

Quickstart: Convert text to speech

Article • 09/18/2024

[Reference documentation](#) | [Package \(npm\)](#) | [Additional samples on GitHub](#) | [Library source code](#)

With Azure AI Speech, you can run an application that synthesizes a human-like voice to read text. You can change the voice, enter text to be spoken, and listen to the output on your computer's speaker.

Tip

You can try text to speech in the [Speech Studio Voice Gallery](#) without signing up or writing any code.

Prerequisites

- ✓ An Azure subscription. You can [create one for free](#).
- ✓ [Create a Speech resource](#) in the Azure portal.
- ✓ Get the Speech resource key and region. After your Speech resource is deployed, select **Go to resource** to view and manage keys.

Set up the environment

To set up your environment, install the Speech SDK for JavaScript. If you just want the package name to install, run `npm install microsoft-cognitiveservices-speech-sdk`. For detailed installation instructions, see [Install the Speech SDK](#).

Set environment variables

You need to authenticate your application to access Azure AI services. This article shows you how to use environment variables to store your credentials. You can then access the environment variables from your code to authenticate your application. For production, use a more secure way to store and access your credentials.

Important

We recommend Microsoft Entra ID authentication with [managed identities for Azure resources](#) to avoid storing credentials with your applications that run in the cloud.

If you use an API key, store it securely somewhere else, such as in [Azure Key Vault](#). Don't include the API key directly in your code, and never post it publicly.

For more information about AI services security, see [Authenticate requests to Azure AI services](#).

To set the environment variables for your Speech resource key and region, open a console window, and follow the instructions for your operating system and development environment.

- To set the `SPEECH_KEY` environment variable, replace *your-key* with one of the keys for your resource.
- To set the `SPEECH_REGION` environment variable, replace *your-region* with one of the regions for your resource.

Windows

Console

```
setx SPEECH_KEY your-key  
setx SPEECH_REGION your-region
```

Note

If you only need to access the environment variables in the current console, you can set the environment variable with `set` instead of `setx`.

After you add the environment variables, you might need to restart any programs that need to read the environment variables, including the console window. For example, if you're using Visual Studio as your editor, restart Visual Studio before you run the example.

Create the application

Follow these steps to create a Node.js console application for speech synthesis.

1. Open a console window where you want the new project, and create a file named *SpeechSynthesis.js*.
2. Install the Speech SDK for JavaScript:

Console

```
npm install microsoft-cognitiveservices-speech-sdk
```

3. Copy the following code into *SpeechSynthesis.js*:

JavaScript

```
(function() {  
  
    "use strict";  
  
    var sdk = require("microsoft-cognitiveservices-speech-sdk");  
    var readline = require("readline");  
  
    var audioFile = "YourAudioFile.wav";  
    // This example requires environment variables named "SPEECH_KEY" and  
    "SPEECH_REGION"  
    const speechConfig =  
    sdk.SpeechConfig.fromSubscription(process.env.SPEECH_KEY,  
    process.env.SPEECH_REGION);  
    const audioConfig = sdk.AudioConfig.fromAudioFileOutput(audioFile);  
  
    // The language of the voice that speaks.  
    speechConfig.speechSynthesisVoiceName = "en-US-AvaMultilingualNeural";  
  
    // Create the speech synthesizer.  
    var synthesizer = new sdk.SpeechSynthesizer(speechConfig, audioCon-  
fig);  
  
    var rl = readline.createInterface({  
        input: process.stdin,  
        output: process.stdout  
    });  
  
    rl.question("Enter some text that you want to speak >\n> ", function  
(text) {  
        rl.close();
```

```
// Start the synthesizer and wait for a result.
synthesizer.speakTextAsync(text,
    function (result) {
        if (result.reason === sdk.ResultReason.SynthesizingAudioCompleted)
        {
            console.log("synthesis finished.");
        } else {
            console.error("Speech synthesis canceled, " +
result.errorDetails +
                "\nDid you set the speech resource key and region values?");
        }
        synthesizer.close();
        synthesizer = null;
    },
    function (err) {
        console.trace("err - " + err);
        synthesizer.close();
        synthesizer = null;
    });
    console.log("Now synthesizing to: " + audioFile);
});
})();
```

4. In *SpeechSynthesis.js*, optionally you can rename *YourAudioFile.wav* to another output file name.
5. To change the speech synthesis language, replace `en-US-AvaMultilingualNeural` with another [supported voice](#).

All neural voices are multilingual and fluent in their own language and English. For example, if the input text in English is *I'm excited to try text to speech* and you set `es-ES-ElviraNeural`, the text is spoken in English with a Spanish accent. If the voice doesn't speak the language of the input text, the Speech service doesn't output synthesized audio.

6. Run your console application to start speech synthesis to a file:

Console

```
node SpeechSynthesis.js
```

Important

Make sure that you set the `SPEECH_KEY` and `SPEECH_REGION` [environment variables](#). If you don't set these variables, the sample fails with an error message.

7. The provided text should be in an audio file:

Output

```
Enter some text that you want to speak >  
> I'm excited to try text to speech  
Now synthesizing to: YourAudioFile.wav  
synthesis finished.
```

Remarks

More speech synthesis options

This quickstart uses the `SpeakTextAsync` operation to synthesize a short block of text that you enter. You can also use long-form text from a file and get finer control over voice styles, prosody, and other settings.

- See [how to synthesize speech](#) and [Speech Synthesis Markup Language \(SSML\) overview](#) for information about speech synthesis from a file and finer control over voice styles, prosody, and other settings.
- See [batch synthesis API for text to speech](#) for information about synthesizing long-form text to speech.

OpenAI text to speech voices in Azure AI Speech

OpenAI text to speech voices are also supported. See [OpenAI text to speech voices in Azure AI Speech](#) and [multilingual voices](#). You can replace `en-US-AvaMultilingualNeural` with a supported OpenAI voice name such as `en-US-FableMultilingualNeural`.

Clean up resources

You can use the [Azure portal](#) or [Azure Command Line Interface \(CLI\)](#) to remove the Speech resource you created.

Next step

[Learn more about speech synthesis](#)

Feedback

Was this page helpful?

 **Yes**

 **No**

[Provide product feedback](#) | [Get help at Microsoft Q&A](#)