar os arquivos de áudio \(p. 185\)](#)
- [Use SSML no seu conteúdo para modificar como ele será falado \(p. 185\)](#)
- [Use tags de Apenas áudio e Apenas palavras em seu conteúdo \(p. 185\)](#)

Ajuste as configurações de plug-in para ajustar os arquivos de áudio

OAWSPara o WordPressAs configurações do plug-in do têm três opções que podem ajudar você a personalizar o som do seu texto do WordPress para o arquivo de áudio:

- Nome da voz: O nome da voz e idioma selecionado permitem que você selecione o sexo da voz do Amazon Polly. Vozes específicas estão disponíveis para cada idioma, e você tem várias opções dentro de cada sexo em vários idiomas. Para obter mais informações, consulte [Vozes no Amazon Polly \(p. 12\)](#).
- Respirações automatizadas: Se habilitado, o Amazon Polly irá inserir automaticamente sons de respiração em seu conteúdo nos locais apropriados. Definir essa opção só permitirá que você use respirações automáticas e não defina manualmente. Para obter mais informações, consulte [respiração automática](#).
- Velocidade do áudio: Esta opção permite que você altere a velocidade de entrega da versão de áudio do seu conteúdo, que varia de 20% a 200% da velocidade padrão da voz.

Use SSML no seu conteúdo para modificar como ele será falado

O Amazon Polly em si é compatível com várias tags SSML que permitem que você controle muitos aspectos de como o Amazon Polly gera fala com base no texto que você fornece. Para obter mais informações sobre o SSML e o Amazon Polly, consulte [Tags SSML compatíveis \(p. 109\)](#). Algumas dessas tags são retomadas nas configurações de plug-in quando você o configura. Atualmente, no entanto, apenas a <break> tag está disponível para uso direto noAWSPara o WordPressPlug-in do .

A tag <break> permite que você adicione uma pausa na versão falada do texto. Você pode ajustar o comprimento da pausa para atender às suas necessidades específicas. O tamanho da pausa padrão é equivalente à pausa após uma vírgula. Para obter mais informações sobre a tag <break>, consulte [Tags SSML compatíveis \(p. 109\)](#).

Para usar tags SSML para aprimorar seu WordPress, a opção Habilitar suporte SSML do Amazon Polly deve ser selecionada naAdministrador do WordPress. A opção de Armazenar áudio no Amazon S3 também deve ser selecionada porque os arquivos de áudio com tags de SSML precisam ser armazenados em um bucket S3.

Use tags de Apenas áudio e Apenas palavras em seu conteúdo

Às vezes, você deseja adicionar algo ao seu podcast de áudio, mas não deseja que ele seja exibido no navegador. Ou você quer algo para mostrar no navegador, mas não deseja incluí-lo no arquivo de áudio.

Isso pode ser feito com as tags `Audio Only` e `Word Only`, que você coloca no conteúdo do WordPress para controlar qual parte do texto será exibido ou falado.

Para converter texto para áudio, mas não o exibir no navegador

1. Isole o texto selecionado na sua página do WordPress, colocando uma linha vazia acima e abaixo.
2. Na linha acima do texto selecionado, insira a seguinte tag:

```
-AMAZONPOLLY-ONLYAUDIO-START-
```

3. Na linha a seguir do texto, insira a seguinte tag:

```
-AMAZONPOLLY-ONLYAUDIO-END-
```

Você pode usar o mesmo procedimento para mostrar o texto no navegador sem incluir no arquivo de áudio usando as tags `-AMAZONPOLLY-ONLYWORDS-START-` e `-AMAZONPOLLY-ONLYWORDS-END-` da mesma maneira.

Por exemplo:

```
Initial text of your blog displayed in the browser and heard in the audio file.]
-AMAZONPOLLY-ONLYAUDIO-START-
[This part will not be displayed in the browser but will be heard in the audio file.]
-AMAZONPOLLY-ONLYAUDIO-END-
[Subsequent text of your blog displayed in the browser and heard in the audio file.]
```

e

```
[Initial text of your blog displayed in the browser and heard in the audio file.]
-AMAZONPOLLY-ONLYWORD-START-
This part will be displayed in the browser but will not be heard in the audio file.]
-AMAZONPOLLY-ONLYWORD-END-
Subsequent text of your blog displayed in the browser and heard in the audio file.
```

Adição de texto traduzido em sua postagem

OAWSPara o WordPressUse o Amazon Translate para criar versões traduzidas da sua postagem em um ou mais idiomas. Além do inglês, quatro idiomas estão disponíveis para esse serviço: Espanhol, francês, alemão e português. Os idiomas e as vozes desses idiomas podem ser configurados na página Settings (Configurações) na configuração do Amazon Translate.

Para traduzir sua postagem do WordPress em outros idiomas

1. Na página Adicionar nova postagem, crie e publique sua nova postagem do WordPress.
2. Na mesma página, verifique se a opção Habilitar o Amazon Polly é escolhido.
3. Para ver o custo aproximado da criação de arquivos de áudio em idiomas originais além de idiomas selecionados adicionalmente, escolha Quanto custará essa conversão? Selecione OK para voltar à página Adicionar nova postagem.
4. Escolha Traduzir

Para definir os idiomas para os quais você traduz sua postagem

1. Na página Settings (Configurações), em Amazon Translate configuration (Configuração do Amazon Translate), escolha o idioma da publicação na lista suspensa Source language (Idioma de origem).

2. Em Idiomas de destino, escolha os idiomas para os quais você deseja converter sua postagem.
3. Na lista suspensa Voz, escolha a voz que deseja usar para cada idioma selecionado.
4. Insira um rótulo para a escolha de idioma.
5. Escolha Save Changes (Salvar alterações).

Podcasts com o Amazon Pollycast

Com a AmazonFeeds de pollycast, seus visitantes podem ouvir seu conteúdo de áudio usando aplicativos padrão de podcasts. Os feeds do Pollycast compatíveis com a especificação RSS 2.0 fornecem os dados XML necessários para agregar podcasts por meio de aplicativos móveis de podcast populares, como o iTunes, e diretórios de podcast.

Quando você instala o WordPress plugin, selecione o iTunes explicitamente para adicionar automaticamente endpoints do Amazon Pollycast a todos os URLs de arquivo do WordPress. Isso permite que você distribua podcasts do site como um todo ou podcasts segmentados. Se você não escolheu a opção iTunes explicit (iTunes explícito) ao instalar o plug-in, siga estas etapas:

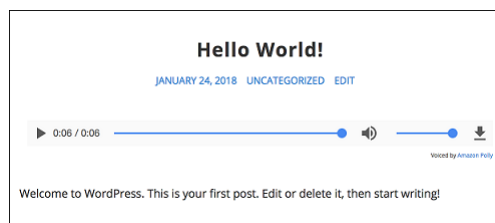
1. Na página WordPress Admin (Administração do WordPress, escolha Settings (Configurações).
2. Na página Settings (Configurações), selecione iTunes explicit (iTunes explícito).

É possível adicionar os endpoints do Amazon Pollycast manualmente, acrescentando `/amazon-pollycast/` ao URL de uma página em um aplicativo de podcast. Por exemplo:

```
example.com/amazon-pollycast/  
example.com/category/news/amazon-pollycast/  
example.com/author/john/amazon-pollycast/
```

Posicionar o reprodutor

Quando você instala o WordPress plugin do plug-in do, Ele exibe um reprodutor de áudio na parte superior do seu site do WordPress por padrão, a menos que você opte por exibi-lo abaixo do texto do site ou para não exibi-lo.



É possível reposicionar o reprodutor, removê-lo ou adicioná-lo (se você tiver optado por não exibi-lo) a qualquer momento.

Para reposicionar, remover ou adicionar o reprodutor ao seu site do WordPress

1. Na página WordPress Admin (Administração do WordPress, escolha Settings (Configurações).
2. No Configurações do Amazon Polly, para a Posição do jogador, escolha a opção apropriada.

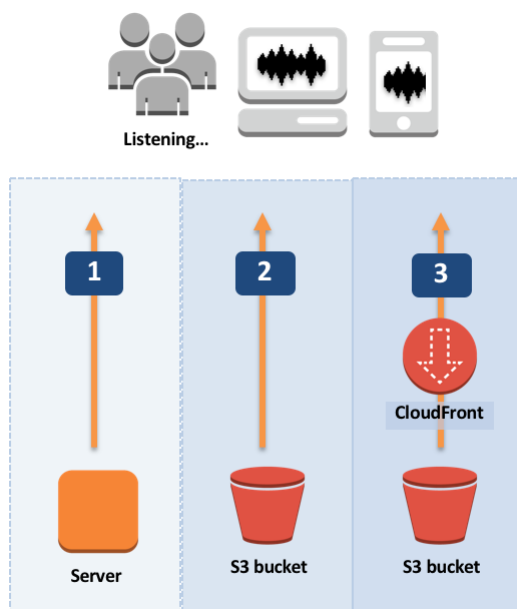
Para obter mais informações sobre como definir as opções de configuração, consulte [Configurar o plug-in \(p. 184\)](#).

Como armazenar arquivos de áudio

Quando você publica um conteúdo em seu site, ele é enviado ao Amazon Polly para síntese. Por padrão, o Amazon Polly armazena os novos arquivos de áudio em seu servidor web. Também é possível armazenar os arquivos no Amazon Simple Storage Service (Amazon S3) ou no Amazon CloudFront, que é uma rede de entrega de conteúdo (CDN) global.

Os usuários terão a mesma experiência de áudio, independentemente de onde você armazenar seus arquivos de áudio. Somente as alterações no local de transmissão:

1. Para arquivos de áudio armazenados no servidor do WordPress, os arquivos são transmitidos diretamente do servidor.
2. Para arquivos armazenados em um bucket do S3, os arquivos são transmitidos do bucket.
3. Se usar o CloudFront, os arquivos serão armazenados no Amazon S3 e transmitidos com o CloudFront.



É possível escolher onde armazenar seus arquivos ao instalar o plug-in do Amazon Polly.

Cotas no Amazon Polly

A seguir estão os limites sobre os quais você deve estar ciente ao usar o Amazon Polly.

Regiões compatíveis

Para uma lista de AWS Regiões onde o Amazon Polly está disponível, consulte [Endpoints e cotas do Amazon Polly](#) no Referência geral do Amazon Web Services.

As vozes neurais não estão disponíveis em todos os AWS Regiões. Para as Regiões que suportam vozes neurais, consulte [the section called “Compatibilidade de recursos e regiões”](#) (p. 90) para TTS neural.

Throttling

- Taxa de limitação por conta: 100 transações (solicitações ou operações) por segundo (tps) com um limite de intermitência de 120 tps.

Conexões simultâneas por conta: 90

- Taxa de controle por operação:

Operação	Limit
Vocabulário	
DeleteLexicon	Quaisquer 2 transações por segundo (tps) entre essas operações combinadas. Intermitência máxima permitida de 4 tps.
PutLexicon	
GetLexicon	
ListLexicons	
Fala	
DescribeVoices	80 tps com um limite de intermitência de 100 tps
SynthesizeSpeech	Voz padrão: 80 tps com um limite de intermitência de 100 tps Voz neural: 8 tps com um limite de intermitência de 10 tps
StartSpeechSynthesisTask	Voz padrão: 10 tps com um limite de intermitência de 12 tps Voz neural: 1 tps
GetSynthesizeSpeechTask e ListSynthesizeSpeechTask	Máximo permitido de 10 tps combinados

Léxicos de pronúncia

- Você pode armazenar até 100 léxicos por conta.

- Os nomes do léxico podem ser uma string alfanumérica de até 20 caracteres.
- Cada léxico pode ser de até 4.000 caracteres de tamanho. (Observe que o tamanho do léxico afeta a latência da operação SynthesizeSpeech.)
- Você pode especificar até 100 caracteres para a substituição de cada <phoneme> ou <alias> em um léxico.

Para obter mais informações sobre o uso de léxicos, consulte [Gerenciamento de léxicos \(p. 131\)](#).

Operação de API SynthesizeSpeech

Observe os seguintes limites relacionados ao uso da operação de API SynthesizeSpeech:

- O tamanho do texto de entrada pode ser de até 3000 caracteres cobrados (6000 caracteres no total). Tags SSML não são contadas como caracteres cobrados.
- Você pode especificar até cinco léxicos para aplicar ao texto de entrada.
- O fluxo de áudio de saída (síntese) é limitado a 10 minutos. Depois de ser atingido, qualquer fala restante será cortada.

Para mais informações, consulte [SynthesizeSpeech \(p. 235\)](#).

Note

Algumas limitações de operação de API SynthesizeSpeech podem ser ignoradas usando a operação de API StartSynthesizeSpeechTask. Para mais informações, consulte [Criar arquivos de áudio longos \(p. 142\)](#).

Operações de API SpeechSynthesisTask

Observe os seguintes limites relacionados ao uso das operações de API StartSpeechSynthesisTask, GetSpeechSynthesisTask e ListSpeechSynthesisTasks:

- O tamanho do texto de entrada pode ser de até 100,000 caracteres cobrados (200,000 caracteres no total). Tags SSML não são contadas como caracteres cobrados.
- Você pode especificar até cinco léxicos para aplicar ao texto de entrada.

Speech Synthesis Markup Language (SSML)

Observe os seguintes limites relacionados ao uso de SSML:

- Não há suporte para as tags <audio>, <lexicon>, <lookup> e <voice>.
- Os elementos <break> podem especificar uma duração máxima de 10 segundos cada.
- A tag <prosody> não suporta valores para o atributo de velocidade inferior a -80%.

Para mais informações, consulte [Geração de fala a partir de documentos SSML \(p. 102\)](#).

Segurança no Amazon Polly

A segurança da nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você se contará com um datacenter e uma arquitetura de rede criados para atender aos requisitos das organizações com as maiores exigências de segurança.

A segurança é uma responsabilidade compartilhada entre a AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem.

- Segurança da nuvem: a AWS é responsável pela proteção da infraestrutura que executa produtos da AWS na Nuvem AWS. A AWS também fornece serviços que podem ser usados com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos [Programas de conformidade da AWS](#). Para saber mais sobre os programas de conformidade que se aplicam ao Amazon Polly, consulte [AWS Serviços no escopo pelo programa de conformidade](#).
- Segurança da nuvem: sua responsabilidade é determinada pelo serviço da AWS que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o Amazon Polly. Os tópicos a seguir mostram como configurar o Amazon Polly para atender aos seus objetivos de segurança e de conformidade. Você também aprende a usar o [AWS Serviços](#) da que ajudam a monitorar e proteger os recursos do Amazon Polly.

Tópicos

- [Proteção de dados no Amazon Polly](#) (p. 191)
- [Identity and Access Management no Amazon Polly](#) (p. 192)
- [Registrar em log e monitorar no Amazon Polly](#) (p. 204)
- [Validação de conformidade com o Amazon Polly](#) (p. 204)
- [Resiliência no Amazon Polly](#) (p. 205)
- [Segurança da infraestrutura no Amazon Polly](#) (p. 205)
- [Melhores práticas de segurança para o Amazon Polly](#) (p. 205)

Proteção de dados no Amazon Polly

A Amazon Polly está em conformidade com a [AWS Modelo de responsabilidade compartilhada](#), que inclui as normas e diretrizes para proteção de dados. A AWS é responsável pela proteção da infraestrutura global que executa todas as [AWS Serviços](#) da .AWSO controle sobre os dados hospedados nessa infraestrutura, incluindo os controles de configuração de segurança para lidar com conteúdos e dados pessoais dos clientes. AWSOs clientes e os parceiros da APN, atuando como controladores de dados ou processadores de dados, são responsáveis por todos os dados pessoais que colocam na [AWS Nuvem](#).

Para fins de proteção de dados, recomendamos que você proteja a AWS e configure contas de usuário individuais com o [AWS Identity and Access Management \(IAM\)](#), para que cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos da AWS.
- Configure o registro em log das atividades da API e do usuário com o [AWS CloudTrail](#).
- Use as soluções de criptografia da AWS com todos os controles de segurança padrão nos produtos da AWS.

É altamente recomendável que você nunca coloque informações de identificação confidenciais, como números de conta dos seus clientes, em campos de formato livre, como um campo Name (Nome). Isso inclui quando você trabalha com a Amazon Polly ou outros AWS usando o console, API, AWS CLI, ou AWS SDKs. Todos os dados inseridos por você no Amazon Polly ou em outros serviços podem ser separados para inclusão em logs de diagnóstico. Ao fornecer um URL para um servidor externo, não inclua informações de credenciais no URL para validar a solicitação a esse servidor.

Para obter mais informações sobre proteção de dados, consulte a publicação [AWS Shared Responsibility Model and GDPR \(Modelo de responsabilidade compartilhada da AWS e GDPR\)](#) no Blog de segurança da AWS.

Criptografia em repouso

A saída da síntese de voz do Amazon Polly pode ser salva em seu próprio sistema. Também é possível chamar o Amazon Polly e criptografar o arquivo com qualquer chave de criptografia de sua escolha e armazená-lo no Amazon Simple Storage Service (Amazon S3) ou em outro armazenamento seguro. Amazon Polly [the section called “SynthesizeSpeech” \(p. 235\)](#) A operação do é stateless e não está associada a uma identidade do cliente. Não é possível recuperá-la do Amazon Polly posteriormente.

Criptografia em trânsito

Todos os envios de texto são protegidos por Secure Sockets Layer (SSL) durante o envio. A Amazon Polly não retém o conteúdo dos envios de texto.

Privacidade do tráfego entre redes

O acesso ao Amazon Polly é feito por meio do AWS CLI ou SDKs. As comunicações utilizam criptografia de sessão TLS (Transport Layer Security) para confidencialidade e [assinaturas digitais](#) para autenticação e integridade.

Identity and Access Management no Amazon Polly

O AWS Identity and Access Management (IAM) é um serviço da AWS que ajuda a controlar o acesso aos recursos da AWS de forma segura. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (ter permissões) para usar os recursos do Amazon Polly. O IAM é um serviço da AWS que pode ser usado sem custo adicional.

Tópicos

- [Audience \(p. 192\)](#)
- [Autenticação com identidades \(p. 193\)](#)
- [Gerenciamento do acesso usando políticas \(p. 195\)](#)
- [Como o Amazon Polly funciona com o IAM \(p. 196\)](#)
- [Exemplos de políticas baseadas em identidade do Amazon Polly \(p. 198\)](#)
- [Permissões da API do Amazon Polly: Referência de ações, permissões e recursos \(p. 203\)](#)

Audience

Como você usa o AWS Identity and Access Management (IAM) varia, dependendo do trabalho que você realiza no Amazon Polly.

Usuário do serviço— se você usar o serviço do Amazon Polly para fazer seu trabalho, o administrador fornecerá as credenciais e as permissões necessárias. À medida que mais recursos do Amazon Polly

forem usados para realizar o trabalho, talvez sejam necessárias permissões adicionais. Entender como o acesso é gerenciado ajuda a solicitar as permissões corretas ao administrador.

Administrador de serviços— se você for o responsável pelos recursos do Amazon Polly em sua empresa, você provavelmente terá acesso total ao Amazon Polly. É seu trabalho determinar quais recursos do Amazon Polly os funcionários devem acessar. Depois, envie solicitações ao administrador do IAM para alterar as permissões dos usuários do serviço. Revise as informações nesta página para entender os conceitos básicos do IAM. Para saber mais sobre como a empresa pode usar o IAM com a Amazon Polly, consulte [Como o Amazon Polly funciona com o IAM](#) (p. 196).

Administrador do IAM— se você é um administrador do IAM, talvez queira saber detalhes sobre políticas de escrita para gerenciar o acesso ao Amazon Polly. Para visualizar exemplos de políticas baseadas em identidade do Amazon Polly que podem ser usadas no IAM, consulte [Exemplos de políticas baseadas em identidade do Amazon Polly](#) (p. 198).

Autenticação com identidades

A autenticação é a forma como você faz login na AWS usando suas credenciais de identidade. Para obter mais informações sobre como fazer login usando o AWS Management Console, consulte [Login no AWS Management Console como usuário do IAM ou usuário root](#) no Manual do usuário do IAM.

É necessário estar autenticado (conectado à AWS) como o usuário root da Conta da AWS ou um usuário do IAM, ou ainda assumindo uma função do IAM. Também é possível usar a autenticação de logon único da sua empresa ou até mesmo fazer login usando o Google ou o Facebook. Nesses casos, o administrador configurou anteriormente federação de identidades usando funções do IAM. Ao acessar a AWS usando credenciais de outra empresa, você estará assumindo uma função indiretamente.

Para fazer login diretamente no [AWS Management Console](#), use sua senha com o e-mail do usuário root ou seu nome de usuário do IAM. É possível acessar a AWS de maneira programática usando chaves de acesso do AWS seu usuário root ou dos usuários do IAM. Se você não utilizar as ferramentas da AWS, você deverá assinar a solicitação por conta própria. Faça isso usando o Signature versão 4, um protocolo para autenticação de solicitações de API de entrada. Para obter mais informações sobre solicitações de autenticação, consulte [Processo de assinatura do Signature Version 4](#) na Referência geral da AWS.

Independentemente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, a AWS recomenda o uso da autenticação multifator (MFA) para aumentar a segurança de sua conta. Para saber mais, consulte [Uso da autenticação multifator \(MFA\) na AWS](#) no Manual do usuário do IAM.

Usuário root da Conta da AWS

Ao criar uma Conta da AWS, você começa com uma única identidade de login que tenha acesso total a todos os recursos e serviços da AWS na conta. Essa identidade é denominada usuário root da Conta da AWS e é acessada pelo login com o endereço de e-mail e a senha que você usou para criar a conta. Recomendamos que não use o usuário root para suas tarefas do dia a dia, nem mesmo as administrativas. Em vez disso, siga as [práticas recomendadas para o uso do usuário root somente a fim de criar seu primeiro usuário do IAM](#). Depois, armazene as credenciais do usuário raiz com segurança e use-as para executar somente algumas tarefas de gerenciamento de contas e de serviços.

Grupos e usuários do IAM

Um [usuário do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas para uma única pessoa ou aplicação. Um usuário do IAM pode ter credenciais de longo prazo, como um nome de usuário e uma senha ou um conjunto de chaves de acesso. Para saber como gerar chaves de acesso, consulte [Gerenciar chaves de acesso para usuários do IAM](#) no Manual do usuário do IAM. Ao gerar chaves de acesso para um usuário do IAM, visualize e salve o par de chaves de maneira segura. Não será possível recuperar a chave de acesso secreta futuramente. Em vez disso, você deverá gerar outro par de chaves de acesso.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e atribuir a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de funções. Um usuário é exclusivamente associado a uma pessoa ou a um aplicativo, mas uma função pode ser assumida por qualquer pessoa que precisar dela. Os usuários têm credenciais permanentes de longo prazo, mas as funções fornecem credenciais temporárias. Para saber mais, consulte [Quando criar um usuário do IAM \(em vez de uma função\)](#) no Manual do usuário do IAM.

Funções do IAM

Uma [função do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas. Ela é semelhante a um usuário do IAM, mas não está associada a uma pessoa específica. É possível assumir temporariamente uma função do IAM no AWS Management Console [alternando funções](#). É possível assumir uma função chamando uma operação de API da AWS CLI ou da AWS, ou usando um URL personalizado. Para mais informações sobre métodos para o uso de funções, consulte [Usar funções do IAM](#) no Manual do usuário do IAM.

As funções do IAM com credenciais temporárias são úteis nas seguintes situações:

- Permissões temporárias para usuários do IAM: um usuário do IAM pode assumir uma função do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- Acesso de usuário federado: em vez de criar um usuário do IAM, você poderá usar identidades de usuários existentes no AWS Directory Service, em seu diretório de usuários corporativos ou em um provedor de identidades da web. Estes são conhecidos como usuários federados. A AWS atribui uma função a um usuário federado quando o acesso é solicitado por meio de um [provedor de identidades](#). Para obter mais informações sobre usuários federados, consulte [Usuários federados e funções](#) no Manual do usuário do IAM.
- Acesso entre contas: é possível usar uma função do IAM para permitir que alguém (um principal confiável) em outra conta acesse recursos em sua conta. As funções são a principal forma de conceder acesso entre contas. No entanto, alguns serviços da AWS permitem que você anexe uma política diretamente a um recurso (em vez de usar uma função como proxy). Para saber a diferença entre funções e políticas baseadas em recurso para acesso entre contas, consulte [Como as funções do IAM diferem das políticas baseadas em recurso](#) no Manual do usuário do IAM.
- Acesso entre serviços: alguns serviços da AWS usam recursos em outros serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicações no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões do principal de chamada, usando uma função de serviço ou uma função vinculada ao serviço.
- Permissões de principal: ao usar um usuário ou uma função do IAM para executar ações na AWS, você é considerado um principal. As políticas concedem permissões a uma entidade principal. Quando você usa alguns serviços, pode executar uma ação que, em seguida, aciona outra ação em outro serviço. Nesse caso, você deve ter permissões para executar ambas as ações. Para ver se uma ação requer ações dependentes adicionais em uma política, consulte [Ações, recursos e chaves de condição do Amazon Polly](#) no Referência de autorização do serviço.
- Função de serviço: uma função de serviço é uma [função do IAM](#) que um serviço assume para realizar ações em seu nome. As funções de serviço fornecem acesso apenas dentro de sua conta e não podem ser usadas para conceder acesso a serviços em outras contas. Um administrador do IAM pode criar, modificar e excluir uma função de serviço do IAM. Para obter mais informações, consulte [Criação de uma função para delegar permissões a um serviço da AWS](#) no Manual do usuário do IAM.
- Função vinculada a serviço: uma função vinculada a serviço é um tipo de função de serviço vinculada a um serviço da AWS. O serviço pode assumir a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em sua conta do IAM e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para funções vinculadas ao serviço.

- Aplicações em execução no Amazon EC2: é possível usar uma função do IAM para gerenciar credenciais temporárias para aplicações em execução em uma instância do EC2 e fazer solicitações da AWS CLI ou da AWS API. É preferível fazer isso do que armazenar chaves de acesso na instância do EC2. Para atribuir uma função da AWS a uma instância do EC2 e disponibilizá-la para todas as suas aplicações, crie um perfil de instância que esteja anexado à instância. Um perfil de instância contém a função e permite que programas que estão em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte [Usar uma função do IAM para conceder permissões a aplicações em execução nas instâncias do Amazon EC2](#) no Manual do usuário do IAM.

Para saber se deseja usar as funções do IAM, consulte [Quando criar uma função do IAM \(em vez de um usuário\)](#) no Guia do usuário do IAM.

Gerenciamento do acesso usando políticas

Você controla o acesso na AWS criando e anexando políticas às identidades do IAM ou aos recursos da AWS. Uma política é um objeto na AWS que, quando associado a uma identidade ou recurso, define suas permissões. Você pode fazer login como o usuário raiz ou um usuário do IAM ou assumir uma função do IAM. Quando você faz uma solicitação, a AWS avalia as políticas relacionadas baseadas em identidade ou baseadas em recursos. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas são armazenadas na AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Manual do usuário do IAM.

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual principal pode executar ações em quais recursos, e em que condições.

Cada entidade do IAM (usuário ou função) começa sem permissões. Em outras palavras, por padrão, os usuários não podem fazer nada, nem mesmo alterar sua própria senha. Para dar permissão a um usuário para fazer algo, um administrador deve anexar uma política de permissões ao usuário. Ou o administrador pode adicionar o usuário a um grupo que tenha as permissões pretendidas. Quando um administrador concede permissões a um grupo, todos os usuários desse grupo recebem essas permissões.

As políticas do IAM definem permissões para uma ação, independentemente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de funções do AWS Management Console, da AWS CLI ou da API da AWS.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou função do IAM. Essas políticas controlam quais ações os usuários e funções podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criar políticas do IAM](#) no Manual do usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda mais como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou função. As políticas gerenciadas são políticas independentes que podem ser anexadas a vários usuários, grupos e funções na Conta da AWS. As políticas gerenciadas incluem políticas gerenciadas pela AWS e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Manual do usuário do IAM.

Políticas com base em recurso

Outros serviços, como Amazon S3, também dão suporte a políticas de permissões baseadas em recursos. Por exemplo: você pode anexar uma política a um bucket do S3 para gerenciar permissões de acesso a esse bucket. A Amazon Polly não é compatível com as políticas baseadas em recursos.

Outros tipos de política

AWSA oferece suporte a tipos de política menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um recurso avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou função do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade da entidade e seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou a função no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Manual do usuário do IAM.
- **Políticas de controle de serviço (SCPs):** SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (UO) no AWS Organizations. O AWS Organizations é um serviço para agrupar e gerenciar centralmente várias Contas da AWS pertencentes à sua empresa. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. Uma SCP limita as permissões para entidades em contas-membro, incluindo cada usuário root da Conta da AWS. Para obter mais informações sobre Organizações e SCPs, consulte [Como os SCPs funcionam](#) no Manual do usuário do AWS Organizations.
- **Políticas de sessão:** são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para uma função ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou da função e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em recurso. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte [Políticas de sessão](#) no Manual do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como a AWS determina se deve permitir uma solicitação quando há vários tipos de política envolvidos, consulte [Lógica da avaliação de políticas](#) no Guia do Usuário do IAM.

Como o Amazon Polly funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao Amazon Polly, você deve entender quais recursos do IAM estão disponíveis para uso com o Amazon Polly. Para obter uma visão detalhada de como a Amazon Polly e outros AWS Serviços funcionam com o IAM, consulte [AWS Serviços compatíveis com o IAM](#) no IAM User Guide.

Tópicos

- [Políticas baseadas em identidade do Amazon Polly \(p. 196\)](#)
- [Funções do Amazon Polly \(p. 197\)](#)

Políticas baseadas em identidade do Amazon Polly

Com as políticas baseadas em identidade do API Gateway, é possível especificar ações ou recursos permitidos ou negados, bem como as condições sob as quais as ações são permitidas ou negadas. O Amazon Polly é compatível com ações, recursos e chaves de condição específicos. Para conhecer todos os elementos usados em uma política JSON, consulte [Referência de elementos de política JSON do IAM](#) no Manual do usuário do IAM.

Actions

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual principal pode executar ações em quais recursos, e em que condições.

O elemento `Action` de uma política JSON descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome que a operação de API da AWS associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Há também algumas operações que exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Inclua ações em uma política para conceder permissões para executar a operação associada.

As ações de políticas no Amazon Polly usam o seguinte prefixo antes da ação: `polly:`. Por exemplo, conceder a alguém permissão para excluir um léxico do Amazon Polly com o `Amazon PollyDeleteLexicon` operação da API, você inclui `opolly:DeletedLexicon` ação em sua política. As instruções de política devem incluir um elemento `Action` ou `NotAction`. A Amazon Polly define seu próprio conjunto de ações que descrevem as tarefas que você pode executar com esse serviço.

Para especificar várias ações em uma única declaração, separe-as com vírgulas, conforme o seguinte:

```
"Action": [
    "polly:action1",
    "polly:action2"]
```

Você também pode especificar várias ações usando caracteres curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `Get`, inclua a seguinte ação:

```
"Action": "polly:Get*"
```

Para ver uma lista das ações do Amazon Polly, consulte [Ações definidas pelo Amazon Polly](#) no IAM User Guide.

Resources

O Amazon Polly não é compatível com a especificação de ARNs de recursos em uma política.

Chaves de condição

O Amazon Polly não fornece nenhuma chave de condição específica ao serviço, mas oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição globais da AWS, consulte [Chaves de contexto de condição globais da AWS](#) no Manual do usuário do IAM.

Examples

Para visualizar exemplos de políticas baseadas em identidade do Amazon Polly, consulte [Exemplos de políticas baseadas em identidade do Amazon Polly](#) (p. 198).

Funções do Amazon Polly

Você pode associar uma política de permissões baseada em identidade a uma função do IAM para conceder permissões entre contas. Por exemplo, o administrador na conta A pode criar uma função para conceder permissões entre contas a outra conta da AWS (por exemplo, a conta B) ou a um serviço da AWS da seguinte forma:

1. O administrador da Conta A cria uma função do IAM e anexa uma política de permissões à função que concede permissões a recursos da conta A.

2. Um administrador da conta A anexa uma política de confiança à função identificando a conta B como a principal, que pode assumir a função.
3. O administrador da conta B pode delegar permissões para assumir a função a todos os usuários na conta B. Isso permite que os usuários na conta B criem ou acessem recursos na conta A. O principal na política de confiança também pode ser um serviço principal da AWS, se você quiser conceder a um serviço da AWS permissões para assumir a função.

Para obter mais informações sobre o uso do IAM para delegar permissões, consulte [Gerenciamento de acesso no IAM User Guide](#).

Veja a seguir um exemplo de política que concede permissões para colocar e obter léxicos, bem como para listar os léxicos disponíveis no momento.

O Amazon Polly é compatível com políticas baseadas em identidade para ações no nível dos recursos. Em alguns casos, o recurso pode ser limitado por um ARN. Isso é válido para as operações `SynthesizeSpeech`, `StartSpeechSynthesisTask`, `PutLexicon`, `GetLexicon` e `DeleteLexicon`. Nesses casos, o valor `Resource` é indicado pelo ARN. Por exemplo: `arn:aws:polly:us-east-2:account-id:lexicon/*` como o valor de `Resource` especifica as permissões em todos os léxicos próprios dentro da região `us-east-2`.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowPut-Get-ListActions",
    "Effect": "Allow",
    "Action": [
      "polly:PutLexicon",
      "polly:GetLexicon",
      "polly:ListLexicons"
    ],
    "Resource": "arn:aws:polly:us-east-2:account-id:lexicon/*"
  }]
}
```

No entanto, nem todas as operações usam ARNs. Este é o caso com `DescribeVoices`, `ListLexicons`, `GetSpeechSynthesisTasks`, `ListSpeechSynthesisTasks` e `DescribeVoices`.

Para obter mais informações sobre usuários, grupos, funções e permissões, consulte [Identidades \(usuários, grupos e funções\)](#) no Guia do usuário do IAM.

Exemplos de políticas baseadas em identidade do Amazon Polly

Por padrão, os usuários e as funções do IAM não têm permissão para criar ou modificar recursos do Amazon Polly. Eles também não podem executar tarefas usando o AWS Management Console, a AWS CLI ou uma API da AWS. Um administrador do IAM deve criar políticas do IAM que concedam aos usuários e funções permissão para executarem operações de API específicas nos recursos especificados de que precisam. O administrador deve anexar essas políticas aos usuários ou grupos do IAM que exigem essas permissões.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte [Criar políticas na guia JSON](#) no Manual do usuário do IAM.

Tópicos

- [Melhores práticas de políticas](#) (p. 199)
- [Usar o console do Amazon Polly](#) (p. 199)

- [Permitir que os usuários visualizem suas próprias permissões \(p. 200\)](#)
- [AWS Políticas gerenciadas \(predefinidas\) da para o Amazon Polly \(p. 200\)](#)
- [Exemplos de política gerenciada pelo cliente \(p. 201\)](#)

Melhores práticas de políticas

As políticas baseadas em identidade são muito eficientes. Elas determinam se alguém pode criar, acessar ou excluir recursos do Amazon Polly em sua conta. Essas ações podem incorrer em custos para a Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece a usar o AWS Políticas gerenciadas pela— Para começar a usar o Amazon Polly rapidamente, use AWS Diretivas gerenciadas pela para conceder a seus funcionários as permissões de que precisam. Essas políticas já estão disponíveis em sua conta e são mantidas e atualizadas pela AWS. Para obter mais informações, consulte [Começar a usar permissões com políticas gerenciadas da AWS](#) no Manual do usuário do IAM.
- Conceder privilégio mínimo: ao criar políticas personalizadas, conceda apenas as permissões necessárias para executar uma tarefa. Comece com um conjunto mínimo de permissões e conceda permissões adicionais conforme necessário. Fazer isso é mais seguro do que começar com permissões que são muito lenientes e tentar restringi-las posteriormente. Para obter mais informações, consulte [Conceder privilégio mínimo](#) no Manual do usuário do IAM.
- Habilitar MFA para operações confidenciais: para aumentar a segurança, exija que os usuários do IAM usem Multi-Factor Authentication (MFA) para acessar recursos ou operações de API confidenciais. Para obter mais informações, consulte [Usar autenticação multifator \(MFA\) na AWS](#) no Manual do usuário do IAM.
- Usar condições de política para segurança adicional: na medida do possível, defina as condições sob as quais suas políticas baseadas em identidade permitem o acesso a um recurso. Por exemplo, você pode gravar condições para especificar um intervalo de endereços IP permitidos do qual a solicitação deve partir. Você também pode escrever condições para permitir somente solicitações em uma data especificada ou período ou para exigir o uso de SSL ou MFA. Para obter mais informações, consulte [Elementos de política JSON do: Condição no IAM User Guide](#).

Usar o console do Amazon Polly

Para que um usuário trabalhe com o console do Amazon Polly, ele deve ter um conjunto mínimo de permissões para descrever os recursos do Amazon Polly em suas AWS conta.

Se você criar uma política do IAM que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para os usuários com essa política do IAM.

Não é necessário conceder permissões mínimas do console aos usuários que só estão fazendo chamadas à AWS CLI ou a API do Amazon Polly.

Para usar o console Amazon Polly, conceda permissões a todas as APIs do Amazon Polly. Não há permissões adicionais necessárias. Para obter a funcionalidade completa do console, é possível usar a seguinte política:

```
}
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "Console-AllowAllPollyActions",
    "Effect": "Allow",
    "Action": [
      "polly:*"
    ],
    "Resource": "*"
  }]
}
```

```
}
```

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como você pode criar uma política que permite que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a API da AWS.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

AWSPolíticas gerenciadas (predefinidas) da para o Amazon Polly

AWS A aborda muitos casos de uso comuns fornecendo políticas autônomas do IAM que são criadas e administradas pela AWS. Essas políticas gerenciadas pela AWS concedem permissões necessárias para casos de uso comuns para que você possa evitar ter que investigar quais permissões são necessárias. Para obter mais informações, consulte [Políticas gerenciadas da AWS](#) no Guia do usuário do IAM.

Os seguintes exemplos deAWSAs políticas gerenciadas pela, que podem ser anexadas a usuários na conta, são específicas do Amazon Polly:

- AmazonPollyReadOnlyAccessConcede acesso de somente leitura a recursos, permite listar e obter léxicos, listar as vozes disponíveis e sintetizar a fala (incluindo a aplicação de léxicos à fala sintetizada).
- AmazonPollyFullAccess – Concede acesso total aos recursos e a todas as operações suportadas.

Note

É possível analisar essas políticas de permissões fazendo login no console do IAM e pesquisando políticas específicas.

Você também pode criar suas próprias políticas do IAM personalizadas e conceder permissões para ações e recursos do Amazon Polly. Você pode anexar essas políticas personalizadas a usuários ou grupos do IAM que exijam essas permissões.

Exemplos de política gerenciada pelo cliente

Nesta seção, você encontrará exemplos de políticas de usuário que concedam permissões para diversas ações do Amazon Polly. Essas políticas funcionam quando você está usando o AWSSDKs ou o AWS CLI. Ao usar o console, conceda permissões a todas as APIs do Amazon Polly.

Note

Todos os exemplos usam a região us-east-2 e contêm IDs de conta fictícios.

Exemplos

- [Exemplo 1: Permitir todas as ações do Amazon Polly \(p. 201\)](#)
- [Exemplo 2: Permitir todas as ações do Amazon Polly, exceto DeleteLexicon \(p. 201\)](#)
- [Exemplo 3: Permitir DeleteLexicon \(p. 202\)](#)
- [Exemplo 4: Permitir a exclusão de léxico de uma região especificada \(p. 202\)](#)
- [Exemplo 5: Permitir DeleteLexicon para léxico especificado \(p. 202\)](#)

Exemplo 1: Permitir todas as ações do Amazon Polly

Depois de se cadastrar (consulte [Etapa 1.1: Cadastre-se no AWS \(p. 4\)](#)), crie um usuário administrador para gerenciar a conta, incluindo a criação de usuários e o gerenciamento das permissões deles.

Você pode criar um usuário que tenha permissões para todas as ações do Amazon Polly. Pense nesse usuário como um administrador específico do serviço para trabalhar com a Amazon Polly. Você pode anexar a política de permissões a seguir para este usuário.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowAllPollyActions",
    "Effect": "Allow",
    "Action": [
      "polly:*"
    ],
    "Resource": "*"
  }]
}
```

Exemplo 2: Permitir todas as ações do Amazon Polly, exceto DeleteLexicon

A política de permissões a seguir concede ao usuário permissões para executar todas as ações, exceto DeleteLexicon, com as permissões para excluir explicitamente negadas em todas as regiões.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowAllActions-DenyDelete",
    "Effect": "Allow",
    "Action": [
      "polly:DescribeVoices",
      "polly:GetLexicon",
      "polly:PutLexicon",
      "polly:SynthesizeSpeech",
      "polly:ListLexicons"
    ],
    "Resource": "*"
  }, {
    "Sid": "DenyDeleteLexicon",
    "Effect": "Deny",
    "Action": "DeleteLexicon",
    "Resource": "*"
  }]
}
```

```
    "Resource": "*"
  }
  {
    "Sid": "DenyDeleteLexicon",
    "Effect": "Deny",
    "Action": [
      "polly:DeleteLexicon",
    ],
    "Resource": "*"
  }
]
```

Exemplo 3: Permitir DeleteLexicon

A política de permissões a seguir concede ao usuário permissões para excluir qualquer léxico que você possua, independentemente do projeto ou da região em que está localizado.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowDeleteLexicon",
    "Effect": "Allow",
    "Action": [
      "polly:DeleteLexicon",
    ],
    "Resource": "*"
  }]
}
```

Exemplo 4: Permitir a exclusão de léxico de uma região especificada

A política de permissões a seguir concede ao usuário permissões para excluir qualquer léxico em qualquer projeto que você possui e que esteja localizado em uma única região (neste caso, us-east-2).

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowDeleteSpecifiedRegion",
    "Effect": "Allow",
    "Action": [
      "polly:DeleteLexicon",
    ],
    "Resource": "arn:aws:polly:us-east-2:123456789012:lexicon/*"
  }]
}
```

Exemplo 5: Permitir DeleteLexicon para léxico especificado

A política de permissões a seguir concede as permissões de usuário para excluir um léxico específico que você possui (neste caso, myLexicon) em uma região específica (neste caso, us-east-2).

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowDeleteForSpecifiedLexicon",
    "Effect": "Allow",
    "Action": [
      "polly:DeleteLexicon",
    ],
    "Resource": "arn:aws:polly:us-east-2:123456789012:lexicon/myLexicon"
  }]
}
```


}

Permissões da API do Amazon Polly: Referência de ações, permissões e recursos

Ao configurar uma política de permissões que pode anexar a uma identidade do IAM (políticas baseadas em identidade), você pode usar a tabela a seguir lista Como referência. O A lista inclui Cada operação da API do Amazon Polly, as ações correspondentes para as quais você pode conceder permissões para executar a ação e a AWS para o qual você pode conceder as permissões. Você especifica as ações no campo `Action` da política e o valor do recurso no campo `Resource` da política.

Você pode usar o AWSAs chaves de condição em toda a em suas políticas do Amazon Polly para expressar condições. Para obter uma lista completa das chaves da AWS, consulte [Chaves disponíveis](#) no Guia do usuário do IAM.

Note

Para especificar uma ação, use o prefixo `polly` seguido do nome da operação da API (por exemplo, `polly:GetLexicon`).

O Amazon Polly é compatível com políticas baseadas em identidade para ações no nível dos recursos. Portanto, o valor `Resource` é indicado pelo ARN. Por exemplo: `arn:aws:polly:us-east-2:account-id:lexicon/*` como o valor de `Resource` especifica as permissões em todos os léxicos próprios dentro da região `us-east-2`.

Como o Amazon Polly não oferece suporte a permissões para ações no nível de recurso, a maioria das políticas especifica um caractere curinga (*) como `Resource` value. No entanto, se for necessário limitar as permissões para uma região específica, esse caractere curinga será substituído pelo ARN adequado: `arn:aws:polly:region:account-id:lexicon/*`.

API do Amazon Polly e permissões necessárias para ações

Operação da API: [DeleteLexicon](#) (p. 213)

Permissões obrigatórias (ação de API): `polly:DeleteLexicon`

Recursos: `arn:aws:polly:region:account-id:lexicon/LexiconName`

Operação da API: [DescribeVoices](#) (p. 215)

Permissões obrigatórias (ação de API): `polly:DescribeVoices`

Recursos: `arn:aws:polly:region:account-id:lexicon/voice-name`

Operação da API: [GetLexicon](#) (p. 218)

Permissões obrigatórias (ação de API): `polly:GetLexicon`

Recursos: `arn:aws:polly:region:account-id:lexicon/voice-name`

Operação da API: [ListLexicons](#) (p. 222)

Permissões obrigatórias (ação de API): `polly:ListLexicons`

Recursos: `arn:aws:polly:region:account-id:lexicon/*`

Operação da API: [PutLexicon](#) (p. 227)

Permissões obrigatórias (ação de API): `polly:ListLexicons`

Recursos: *

Operação da API: [SynthesizeSpeech](#) (p. 235)

Permissões obrigatórias (ação de API): `polly:SynthesizeSpeech`

Recursos: *

Registrar em log e monitorar no Amazon Polly

O monitoramento é uma parte importante para manter a confiabilidade, a disponibilidade e o desempenho dos aplicativos do Amazon Polly. Para monitorar as chamadas da API do Amazon Polly, é possível usar o AWS CloudTrail. Para monitorar o status de seus trabalhos, use o Amazon CloudWatch Logs.

- **Alarmes do Amazon CloudWatch**— Ao usar alarmes do CloudWatch, você observa uma única métrica durante um período especificado. Se a métrica exceder determinado limite, uma notificação será enviada para um tópico do Amazon Simple Notification Service ou para uma política do AWS Auto Scaling. Os alarmes do CloudWatch não invocam ações quando uma métrica está em um estado específico. O estado deve ter sido alterado e mantido por uma quantidade especificada de períodos. Para obter mais informações, consulte [Integração do CloudWatch com o Amazon Polly](#) (p. 209).
- **Logs do CloudTrail**— O CloudTrail fornece um registro de ações executadas por um usuário, uma função ou um AWS serviço no Amazon Polly. Ao usar as informações coletadas pelo CloudTrail, é possível determinar a solicitação que foi feita ao Amazon Polly. Você também pode determinar o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, e detalhes adicionais. Para obter mais informações, consulte [Registrar chamadas de API do Amazon Polly com AWS CloudTrail](#) (p. 206).

Validação de conformidade com o Amazon Polly

Audidores de terceiros avaliam a segurança e a conformidade do Amazon Polly como parte de vários programas de conformidade. Isso inclui SOC, PCI, FedRAMP, HIPAA e outros.

Para obter uma lista de serviços da AWS no escopo de programas de conformidade específicos, consulte [Serviços da AWS no escopo por programa de conformidade](#). Para obter informações gerais, consulte [Programas de conformidade da AWS](#).

Você pode fazer download de relatórios de auditoria externa usando o AWS Artifact. Para obter mais informações, consulte [Fazer download de relatórios no AWS Artifact](#).

Sua responsabilidade de conformidade ao usar a Amazon Polly é determinada pela confidencialidade dos dados, pelos objetivos de conformidade da empresa e pelos regulamentos e leis aplicáveis. AWS fornece os seguintes recursos para ajudar com a conformidade:

- **Guias de início rápido de segurança e conformidade**: esses guias de implantação abordam as considerações de arquitetura e fornecem etapas para implantação de ambientes de linha de base focados em conformidade e segurança na AWS.
- **Whitepaper Arquitetura para segurança e conformidade com a HIPAA**: esse whitepaper descreve como as empresas podem usar a AWS para criar aplicações em conformidade com a HIPAA.
- **Recursos de conformidade da AWS**: essa coleção de manuais e guias pode ser aplicada a seu setor e local.
- **Avaliar recursos com regras** no AWS Config Developer Guide (Guia do desenvolvedor do AWS Config): o serviço AWS Config avalia como as configurações de recursos estão em conformidade com práticas internas, diretrizes do setor e regulamentos.
- **AWS Security Hub**: esse serviço da AWS fornece uma visão abrangente do estado de sua segurança na AWS que ajuda você a verificar sua conformidade com padrões e práticas recomendadas de segurança do setor.

Resiliência no Amazon Polly

A AWS infraestrutura global da é criada com base em AWS Regiões e zonas de disponibilidade da AWS. As regiões da fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, que são conectadas com baixa latência, altas taxas de transferência e redes altamente redundantes. Com as zonas de disponibilidade, você pode projetar e operar aplicativos e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para mais informações sobre regiões e zonas de disponibilidade da AWS, consulte [Infraestrutura global da AWS](#).

Segurança da infraestrutura no Amazon Polly

Como um serviço gerenciado, o Amazon Polly é protegido pela AWS Os procedimentos de segurança da rede global da descritos na [Amazon Web Services: Visão geral dos processos de segurança](#) Whitepaper.

Você usa AWS As chamadas de API publicadas pela para acessar o Amazon Polly pela rede. Os clientes devem oferecer suporte a Transport Layer Security (TLS) 1.0 ou posterior. Recomendamos TLS 1.2 ou posterior. Os clientes também devem ter suporte a conjuntos de criptografia com perfect forward secrecy (PFS) como Ephemeral Diffie-Hellman (DHE) ou Ephemeral Elliptic Curve Diffie-Hellman (ECDHE). A maioria dos sistemas modernos como Java 7 e versões posteriores oferece suporte a esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Melhores práticas de segurança para o Amazon Polly

Sua confiança, privacidade e a segurança do seu conteúdo são nossas maiores prioridades. Implementamos controles técnicos e físicos que são responsáveis e avançados para impedir o acesso não autorizado ao seu conteúdo, bem como a sua divulgação, e garantir que o nosso uso esteja de acordo com os compromissos que firmamos com você. Para obter mais informações, consulte [AWS Perguntas frequentes sobre privacidade de dados](#).

A Amazon Polly não retém o conteúdo dos envios de texto.

Para uma visão ampla do AWS Segurança da, incluindo conformidade, testes de penetração, boletins e recursos, acesse o [AWS Segurança na nuvem](#) Site da.

Registrar chamadas de API do Amazon Polly com AWS CloudTrail

O Amazon Polly é integrado ao AWS CloudTrail, um serviço que fornece um registro de ações executadas por um usuário, uma função ou um AWS serviço no Amazon Polly. O CloudTrail captura todas as chamadas de API para o Amazon Polly como eventos. As chamadas capturadas incluem chamadas do console do Amazon Polly e chamadas de código para as operações de API do Amazon Polly. Se você criar uma trilha, poderá habilitar a entrega contínua de eventos do CloudTrail para um bucket do Amazon S3, incluindo eventos para o Amazon Polly. Se você não configurar uma trilha, ainda poderá visualizar os eventos mais recentes no console do CloudTrail em Event history (Histórico de eventos). Usando as informações coletadas pelo CloudTrail, é possível determinar a solicitação feita para o Amazon Polly, o endereço IP no qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, além de detalhes adicionais.

Para saber mais sobre o CloudTrail, incluindo como configurá-lo e habilitá-lo, consulte [o AWS CloudTrail Guia do usuário](#).

Informações do Amazon Polly no CloudTrail

O CloudTrail está habilitado no AWS. Ao criar a conta. Quando a atividade do evento compatível ocorrer no Amazon Polly, ela será registrada em um evento do CloudTrail juntamente com outros AWS eventos de serviços em Histórico do evento. Você pode visualizar, pesquisar e fazer download de eventos recentes em sua conta da AWS. Para obter mais informações, consulte [Como visualizar eventos com o histórico de eventos do CloudTrail](#).

Para obter um registro contínuo de eventos em seu AWS, incluindo eventos para o Amazon Polly, crie uma trilha. Uma trilha permite que o CloudTrail entregue arquivos de log a um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as regiões da AWS. A trilha registra em log eventos de todas as regiões da AWS. O AWS fornece os arquivos de log para o bucket do Amazon S3 especificado por você. Além disso, você pode configurar outros AWS para analisar mais profundamente e agir sobre os dados de eventos coletados nos logs do CloudTrail. Para mais informações, consulte:

- [Visão geral da criação de uma trilha](#)
- [Serviços e integrações compatíveis com o CloudTrail](#)
- [Configuração de notificações do Amazon SNS para o CloudTrail](#)
- [Receber arquivos de log do CloudTrail de várias regiões e receber arquivos de log do CloudTrail de várias contas](#)

O Amazon Polly oferece suporte ao registro das ações a seguir como eventos nos arquivos de log do CloudTrail:

- [DeleteLexicon](#) (p. 213)
- [DescribeVoices](#) (p. 215)
- [GetLexicon](#) (p. 218)
- [GetSpeechSynthesisTask](#) (p. 220)
- [ListLexicons](#) (p. 222)
- [ListSpeechSynthesisTasks](#) (p. 224)

- [PutLexicon](#) (p. 227)
- [StartSpeechSynthesisTask](#) (p. 229)
- [SynthesizeSpeech](#) (p. 235)

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário raiz ou do AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro serviço da AWS.

Para obter mais informações, consulte o [Elemento userIdentity do CloudTrail](#).

Exemplo: Entradas dos arquivos de log do Amazon Polly

Uma trilha é uma configuração que permite a entrega de eventos como registros de log a um bucket do Amazon S3 especificado. Os arquivos de log do CloudTrail contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer origem e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros de solicitação e assim por diante. Os arquivos de log do CloudTrail não são um rastreamento de pilha ordenada de chamadas de API pública. Dessa forma, eles não são exibidos em uma ordem específica.

O exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra o `SynthesizeSpeech`.

```
{
  "Records": [
    {
      "awsRegion": "us-east-2",
      "eventID": "19bd70f7-5e60-4cdc-9825-936c552278ae",
      "eventName": "SynthesizeSpeech",
      "eventSource": "polly.amazonaws.com",
      "eventTime": "2016-11-02T03:49:39Z",
      "eventType": "AwsApiCall",
      "eventVersion": "1.05",
      "recipientAccountId": "123456789012",
      "requestID": "414288c2-alaf-11e6-b17f-d7cfc06cb461",
      "requestParameters": {
        "lexiconNames": [
          "SampleLexicon"
        ],
        "outputFormat": "mp3",
        "sampleRate": "22050",
        "text": "*****",
        "textType": "text",
        "voiceId": "Kendra"
      },
      "responseElements": {
        "contentType": "audio/mpeg",
        "requestCharacters": 25
      },
      "sourceIPAddress": "1.2.3.4",
      "userAgent": "Amazon CLI/Polly 1.10 API 2016-06-10",
    }
  ]
}
```

```
    "userIdentity": {  
      "accessKeyId": "EXAMPLE_KEY_ID",  
      "accountId": "123456789012",  
      "arn": "arn:aws:iam::123456789012:user/Alice",  
      "principalId": "EX_PRINCIPAL_ID",  
      "type": "IAMUser",  
      "userName": "Alice"  
    }  
  }  
]  
}
```

Integração do CloudWatch com o Amazon Polly

Quando você interage com o Amazon Polly, ele envia as seguintes métricas e dimensões para o CloudWatch a cada minuto. É possível usar os procedimentos a seguir para visualizar as métricas do Amazon Polly.

É possível monitorar o Amazon Polly usando o CloudWatch, que coleta e processa dados brutos do Amazon Polly em métricas legíveis quase em tempo real. Essas estatísticas são registradas por um período de duas semanas, para que você possa acessar `historical information` e obter uma perspectiva melhor sobre como seu serviço ou sua aplicação web estão se saindo. Por padrão, dados métricos do Amazon Polly são enviados para o CloudWatch em intervalos de 1 minuto. Para obter mais informações, consulte [O que é o Amazon CloudWatch](#) na AmazonCloudWatch.

Obmissão das métricas do CloudWatch (Console)

1. Abra o console do CloudWatch em <https://console.aws.amazon.com/cloudwatch/>.
2. No painel de navegação, selecione Metrics (Métricas).
3. No Métricas do CloudWatch por Categoria, na categoria de métricas do Amazon Polly, selecione uma opção e, em seguida, no painel superior, role para baixo para ver a lista completa de métricas.

Obmissão das métricas do CloudWatch (CLI)

O código a seguir exibe métricas disponíveis para o Amazon Polly.

```
aws cloudwatch list-metrics --namespace "AWS/Polly"
```

O comando anterior apresenta uma lista de métricas do Amazon Polly semelhante à lista a seguir. O elemento `MetricName` identifica qual é a métrica.

```
{
  "Metrics": [
    {
      "Namespace": "AWS/Polly",
      "Dimensions": [
        {
          "Name": "Operation",
          "Value": "SynthesizeSpeech"
        }
      ],
      "MetricName": "ResponseLatency"
    },
    {
      "Namespace": "AWS/Polly",
      "Dimensions": [
        {
          "Name": "Operation",
          "Value": "SynthesizeSpeech"
        }
      ],
      "MetricName": "RequestCharacters"
    }
  ]
}
```

```
}
```

Para obter mais informações, consulte [GetMetricStatistics](#) na Referência da API do Amazon CloudWatch.

Métricas do Amazon Polly

O Amazon Polly produz as seguintes métricas para cada solicitação. Essas métricas são agregadas e, em intervalos de um minuto, enviadas para o CloudWatch, onde elas estão disponíveis.

Métrica	Descrição
<code>RequestCharacters</code>	<p>O número de caracteres na solicitação. Isso é apenas caracteres faturáveis e não inclui tags SSML.</p> <p>Dimensão válida: Operação</p> <p>Estatísticas válidas: Mínimo, máximo, Média, SampleCount, Soma</p> <p>Unidade: contagem</p>
<code>ResponseLatency</code>	<p>A latência entre quando a solicitação foi feita e o início da resposta de streaming.</p> <p>Dimensões válidas: Operação</p> <p>Estatísticas válidas: Mínimo, máximo, Média, SampleCount</p> <p>Unidade: milissegundos</p>
<code>2XXCount</code>	<p>Código de nível HTTP 200 retornado após uma resposta bem-sucedida.</p> <p>Dimensões válidas: Operação</p> <p>Estatísticas válidas: Média, SampleCount, Soma</p> <p>Unidade: contagem</p>
<code>4XXCount</code>	<p>Código de erro de nível HTTP 400 retornado em um erro. Para cada resposta bem-sucedida, um zero (0) é emitido.</p> <p>Dimensões válidas: Operação</p> <p>Estatísticas válidas: Média, SampleCount, Soma</p> <p>Unidade: contagem</p>
<code>5XXCount</code>	<p>Código de erro de nível HTTP 500 retornado após um erro. Para cada resposta bem-sucedida, um zero (0) é emitido.</p> <p>Dimensões válidas: Operação</p> <p>Estatísticas válidas: Média, SampleCount, Soma</p> <p>Unidade: contagem</p>

Dimensões para métricas do Amazon Polly

As métricas do Amazon Polly usam oAWSNamespace /Polly e fornece métricas para a dimensão a seguir:

Dimensão	Descrição
Operation	As métricas são agrupadas pelo método de API a que se referem. Os valores sãoSynthesizeSpeech,PutLexicon,DescribeVoices, etc.

Referência da API do Amazon Polly

Esta seção contém a referência da API do Amazon Polly.

Note

Chamadas de API autenticadas devem ser assinadas usando o processo de assinatura do Signature versão 4. Para obter mais informações, consulte [Assinar AWS Solicitações da API](#) [do](#) [Referência geral do Amazon Web Services](#).

Tópicos

- [Actions](#) (p. 212)
- [Tipos de dados](#) (p. 239)

Actions

As ações a seguir são compatíveis:

- [DeleteLexicon](#) (p. 213)
- [DescribeVoices](#) (p. 215)
- [GetLexicon](#) (p. 218)
- [GetSpeechSynthesisTask](#) (p. 220)
- [ListLexicons](#) (p. 222)
- [ListSpeechSynthesisTasks](#) (p. 224)
- [PutLexicon](#) (p. 227)
- [StartSpeechSynthesisTask](#) (p. 229)
- [SynthesizeSpeech](#) (p. 235)

DeleteLexicon

Exclui o léxico de pronúncia armazenado em uma Região da AWS . Um léxico que foi excluído não está disponível para síntese de voz, nem é possível recuperá-lo usando o comando `GetLexicon` ou `ListLexicon` APIs.

Para obter mais informações, consulte [Gerenciamento de léxicos](#).

Sintaxe da solicitação

```
DELETE /v1/lexicons/LexiconName HTTP/1.1
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

[LexiconName](#) (p. 213)

O nome do léxico a ser excluído. Deve ser um léxico existente na região.

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Sim

Corpo da solicitação

A solicitação não tem um corpo de solicitação.

Sintaxe da resposta

```
HTTP/1.1 200
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta 200 HTTP com um corpo HTTP vazio.

Errors

`LexiconNotFoundException`

O Amazon Polly não consegue encontrar o léxico especificado. Isso pode ser causado por um léxico que está faltando, seu nome está escrito incorretamente ou especificando um léxico que está em uma região diferente.

Verifique se o léxico existe, está na região (consulte [ListLexicons](#) (p. 222)) e que você digitou seu nome está digitado corretamente. Então tente novamente.

Código de status HTTP: 404

`ServiceFailureException`

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

DescribeVoices

Retorna a lista de vozes disponíveis para uso ao solicitar uma síntese de fala. Cada voz fala um idioma especificado, é masculino ou feminino, e é identificada por um ID, que é a versão ASCII do nome da voz.

Ao sintetizar a fala (`SynthesizeSpeech`), você fornece o ID de voz para a voz desejada na lista de vozes retornadas pelo `DescribeVoices`.

Por exemplo, você deseja que seu aplicativo leitor de notícias leia notícias em um idioma específico, mas dando a um usuário a opção de escolher a voz. Como usar o `DescribeVoices`, você pode fornecer ao usuário uma lista de vozes disponíveis para selecionar.

Opcionalmente, você pode especificar um código de idioma para filtrar as vozes disponíveis. Por exemplo, se você especificar `en-US` a operação retorna uma lista com todas as vozes disponíveis em inglês dos EUA.

Essa operação exige permissões para executar a ação `polly:DescribeVoices`.

Sintaxe da solicitação

```
GET /v1/voices?  
Engine=Engine&IncludeAdditionalLanguageCodes=IncludeAdditionalLanguageCodes&LanguageCode=LanguageCode&  
HTTP/1.1
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

[Mecanismo \(p. 215\)](#)

Especifica o mecanismo (`standard` ou `neural`) usado pelo Amazon Polly ao processar texto de entrada para síntese de fala.

Valores válidos: `standard` | `neural`

[IncludeAdditionalAllLanguageCodes \(p. 215\)](#)

Valor booleano que indica se deve retornar quaisquer vozes bilíngües que usam o idioma especificado como um idioma adicional. Por exemplo, se você solicitar todos os idiomas que usam inglês dos EUA (`es-US`) e houver uma voz italiana que fala italiano (`it-it`) e inglês dos EUA, essa voz será incluída se você especificar `yes` mas não se você especificar `no`.

[LanguageCode \(p. 215\)](#)

A tag de identificação de idioma (código ISO 639 para o nome do idioma—código ISO 3166 para o país) para filtrar a lista de vozes retornadas. Se você não especificar esse parâmetro opcional, todas as vozes disponíveis serão retornadas.

Valores válidos: `arb` | `cmn-CN` | `cy-GB` | `da-DK` | `de-DE` | `en-AU` | `en-GB` | `en-GB-WLS` | `en-IN` | `en-US` | `es-ES` | `es-MX` | `es-US` | `fr-CA` | `fr-FR` | `is-IS` | `it-IT` | `ja-JP` | `hi-IN` | `ko-KR` | `nb-NO` | `nl-NL` | `pl-PL` | `pt-BR` | `pt-PT` | `ro-RO` | `ru-RU` | `sv-SE` | `tr-TR`

[NextToken \(p. 215\)](#)

Um token de paginação opcional retornado da `DescribeVoices` operação. Se estiver presente, isso indica onde continuar a listagem.

Restrições de comprimento Tamanho mínimo de 0. Tamanho máximo de 4096.

Corpo da solicitação

A solicitação não tem um corpo de solicitação.

Sintaxe da resposta

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "Voices": [
    {
      "AdditionalLanguageCodes": [ "string" ],
      "Gender": "string",
      "Id": "string",
      "LanguageCode": "string",
      "LanguageName": "string",
      "Name": "string",
      "SupportedEngines": [ "string" ]
    }
  ]
}
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

Os seguintes dados são retornados no formato JSON pelo serviço.

[NextToken \(p. 216\)](#)

O token de paginação a ser usado na próxima solicitação para continuar a listagem de vozes. `NextToken` é retornado somente se a resposta estiver truncada.

Type: String

Restrições de comprimento Tamanho mínimo de 0. Tamanho máximo de 4096.

[Vozes \(p. 216\)](#)

Uma lista de vozes com suas propriedades.

Type: Array de [Voice \(p. 247\)](#) objects

Errors

`InvalidNextTokenException`

O `NextToken` é inválido. Verifique se está escrito correctamente e, em seguida, tente novamente.

Código de status HTTP: 400

`ServiceFailureException`

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

GetLexicon

Retorna o conteúdo do léxico de pronúncia especificado armazenado em uma Região da AWS . Para obter mais informações, consulte [Gerenciamento de léxicos](#).

Sintaxe da solicitação

```
GET /v1/lexicons/LexiconName HTTP/1.1
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

[LexiconName](#) (p. 218)

Nome do léxico.

Padrão: [0-9A-Za-z]{1,20}

: obrigatório Sim

Corpo da solicitação

A solicitação não tem um corpo de solicitação.

Sintaxe da resposta

```
HTTP/1.1 200
Content-type: application/json

{
  "Lexicon": {
    "Content": "string",
    "Name": "string"
  },
  "LexiconAttributes": {
    "Alphabet": "string",
    "LanguageCode": "string",
    "LastModified": number,
    "LexemesCount": number,
    "LexiconArn": "string",
    "Size": number
  }
}
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

Os seguintes dados são retornados no formato JSON pelo serviço.

[Vocabulário](#) (p. 218)

Objeto léxico que fornece nome e conteúdo de string do léxico.

Tipo: objeto [Lexicon](#) (p. 240)

[LexiconAttributes \(p. 218\)](#)

Metadados do léxico, incluindo alfabético fonético usado, código de linguagem, ARN léxico, número de léxico definidos no léxico e tamanho do léxico em bytes.

Tipo: objeto [LexiconAttributes \(p. 241\)](#)

Errors

LexiconNotFoundException

O Amazon Polly não consegue encontrar o léxico especificado. Isso pode ser causado por um léxico que está faltando, seu nome está escrito incorretamente ou especificando um léxico que está em uma região diferente.

Verifique se o léxico existe, está na região (consulte [ListLexicons \(p. 222\)](#)) e que você digitou seu nome está digitado corretamente. Então tente novamente.

Código de status HTTP: 404

ServiceFailureException

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

GetSpeechSynthesisTask

Recupera um objeto `SpeechSynthesisTask` específico com base em seu `TaskID`. Este objeto contém informações sobre a tarefa de síntese de fala fornecida, incluindo o status da tarefa, e um link para o bucket S3 contendo a saída da tarefa.

Sintaxe da solicitação

```
GET /v1/synthesisTasks/TaskId HTTP/1.1
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

`TaskId` (p. 220)

O identificador gerado pelo Amazon Polly para uma tarefa de síntese de fala.

Padrão: `^[a-zA-Z0-9_-]{1,100}$`

: obrigatório Sim

Corpo da solicitação

A solicitação não tem um corpo de solicitação.

Sintaxe da resposta

```
HTTP/1.1 200
Content-type: application/json

{
  "SynthesisTask": {
    "CreationTime": number,
    "Engine": "string",
    "LanguageCode": "string",
    "LexiconNames": [ "string" ],
    "OutputFormat": "string",
    "OutputUri": "string",
    "RequestCharacters": number,
    "SampleRate": "string",
    "SnsTopicArn": "string",
    "SpeechMarkTypes": [ "string" ],
    "TaskId": "string",
    "TaskStatus": "string",
    "TaskStatusReason": "string",
    "TextType": "string",
    "VoiceId": "string"
  }
}
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

Os seguintes dados são retornados no formato JSON pelo serviço.

[SynthesisTask \(p. 220\)](#)

Objeto `SynthesisTask` que fornece informações da tarefa solicitada, incluindo formato de saída, tempo de criação, status da tarefa e assim por diante.

Tipo: objeto [SynthesisTask \(p. 244\)](#)

Errors

`InvalidTaskIdException`

O ID da tarefa fornecido é inválido. Forneça um ID de tarefa válido e tente novamente.

Código de status HTTP: 400

`ServiceFailureException`

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

`SynthesisTaskNotFoundException`

A tarefa de síntese de fala com ID de tarefa solicitada não pode ser encontrada.

Código de status HTTP: 400

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

ListLexicons

Retorna uma lista de léxicos de pronúncia armazenados em uma Região da AWS . Para obter mais informações, consulte [Gerenciamento de léxicos](#).

Sintaxe da solicitação

```
GET /v1/lexicons?NextToken=NextToken HTTP/1.1
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

[NextToken](#) (p. 222)

Um token de paginação opaco retornado do anterior `ListLexicons` operação. Se presente, indica onde continuar a lista de léxicos.

Restrições de comprimento Tamanho mínimo de 0. Tamanho máximo de 4096.

Corpo da solicitação

A solicitação não tem um corpo de solicitação.

Sintaxe da resposta

```
HTTP/1.1 200  
Content-type: application/json
```

```
{  
  "Lexicons": [  
    {  
      "Attributes": {  
        "Alphabet": "string",  
        "LanguageCode": "string",  
        "LastModified": number,  
        "LexemesCount": number,  
        "LexiconArn": "string",  
        "Size": number  
      },  
      "Name": "string"  
    },  
    ...  
  ],  
  "NextToken": "string"  
}
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

Os seguintes dados são retornados no formato JSON pelo serviço.

[Lexicons](#) (p. 222)

Uma lista de nomes e atributos léxico.

Type: Array de [LexiconDescription](#) (p. 243)objects

[NextToken](#) (p. 222)

O token de paginação a ser usado na próxima solicitação para continuar a listagem de léxicos. `NextToken` é retornado somente se a resposta estiver truncada.

Type: String

Restrições de comprimento Tamanho mínimo de 0. Tamanho máximo de 4096.

Errors

`InvalidNextTokenException`

O `NextToken` é inválido. Verifique se está escrito correctamente e, em seguida, tente novamente.

Código de status HTTP: 400

`ServiceFailureException`

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

ListSpeechSynthesisTasks

Retorna uma lista de objetos SpeechSynthesisTask ordenados por sua data de criação. Esta operação pode filtrar as tarefas por seu status, por exemplo, permitindo que os usuários listem apenas as tarefas concluídas.

Sintaxe da solicitação

```
GET /v1/synthesisTasks?MaxResults=MaxResults&NextToken=NextToken&Status=Status HTTP/1.1
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

[MaxResults](#) (p. 224)

Número máximo de tarefas de síntese de fala retornadas em uma operação Lista.

Intervalo válido: Valor mínimo de 1. Valor máximo de 100.

[NextToken](#) (p. 224)

O token de paginação a ser usado na próxima solicitação para continuar a listagem de tarefas de síntese de fala.

Restrições de comprimento Tamanho mínimo de 0. Tamanho máximo de 4096.

[Status](#) (p. 224)

Status das tarefas de síntese de fala retornadas em uma operação Lista

Valores válidos: `scheduled` | `inProgress` | `completed` | `failed`

Corpo da solicitação

A solicitação não tem um corpo de solicitação.

Sintaxe da resposta

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "SynthesisTasks": [
    {
      "CreationTime": number,
      "Engine": "string",
      "LanguageCode": "string",
      "LexiconNames": [ "string" ],
      "OutputFormat": "string",
      "OutputUri": "string",
      "RequestCharacters": number,
      "SampleRate": "string",
      "SnsTopicArn": "string",
      "SpeechMarkTypes": [ "string" ],
      "TaskId": "string",
      "TaskStatus": "string",
    }
  ]
}
```

```
        "TaskStatusReason": "string",  
        "TextType": "string",  
        "VoiceId": "string"  
    }  
]  
}
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

Os seguintes dados são retornados no formato JSON pelo serviço.

[NextToken](#) (p. 224)

Um token de paginação opaco retornado da operação Lista anterior nesta solicitação. Se estiver presente, isso indica onde continuar a listagem.

Type: String

Restrições de comprimento Tamanho mínimo de 0. Tamanho máximo de 4096.

[SynthesisTasks](#) (p. 224)

Lista de objetos SynthesisTask que fornece informações da tarefa especificada na solicitação de lista, incluindo formato de saída, tempo de criação, status da tarefa e assim por diante.

Type: Array de [SynthesisTask](#) (p. 244) objects

Errors

InvalidNextTokenException

O NextToken é inválido. Verifique se está escrito correctamente e, em seguida, tente novamente.

Código de status HTTP: 400

ServiceFailureException

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

PutLexicon

Armazena um léxico de pronúncia em uma Região da AWS . Se um léxico com o mesmo nome já existir na região, ele será substituído pelo novo léxico. Operações Lexicon têm consistência eventual, portanto, pode levar algum tempo antes que o léxico esteja disponível para a operação SynthesizeSpeech.

Para obter mais informações, consulte [Gerenciamento de léxicos](#).

Sintaxe da solicitação

```
PUT /v1/lexicons/LexiconName HTTP/1.1
Content-type: application/json

{
  "Content": "string"
}
```

Parâmetros da solicitação de

A solicitação usa os seguintes parâmetros de URI.

[LexiconName](#) (p. 227)

Nome do léxico. O nome deve seguir o formato expresso regular `[0-9a-Za-z]{1,20}`. Ou seja, o nome é uma string alfanumérica que diferencia maiúsculas e minúsculas de até 20 caracteres.

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Sim

Corpo da solicitação

A solicitação aceita os dados a seguir no formato JSON.

[Conteúdo](#) (p. 227)

Conteúdo do léxico PLS como dados de string.

Type: String

: obrigatório Sim

Sintaxe da resposta

```
HTTP/1.1 200
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta 200 HTTP com um corpo HTTP vazio.

Errors

InvalidLexiconException

O Amazon Polly não consegue encontrar o léxico especificado. Verifique se o nome do léxico está escrito corretamente e tente novamente.

Código de status HTTP: 400

LexiconSizeExceedException

O tamanho máximo do léxico especificado seria excedido por essa operação.

Código de status HTTP: 400

MaxLexemeLengthExceedException

O tamanho máximo do léxico seria excedido por essa operação.

Código de status HTTP: 400

MaxLexiconNumberExceedException

O número máximo de léxicos seria excedido por essa operação.

Código de status HTTP: 400

ServiceFailureException

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

UnsupportedDPLSALPHABETException

O alfabeto especificado pelo léxico não é um alfabeto suportado. Os valores válidos são x-sampa e ipa.

Código de status HTTP: 400

UnsupportedDPLSLanguageException

O idioma especificado no léxico não é suportado. Para ver uma lista dos idiomas, consulte [Atributos do léxicos](#).

Código de status HTTP: 400

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

StartSpeechSynthesisTask

Permite a criação de uma tarefa de síntese assíncrona, iniciando um novo `SpeechSynthesisTask`. Essa operação requer todas as informações padrão necessárias para a síntese de fala, além do nome de um bucket do Amazon S3 para que o serviço armazene a saída da tarefa de síntese e dois parâmetros opcionais (`OutputS3KeyPrefix` e `SnsTopicArn`). Uma vez criada a tarefa de síntese, esta operação retornará um `SpeechSynthesisTask` objeto, que incluirá um identificador dessa tarefa, bem como o status atual. O `SpeechSynthesisTask` está disponível por 72 horas após iniciar a tarefa de síntese assíncrona.

Sintaxe da solicitação

```
POST /v1/synthesisTasks HTTP/1.1
Content-type: application/json

{
  "Engine": "string",
  "LanguageCode": "string",
  "LexiconNames": [ "string" ],
  "OutputFormat": "string",
  "OutputS3BucketName": "string",
  "OutputS3KeyPrefix": "string",
  "SampleRate": "string",
  "SnsTopicArn": "string",
  "SpeechMarkTypes": [ "string" ],
  "Text": "string",
  "TextType": "string",
  "VoiceId": "string"
}
```

Parâmetros da solicitação de

A solicitação não usa parâmetros de URI.

Corpo da solicitação

A solicitação aceita os dados a seguir no formato JSON.

Mecanismo (p. 229)

Especifica o mecanismo (`standard` ou `neural`) para o Amazon Polly usar ao processar texto de entrada para síntese de fala. O uso de uma voz que não é suportada para o mecanismo selecionado resultará em um erro.

Type: String

Valores válidos: `standard` | `neural`

: obrigatório Não

LanguageCode (p. 229)

Código de idioma opcional para a solicitação de Síntese de Fala. Isso só é necessário se usar uma voz bilíngüe, como Aditi, que pode ser usada para inglês indiano (`en-in`) ou hindi (`hi-in`).

Se uma voz bilíngüe for usada e nenhum código de idioma for especificado, o Amazon Polly usará o idioma padrão da voz bilíngüe. O idioma padrão para qualquer voz é o retornado pelo [DescribeVoices](#) operação para o `LanguageCode` parâmetro. Por exemplo, se nenhum código de idioma for especificado, o Aditi usará inglês indiano em vez de hindi.

Type: String

Valores válidos: `arb` | `cmn-CN` | `cy-GB` | `da-DK` | `de-DE` | `en-AU` | `en-GB` | `en-GB-WLS` | `en-IN` | `en-US` | `es-ES` | `es-MX` | `es-US` | `fr-CA` | `fr-FR` | `is-IS` | `it-IT` | `ja-JP` | `hi-IN` | `ko-KR` | `nb-NO` | `nl-NL` | `pl-PL` | `pt-BR` | `pt-PT` | `ro-RO` | `ru-RU` | `sv-SE` | `tr-TR`

: obrigatório Não

[LexiconNames \(p. 229\)](#)

Lista de um ou mais nomes de léxico de pronúncia que você deseja que o serviço aplique durante a síntese. Os léxico só serão aplicados se o idioma do léxico for o mesmo que o idioma da voz.

Type: Matriz de strings

Membros da matriz: Número máximo de 5 itens.

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Não

[OutputFormat \(p. 229\)](#)

O formato no qual a saída retornada será codificada. Para fluxo de áudio, este será mp3, ogg_vorbis ou pcm. Para marcas de fala, este será JSON.

Type: String

Valores válidos: `json` | `mp3` | `ogg_vorbis` | `pcm`

: obrigatório Sim

[OutputS3BucketName \(p. 229\)](#)

Nome do bucket do Amazon S3 no qual o arquivo de saída será salvo.

Type: String

Padrão: `^[a-z0-9][\.\-a-z0-9]{1,61}[a-z0-9]$`

: obrigatório Sim

[OutputS3KeyPrefix \(p. 229\)](#)

O key prefix do Amazon S3 para o arquivo de fala de saída.

Type: String

Padrão: `^[0-9a-zA-Z\/\!\@_\.*\'\(\)\:;\$@+=\,\?&]{0,800}$`

: obrigatório Não

[SampleRate \(p. 229\)](#)

A frequência de áudio especificada em Hz.

Os valores válidos para mp3 e ogg_vorbis são "8000", "16000", "22050" e "24000". O valor padrão para vozes padrão é "22050". O valor padrão para vozes neurais é "24000".

Os valores válidos para pcm são "8000" e "16000". O valor padrão é "16000".

Type: String

: obrigatório Não

[SnsTopicArn \(p. 229\)](#)

ARN para o tópico SNS usado opcionalmente para fornecer notificação de status para uma tarefa de síntese de fala.

Type: String

Padrão: `^arn:aws(-(cn|iso(-b)?|us-gov))?:sns:[a-z0-9_-]{1,50}:\d{12}::[a-zA-Z0-9_-]{1,256}$`

: obrigatório Não

[SpeechMarkTypes \(p. 229\)](#)

O tipo de marcas de fala retornadas para o texto de entrada.

Type: Matriz de strings

Membros da matriz: Número máximo de 4 itens.

Valores válidos: `sentence` | `ssml` | `viseme` | `word`

: obrigatório Não

[Texto \(p. 229\)](#)

O texto de entrada a ser sintetizado. Se você especificar `ssml` como `TextType`, siga o formato SSML para o texto de entrada.

Type: String

: obrigatório Sim

[TextType \(p. 229\)](#)

Especifica se o texto de entrada é texto sem formatação ou SSML. O valor padrão é texto sem formatação.

Type: String

Valores válidos: `ssml` | `text`

: obrigatório Não

[VoiceID \(p. 229\)](#)

ID de voz a ser usada para a síntese.

Type: String

Valores válidos: `Aditi` | `Amy` | `Astrid` | `Bianca` | `Brian` | `Camila` | `Carla` | `Carmen` | `Celine` | `Chantal` | `Conchita` | `Cristiano` | `Dora` | `Emma` | `Enrique` | `Ewa` | `Filiz` | `Geraint` | `Giorgio` | `Gwyneth` | `Hans` | `Ines` | `Ivy` | `Jacek` | `Jan` | `Joanna` | `Joey` | `Justin` | `Karl` | `Kendra` | `Kevin` | `Kimberly` | `Lea` | `Liv` | `Lotte` | `Lucia` | `Lupe` | `Mads` | `Maja` | `Marlene` | `Mathieu` | `Matthew` | `Maxim` | `Mia` | `Miguel` | `Mizuki` | `Naja` | `Nicole` | `Penelope` | `Raveena` | `Ricardo` | `Ruben` | `Russell` | `Salli` | `Seoyeon` | `Takumi` | `Tatyana` | `Vicki` | `Vitoria` | `Zeina` | `Zhiyu`

: obrigatório Sim

Sintaxe da resposta

HTTP/1.1 200

Content-type: application/json

```
{
  "SynthesisTask": {
    "CreationTime": number,
    "Engine": "string",
    "LanguageCode": "string",
    "LexiconNames": [ "string " ],
    "OutputFormat": "string",
    "OutputUri": "string",
    "RequestCharacters": number,
    "SampleRate": "string",
    "SnsTopicArn": "string",
    "SpeechMarkTypes": [ "string " ],
    "TaskId": "string",
    "TaskStatus": "string",
    "TaskStatusReason": "string",
    "TextType": "string",
    "VoiceId": "string"
  }
}
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

Os seguintes dados são retornados no formato JSON pelo serviço.

[SynthesisTask](#) (p. 231)

Objeto `SynthesisTask` que fornece informações e atributos sobre uma tarefa de síntese de fala recém-enviada.

Tipo: objeto [SynthesisTask](#) (p. 244)

Errors

`EngineNotSupportedException`

Este mecanismo não é compatível com a voz que você designou. Escolha uma nova voz compatível com o motor ou altere o motor e reinicie a operação.

Código de status HTTP: 400

`InvalidS3BucketException`

O nome do bucket do Amazon S3 fornecido é inválido. Verifique sua entrada com os requisitos de nomenclatura do bucket S3 e tente novamente.

Código de status HTTP: 400

`InvalidS3KeyException`

O key prefix do Amazon S3 fornecido é inválido. Forneça um nome de chave de objeto S3 válido.

Código de status HTTP: 400

`InvalidSampleRateException`

A taxa de amostragem especificada é inválida.

Código de status HTTP: 400

InvalidSnsTopicArnException

O ARN do tópico do SNS fornecido é inválido. Forneça um ARN de tópico válido do SNS e tente novamente.

Código de status HTTP: 400

InvalidSMLException

O SSML fornecido é inválido. Verifique a sintaxe SSML, ortografia de tags e valores e tente novamente.

Código de status HTTP: 400

LanguageNotSupportedException

O idioma especificado não é suportado atualmente pelo Amazon Polly nessa capacidade.

Código de status HTTP: 400

LexiconNotFoundException

O Amazon Polly não consegue encontrar o léxico especificado. Isso pode ser causado por um léxico que está faltando, seu nome está escrito incorretamente ou especificando um léxico que está em uma região diferente.

Verifique se o léxico existe, está na região (consulte [ListLexicons \(p. 222\)](#)) e que você digitou seu nome está digitado corretamente. Então tente novamente.

Código de status HTTP: 404

MarksNotSupportedForFormatException

As marcas de fala não são suportadas para o `OutputFormat` selecionado. As marcas de fala só estão disponíveis para conteúdo no `jsonformat`.

Código de status HTTP: 400

ServiceFailureException

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

ssmlmarksNotSupportedForTextTypeException

As marcas de fala SSML não são suportadas para entrada de texto simples.

Código de status HTTP: 400

TextLengthExceedException

O valor do parâmetro "Text" é maior do que os limites aceitos. Para o `synthesizeSpeech`, o limite para o texto de entrada é um máximo de 6000 caracteres no total, dos quais não mais de 3000 caracteres podem ser faturados. Para o `startSpeechSynthesisTask`, o máximo é 200.000 caracteres, dos quais não mais de 100.000 caracteres podem ser faturados. Tags SSML não são contadas como caracteres cobrados.

Código de status HTTP: 400

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWSOs SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)

- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

SynthesizeSpeech

Sintetiza entrada UTF-8, texto sem formatação ou SSML em um fluxo de bytes. A entrada SSML deve ser válida, bem formada SSML. Alguns alfabetos podem não estar disponíveis com todas as vozes (por exemplo, cirílico pode não ser lido por vozes inglesas) a menos que o mapeamento fonema seja usado. Para obter mais informações, consulte [Como ele funciona](#).

Sintaxe da solicitação

```
POST /v1/speech HTTP/1.1
Content-type: application/json

{
  "Engine": "string",
  "LanguageCode": "string",
  "LexiconNames": [ "string" ],
  "OutputFormat": "string",
  "SampleRate": "string",
  "SpeechMarkTypes": [ "string" ],
  "Text": "string",
  "TextType": "string",
  "VoiceId": "string"
}
```

Parâmetros da solicitação de

A solicitação não usa parâmetros de URI.

Corpo da solicitação

A solicitação aceita os dados a seguir no formato JSON.

[Mecanismo \(p. 235\)](#)

Especifica o mecanismo (`standard` ou `neural`) para o Amazon Polly usar ao processar texto de entrada para síntese de fala. Para obter informações sobre as vozes do Amazon Polly e quais vozes estão disponíveis nos formatos somente padrão, somente NTTS e padrão e NTTS, consulte [Vozes disponíveis](#).

Vozes somente para NTTS

Ao usar vozes somente NTTS, como Kevin (`en-US`), esse parâmetro é obrigatório e deve ser definido como `neural`. Se o motor não for especificado, ou estiver definido como `standard`, isso resultará em um erro.

Type: String

Valores válidos: `standard` | `neural`

: obrigatório Sim

Vozes padrão

Para vozes padrão, isso não é necessário; o parâmetro do motor usa como padrão `standard`. Se o motor não for especificado, ou estiver definido como `standard` e uma voz somente de NTTS estiver selecionada, isso resultará em um erro.

Type: String

Valores válidos: `standard` | `neural`

: obrigatório Não

[LanguageCode \(p. 235\)](#)

Código de idioma opcional para a solicitação Sintetizar Voz. Isso só é necessário se usar uma voz bilíngüe, como Aditi, que pode ser usada para inglês indiano (en-in) ou hindi (hi-in).

Se uma voz bilíngüe for usada e nenhum código de idioma for especificado, o Amazon Polly usará o idioma padrão da voz bilíngüe. O idioma padrão para qualquer voz é o retornado pelo [DescribeVoices](#) operação para o `LanguageCode` parâmetro. Por exemplo, se nenhum código de idioma for especificado, o Aditi usará inglês indiano em vez de hindi.

Type: String

Valores válidos: `arb` | `cmn-CN` | `cy-GB` | `da-DK` | `de-DE` | `en-AU` | `en-GB` | `en-GB-WLS` | `en-IN` | `en-US` | `es-ES` | `es-MX` | `es-US` | `fr-CA` | `fr-FR` | `is-IS` | `it-IT` | `ja-JP` | `hi-IN` | `ko-KR` | `nb-NO` | `nl-NL` | `pl-PL` | `pt-BR` | `pt-PT` | `ro-RO` | `ru-RU` | `sv-SE` | `tr-TR`

: obrigatório Não

[LexiconNames \(p. 235\)](#)

Lista de um ou mais nomes de léxico de pronúncia que você deseja que o serviço aplique durante a síntese. Os léxico só serão aplicados se o idioma do léxico for o mesmo que o idioma da voz. Para obter mais informações sobre como armazenar léxicos, consulte [PutLexicon](#).

Type: Matriz de strings

Membros da matriz: Número máximo de 5 itens.

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Não

[OutputFormat \(p. 235\)](#)

O formato no qual a saída retornada será codificada. Para fluxo de áudio, este será `mp3`, `ogg_vorbis` ou `pcm`. Para marcas de fala, este será `JSON`.

Quando o `pcm` for usado, o conteúdo retornado é de áudio/pcm em um formato little-endian assinado de 16 bits e 1 canal (mono).

Type: String

Valores válidos: `json` | `mp3` | `ogg_vorbis` | `pcm`

: obrigatório Sim

[SampleRate \(p. 235\)](#)

A frequência de áudio especificada em Hz.

Os valores válidos para `mp3` e `ogg_vorbis` são "8000", "16000", "22050" e "24000". O valor padrão para vozes padrão é "22050". O valor padrão para vozes neurais é "24000".

Os valores válidos para `pcm` são "8000" e "16000". O valor padrão é "16000".

Type: String

: obrigatório Não

SpeechMarkTypes (p. 235)

O tipo de marcas de fala retornadas para o texto de entrada.

Type: Matriz de strings

Membros da matriz: Número máximo de 4 itens.

Valores válidos: `sentence` | `ssml` | `viseme` | `word`

: obrigatório Não

Texto (p. 235)

Texto de entrada para sintetizar. Se você especificar `ssml` como o `TextType`, siga o formato SSML para o texto de entrada.

Type: String

: obrigatório Sim

TextType (p. 235)

Especifica se o texto de entrada é texto sem formatação ou SSML. O valor padrão é texto sem formatação. Para obter mais informações, consulte [Using SSML](#).

Type: String

Valores válidos: `ssml` | `text`

: obrigatório Não

VoiceID (p. 235)

ID de voz a ser usada para a síntese. Você pode obter uma lista de IDs de voz disponíveis chamando o método [DescribeVoices](#) operação.

Type: String

Valores válidos: `Aditi` | `Amy` | `Astrid` | `Bianca` | `Brian` | `Camila` | `Carla` | `Carmen` | `Celine` | `Chantal` | `Conchita` | `Cristiano` | `Dora` | `Emma` | `Enrique` | `Ewa` | `Filiz` | `Geraint` | `Giorgio` | `Gwyneth` | `Hans` | `Ines` | `Ivy` | `Jacek` | `Jan` | `Joanna` | `Joey` | `Justin` | `Karl` | `Kendra` | `Kevin` | `Kimberly` | `Lea` | `Liv` | `Lotte` | `Lucia` | `Lupe` | `Mads` | `Maja` | `Marlene` | `Mathieu` | `Matthew` | `Maxim` | `Mia` | `Miguel` | `Mizuki` | `Naja` | `Nicole` | `Penelope` | `Raveena` | `Ricardo` | `Ruben` | `Russell` | `Salli` | `Seoyeon` | `Takumi` | `Tatyana` | `Vicki` | `Vitoria` | `Zeina` | `Zhiyu`

: obrigatório Sim

Sintaxe da resposta

```
HTTP/1.1 200
Content-Type: ContentType
x-amzn-RequestCharacters: RequestCharacters

AudioStream
```

Elementos de resposta

Se a ação for bem-sucedida, o serviço reenviará uma resposta HTTP 200.

A resposta retorna os seguintes cabeçalhos HTTP.

[ContentType \(p. 237\)](#)

Especifica o tipo de fluxo de áudio. Isso deve refletir o `OutputFormat` parâmetro em sua solicitação.

- Se você solicitar `mp3` como o `OutputFormat`, o `ContentType` retornado é áudio/mpeg.
- Se você solicitar `ogg_vorbis` como o `OutputFormat`, o `ContentType` retornado é áudio/ogg.
- Se você solicitar `pcm` como o `OutputFormat`, o `ContentType` retornado é áudio/pcm em um formato little-endian assinado de 16 bits e 1 canal (mono).
- Se você solicitar `json` como o `OutputFormat`, o `ContentType` retornado é áudio/json.

[Caracteres de solicitação \(p. 237\)](#)

Número de caracteres sintetizados.

A resposta retorna o seguinte como o corpo HTTP.

[AudioStream \(p. 237\)](#)

Stream contendo o discurso sintetizado.

Errors

`EngineNotSupportedException`

Este mecanismo não é compatível com a voz que você designou. Escolha uma nova voz compatível com o motor ou altere o motor e reinicie a operação.

Código de status HTTP: 400

`InvalidSampleRateException`

A taxa de amostragem especificada é inválida.

Código de status HTTP: 400

`InvalidSSMLException`

O SSML fornecido é inválido. Verifique a sintaxe SSML, ortografia de tags e valores e tente novamente.

Código de status HTTP: 400

`LanguageNotSupportedException`

O idioma especificado não é suportado atualmente pelo Amazon Polly nessa capacidade.

Código de status HTTP: 400

`LexiconNotFoundException`

O Amazon Polly não consegue encontrar o léxico especificado. Isso pode ser causado por um léxico que está faltando, seu nome está escrito incorretamente ou especificando um léxico que está em uma região diferente.

Verifique se o léxico existe, está na região (consulte [ListLexicons \(p. 222\)](#)) e que você digitou seu nome está digitado corretamente. Então tente novamente.

Código de status HTTP: 404

MarksNotSupportedForForForFormatException

As marcas de fala não são suportadas para o `OutputFormat` selecionado. As marcas de fala só estão disponíveis para conteúdo no `jsonformat`.

Código de status HTTP: 400

ServiceFailureException

Uma condição desconhecida causou uma falha de serviço.

Código de status HTTP: 500

ssmlmarksNotSupportedForTextTypeException

As marcas de fala SSML não são suportadas para entrada de texto simples.

Código de status HTTP: 400

TextLengthExceedException

O valor do parâmetro "Text" é maior do que os limites aceitos. Para o `SynthesizeSpeech`, o limite para o texto de entrada é um máximo de 6000 caracteres no total, dos quais não mais de 3000 caracteres podem ser faturados. Para o `startSpeechSynthesisTask`, o máximo é 200.000 caracteres, dos quais não mais de 100.000 caracteres podem ser faturados. Tags SSML não são contadas como caracteres cobrados.

Código de status HTTP: 400

Consulte também

Para obter mais informações sobre como usar essa API em uma das AWS SDKs, consulte o seguinte:

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para JavaScript](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)
- [AWS SDK para Ruby V3](#)

Tipos de dados

Os seguintes tipos de dados são compatíveis:

- [Lexicon](#) (p. 240)
- [LexiconAttributes](#) (p. 241)
- [LexiconDescription](#) (p. 243)
- [SynthesisTask](#) (p. 244)
- [Voice](#) (p. 247)

Lexicon

Fornece o nome do léxico e o conteúdo do léxico em formato de string. Para obter mais informações, consulte [Pronunciation Lexicon Specification \(PLS\) Versão 1.0](#).

Contents

Conteúdo

Conteúdo do léxico em formato de string. O conteúdo de um léxico deve estar no formato PLS.

Type: String

: obrigatório Não

Nome

Nome do léxico.

Type: String

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Não

Consulte também

Para obter mais informações sobre como usar essa API em um dos AWS SDKs, consulte o seguinte:

- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para Ruby V3](#)

LexiconAttributes

Contém metadados que descrevem o léxico, como o número de lexemes, código de idioma e assim por diante. Para obter mais informações, consulte [Gerenciamento de léxicos](#).

Contents

Alfabeto

Alfabeto fonético usado no léxico. Os valores válidos são `ipa` e `x-sampa`.

Type: String

: obrigatório Não

LanguageCode

Código de idioma ao qual o léxico se aplica. Um léxico com um código de idioma como “en” seria aplicado a todos os idiomas ingleses (en-BR, en-US, en-AUS, en-WLs e assim por diante).

Type: String

Valores válidos: `arb` | `cmn-CN` | `cy-GB` | `da-DK` | `de-DE` | `en-AU` | `en-GB` | `en-GB-WLS` | `en-IN` | `en-US` | `es-ES` | `es-MX` | `es-US` | `fr-CA` | `fr-FR` | `is-IS` | `it-IT` | `ja-JP` | `hi-IN` | `ko-KR` | `nb-NO` | `nl-NL` | `pl-PL` | `pt-BR` | `pt-PT` | `ro-RO` | `ru-RU` | `sv-SE` | `tr-TR`

: obrigatório Não

ÚltimaModificação

O léxico de data foi modificado pela última vez (um valor de carimbo de data/hora).

Type: Time stamp

: obrigatório Não

LexemesCount

Número de léxicos no léxico.

Type: Inteiro

: obrigatório Não

Lexiconarn

Nome de recurso da Amazon (ARN) do léxico.

Type: String

: obrigatório Não

Tamanho

O tamanho total do léxico, em caracteres.

Type: Inteiro

: obrigatório Não

Consulte também

Para obter mais informações sobre como usar essa API em um dos AWS SDKs, consulte o seguinte:

- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para Ruby V3](#)

LexiconDescription

Descreve o conteúdo do léxico.

Contents

Atributos.

Fornece metadados léxico.

Tipo: objeto [LexiconAttributes](#) (p. 241)

: obrigatório Não

Nome

Nome do léxico.

Type: String

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Não

Consulte também

Para obter mais informações sobre como usar essa API em um dos AWS SDKs, consulte o seguinte:

- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para Ruby V3](#)

SynthesisTask

Objeto SynthesisTask que fornece informações sobre uma tarefa de síntese de fala.

Contents

CreationTime

Carimbo de data/hora para o momento em que a tarefa de síntese foi iniciada.

Type: Time stamp

: obrigatório Não

Mecanismo

Especifica o mecanismo (`standard` ou `neural`) para o Amazon Polly usar ao processar texto de entrada para síntese de fala. O uso de uma voz que não é suportada para o mecanismo selecionado resultará em um erro.

Type: String

Valores válidos: `standard` | `neural`

: obrigatório Não

LanguageCode

Código de idioma opcional para uma tarefa de síntese. Isso só é necessário se usar uma voz bilíngüe, como Aditi, que pode ser usada para inglês indiano (`en-in`) ou hindi (`hi-in`).

Se uma voz bilíngüe for usada e nenhum código de idioma for especificado, o Amazon Polly usará o idioma padrão da voz bilíngüe. O idioma padrão para qualquer voz é o retornado pelo [DescribeVoices](#) operação para `LanguageCode` parâmetro. Por exemplo, se nenhum código de idioma for especificado, o Aditi usará inglês indiano em vez de hindi.

Type: String

Valores válidos: `arb` | `cmn-CN` | `cy-GB` | `da-DK` | `de-DE` | `en-AU` | `en-GB` | `en-GB-WLS` | `en-IN` | `en-US` | `es-ES` | `es-MX` | `es-US` | `fr-CA` | `fr-FR` | `is-IS` | `it-IT` | `ja-JP` | `hi-IN` | `ko-KR` | `nb-NO` | `nl-NL` | `pl-PL` | `pt-BR` | `pt-PT` | `ro-RO` | `ru-RU` | `sv-SE` | `tr-TR`

: obrigatório Não

LexiconNames

Lista de um ou mais nomes de léxico de pronúncia que você deseja que o serviço aplique durante a síntese. Os léxicos só são aplicados se o idioma do léxico for o mesmo que o idioma da voz.

Type: Matriz de strings

Membros da matriz: Número máximo de 5 itens.

Padrão: `[0-9A-Za-z]{1,20}`

: obrigatório Não

OutputFormat

O formato no qual a saída retornada será codificada. Para fluxo de áudio, este será `mp3`, `ogg_vorbis` ou `pcm`. Para marcas de fala, este será `JSON`.

Type: String

Valores válidos: `json` | `mp3` | `ogg_vorbis` | `pcm`

: obrigatório Não

OutputURI

Caminho do arquivo de saída da fala.

Type: String

: obrigatório Não

Caracteres de solicitação

Número de caracteres faturáveis sintetizados.

Type: Inteiro

: obrigatório Não

SampleRate

A frequência de áudio especificada em Hz.

Os valores válidos para `mp3` e `ogg_vorbis` são "8000", "16000", "22050" e "24000". O valor padrão para vozes padrão é "22050". O valor padrão para vozes neurais é "24000".

Os valores válidos para `pcm` são "8000" e "16000". O valor padrão é "16000".

Type: String

: obrigatório Não

SnsTopicArn

ARN para o tópico SNS usado opcionalmente para fornecer notificação de status para uma tarefa de síntese de fala.

Type: String

Padrão: `^arn:aws(-(cn|iso(-b)?|us-gov))?:sns:[a-z0-9_-]{1,50}:\d{12}:[a-zA-Z0-9_-]{1,256}$`

: obrigatório Não

SpeechMarkTypes

O tipo de marcas de fala retornadas para o texto de entrada.

Type: Matriz de strings

Membros da matriz: Número máximo de 4 itens.

Valores válidos: `sentence` | `ssml` | `viseme` | `word`

: obrigatório Não

TaskId

O identificador gerado pelo Amazon Polly para uma tarefa de síntese de fala.

Type: String

Padrão: `^[a-zA-Z0-9_-]{1,100}$`

: obrigatório Não

taskStatus

Status atual da tarefa de síntese de fala individual.

Type: String

Valores válidos: `scheduled` | `inProgress` | `completed` | `failed`

: obrigatório Não

TaskStatusReason

Motivo para o status atual de uma tarefa específica de síntese de fala, incluindo erros se a tarefa falhou.

Type: String

: obrigatório Não

TextType

Especifica se o texto de entrada é texto sem formatação ou SSML. O valor padrão é texto sem formatação.

Type: String

Valores válidos: `ssml` | `text`

: obrigatório Não

VoiceID

ID de voz a ser usada para a síntese.

Type: String

Valores válidos: `Aditi` | `Amy` | `Astrid` | `Bianca` | `Brian` | `Camila` | `Carla` | `Carmen` | `Celine` | `Chantal` | `Conchita` | `Cristiano` | `Dora` | `Emma` | `Enrique` | `Ewa` | `Filiz` | `Geraint` | `Giorgio` | `Gwyneth` | `Hans` | `Ines` | `Ivy` | `Jacek` | `Jan` | `Joanna` | `Joey` | `Justin` | `Karl` | `Kendra` | `Kevin` | `Kimberly` | `Lea` | `Liv` | `Lotte` | `Lucia` | `Lupe` | `Mads` | `Maja` | `Marlene` | `Mathieu` | `Matthew` | `Maxim` | `Mia` | `Miguel` | `Mizuki` | `Naja` | `Nicole` | `Penelope` | `Raveena` | `Ricardo` | `Ruben` | `Russell` | `Salli` | `Seoyeon` | `Takumi` | `Tatyana` | `Vicki` | `Vitoria` | `Zeina` | `Zhiyu`

: obrigatório Não

Consulte também

Para obter mais informações sobre como usar essa API em um dos AWS SDKs, consulte o seguinte:

- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para Ruby V3](#)

Voice

Descrição da voz.

Contents

AdicionalAlanguageCodes

Códigos adicionais para idiomas disponíveis para a voz especificada, além do idioma padrão.

Por exemplo, o idioma padrão para Aditi é inglês indiano (en-IN) porque foi usado pela primeira vez para esse idioma. Como Aditi é bilíngüe e fluente em inglês indiano e hindi, este parâmetro mostraria o código hi-IN.

Type: Matriz de strings

Valores válidos: arb | cmn-CN | cy-GB | da-DK | de-DE | en-AU | en-GB | en-GB-WLS | en-IN | en-US | es-ES | es-MX | es-US | fr-CA | fr-FR | is-IS | it-IT | ja-JP | hi-IN | ko-KR | nb-NO | nl-NL | pl-PL | pt-BR | pt-PT | ro-RO | ru-RU | sv-SE | tr-TR

: obrigatório Não

Sexo

Gênero da voz.

Type: String

Valores válidos: Female | Male

: obrigatório Não

Id

Amazon Polly atribuiu ID de voz. Este é o ID que você especifica ao chamar o método SynthesizeSpeech operação.

Type: String

Valores válidos: Aditi | Amy | Astrid | Bianca | Brian | Camila | Carla | Carmen | Celine | Chantal | Conchita | Cristiano | Dora | Emma | Enrique | Ewa | Filiz | Geraint | Giorgio | Gwyneth | Hans | Ines | Ivy | Jacek | Jan | Joanna | Joey | Justin | Karl | Kendra | Kevin | Kimberly | Lea | Liv | Lotte | Lucia | Lupe | Mads | Maja | Marlene | Mathieu | Matthew | Maxim | Mia | Miguel | Mizuki | Naja | Nicole | Penelope | Raveena | Ricardo | Ruben | Russell | Salli | Seoyeon | Takumi | Tatyana | Vicki | Vitoria | Zeina | Zhiyu

: obrigatório Não

LanguageCode

Código de linguagem da voz.

Type: String

Valores válidos: arb | cmn-CN | cy-GB | da-DK | de-DE | en-AU | en-GB | en-GB-WLS | en-IN | en-US | es-ES | es-MX | es-US | fr-CA | fr-FR | is-IS | it-IT | ja-JP | hi-IN | ko-KR | nb-NO | nl-NL | pl-PL | pt-BR | pt-PT | ro-RO | ru-RU | sv-SE | tr-TR

: obrigatório Não

LanguageName

Nome legível do idioma em inglês.

Type: String

: obrigatório Não

Nome

Nome da voz (por exemplo, Salli, Kendra, etc.). Isso fornece um nome de voz legível por humanos que você pode exibir em seu aplicativo.

Type: String

: obrigatório Não

SupportedEngines

Especifica quais mecanismos (`standard` ou `neural`) que são suportados por uma determinada voz.

Type: Matriz de strings

Valores válidos: `standard` | `neural`

: obrigatório Não

Consulte também

Para obter mais informações sobre como usar essa API em um dos AWS SDKs, consulte o seguinte:

- [AWS SDK para C++](#)
- [AWS SDK para Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK para Ruby V3](#)

Histórico de documentos do Amazon Polly

A tabela a seguir descreve as alterações importantes em cada versão do Guia do desenvolvedor do Amazon Polly. Para receber notificações sobre atualizações dessa documentação, você pode se inscrever em um feed RSS.

- Última atualização de documentação: 28 de junho de 2021

update-history-change	update-history-description	update-history-date
Novo recurso (p. 249)	Amazon Polly torna o estilo de conversação a versão padrão para as vozes neurais de Matthew e Joanna. Removemos referências ao estilo de conversação.	28 de junho de 2021
Nova voz adicionada para NTTS (p. 249)	Amazon Polly agora fornece a voz alemã Vicki como uma voz NTTS.	15 de junho de 2021
Nova voz adicionada (p. 249)	Uma nova voz feminina, Gabrielle, foi adicionada à localidade francesa (canadense) (FR-CA). A voz é de alta qualidade e está disponível somente como voz NTTS. Como todas as vozes neurais, ela está disponível somente em determinadas regiões. Para obter uma lista de regiões, consulte Compatibilidade de recursos e regiões .	1º de junho de 2021
Nova voz adicionada para NTTS (p. 249)	Amazon Polly agora fornece a voz coreana Seoyeon como uma voz NTTS.	11 de maio de 2021
Nova região adicionada para NTTS (p. 249)	Agora o Amazon Polly oferece suporte a TTS neural (NTTS) no Canadá (Central)AWSRegião : Para obter mais informações, consulte Compatibilidade de recursos e regiões Para NTTS.	17 de março de 2021
Nova voz disponível para o estilo newscaster (p. 249)	Além das vozes de Matthew, Joanna e Lupe para o estilo de fala de locutor, o Amazon Polly oferece uma opção adicional para esse estilo de fala. Usando o mecanismo neural, você pode	10 de novembro de 2020

	usar a voz de Amy em inglês britânico para o estilo de locutor. Para obter mais informações, consulte Estilos de fala NTTS .	
Novas regiões adicionadas para NTTS (p. 249)	Além das regiões existentes para NTTS (us-east-1, us-west-2, eu-west-1 e ap-southeast-2), vozes neurais são agora compatíveis em quatro regiões adicionais: (ap-northeast-1 (Tóquio), ap-northeast-1 (Singapura), eu-central-1 (Frankfurt) e eu-west-2 (Londres). Para obter mais informações, consulte Compatibilidade de recursos e regiões Para NTTS.	3 de setembro de 2020
Nova voz adicionada (p. 249)	Além das vozes infantis de Ivy e Justin, uma nova voz masculina, Kevin, foi adicionada ao inglês dos EUA (en-US). Essa nova voz é de alta qualidade e está disponível somente como voz NTTS. Como todas as vozes neurais, ela só é compatível em quatro regiões: us-east-1 (Virgínia do Norte), us-west-2 (Oregon), eu-west-1 (Irlanda) e ap-southeast-2 (Sydney). Para obter mais informações, consulte Vozes NTTS .	16 de junho de 2020
Nova voz disponível para o estilo newscaster (p. 249)	Além das vozes de Matthew e Joanna para o estilo de fala de locutor, o Amazon Polly oferece uma opção adicional para esse estilo de fala. Usando o mecanismo neural, você pode usar a voz de Lupe em espanhol (EUA) para o estilo de locutor. Para obter mais informações, consulte Estilos de fala NTTS .	16 de abril de 2020
Novo recurso (p. 249)	Além do estilo de fala de locutor, o Amazon Polly agora fornece um segundo estilo de fala NTTS para ajudar você a sintetizar ainda melhor texto para passagens de fala. O estilo de conversação usa o sistema neural para gerar fala em um estilo de conversação mais amigável e expressivo que pode ser usado em muitos casos de uso. Para obter mais informações, consulte Estilos de fala NTTS .	25 de novembro de 2019

Novas vozes adicionadas (p. 249)	Duas novas vozes adicionadas: Camila (feminina, português-Brasil) e Lupe (feminina, espanhol-EUA).	23 de outubro de 2019
Novo recurso adicionado (p. 249)	Adição de Plugin Amazon Polly para Windows Para incorporar toda a gama de vozes do Amazon Polly em aplicativos compatíveis com a SAPI do Windows.	26 de setembro de 2019
Novo recurso (p. 249)	Além das vozes padrão de conversão de texto em fala (TTS) compatíveis pelo Amazon Polly desde seu lançamento, o Amazon Polly oferece agora um sistema de TTS neural aprimorado (NTTS) que pode fornecer vozes de qualidade ainda mais alta, fornecendo a você vozes de conversão de texto em fala mais naturais e semelhantes a vozes humanas possíveis. Para obter mais informações, consulte Conversão de texto em fala neural .	30 de julho de 2019
Novas vozes adicionadas (p. 249)	Novas vozes adicionadas: Lúcia (feminina, espanhol) e Bianca (feminina, italiano).	2 de agosto de 2018
Novo idioma adicionado (p. 249)	Novo idioma adicionado: Espanhol mexicano (es-MX). Este idioma usa a voz feminina de Mia.	2 de agosto de 2018
Novo idioma adicionado (p. 249)	Novo idioma adicionado: Hindi (hi-IN). Essa voz usa a voz feminina de Aditi, que também é usada para o inglês indiano, o que torna a Aditi a primeira voz bilíngue do Amazon Polly.	2 de agosto de 2018
Novo recurso adicionado (p. 249)	Adição da Síntese de fala de passagens de texto longas (até 100.000 caracteres cobrados).	17 de julho de 2018
Novo recurso SSML adicionado (p. 249)	Adição da Duração máxima para fala sintetizada .	17 de julho de 2018
Nova voz adicionada (p. 249)	Nova voz adicionada: Léa (feminina, francês).	5 de junho de 2018
Expansão de região (p. 249)	Expansão de serviço do Amazon Polly a todas as regiões comerciais.	4 de junho de 2018

Novo idioma adicionado (p. 249)	Novo idioma adicionado: Coreano (ko-KR).	4 de junho de 2018
Recurso expandido (p. 249)	Expansão do recurso do Amazon Polly WordPress de plug-in, incluindo a adição de recursos do Amazon Translate.	4 de junho de 2018
Novas vozes adicionadas (p. 249)	Duas novas vozes adicionadas: Aditi (feminina, inglês indiano) e Seoyeon (feminina, coreano).	15 de novembro de 2017
Novo recurso (p. 249)	Adição do novo recurso Marcas de fala , bem como uma expansão dos recursos SSML .	19 de abril de 2017
Novo guia (p. 249)	Esta é a primeira versão do Guia do desenvolvedor do Amazon Polly.	30 de novembro de 2016

AWSGlossário

Para a mais recente AWS terminologia, consulte o [AWSGlossário](#) no AWS Referência geral.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.