

# Speech to Text Transcription with the Cloud Speech API

**GSP048**



Google Cloud Self-Paced Labs

# Overview

The Cloud Speech API lets you do speech to text transcription from audio files in over 80 languages.

In this lab you'll send an audio file to the Cloud Speech API for transcription.

## What you'll learn

- Creating a Speech API request and calling the API with curl
- Calling the Speech API with audio files in different languages

## What you'll need

- A Google Cloud Project
- A Browser, such [Chrome](#) or [Firefox](#)

# Setup

## Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This Qwiklabs hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

## What you need

To complete this lab, you need:

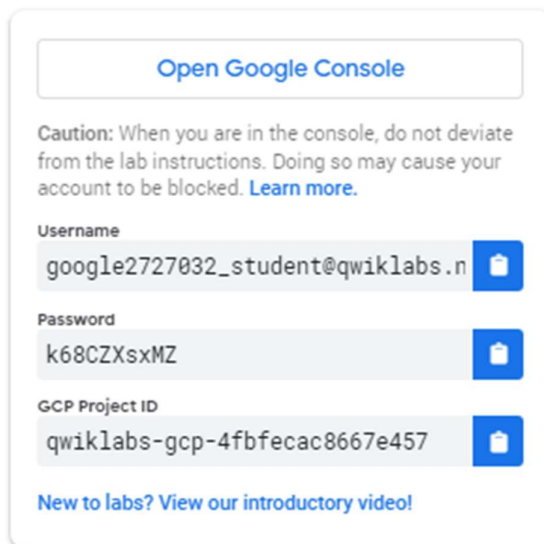
- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.

**Note:** If you already have your own personal Google Cloud account or project, do not use it for this lab.

**Note:** If you are using a Pixelbook, open an Incognito window to run this lab.


## How to start your lab and sign in to the Google Cloud Console


1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is a panel populated with the temporary credentials that you must use for this lab.




[Open Google Console](#)

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

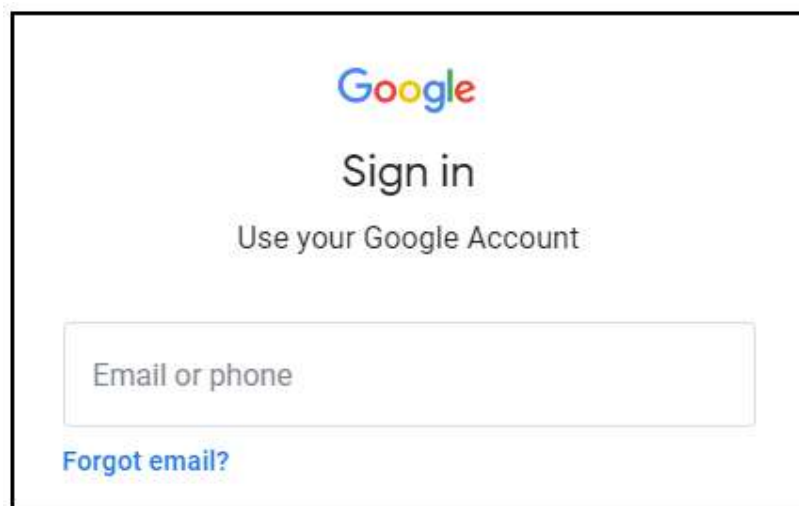
Username  
google2727032\_student@qwiklabs.n 

Password  
k68CZXsxMZ 

GCP Project ID  
qwiklabs-gcp-4fbfecac8667e457 

[New to labs? View our introductory video!](#)

2. Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Sign in** page.



Google

Sign in

Use your Google Account

Email or phone

[Forgot email?](#)

**Tip:** Open the tabs in separate windows, side-by-side.

If you see the **Choose an account** page, click **Use Another**



**Account.**

3. In the **Sign in** page, paste the username that you copied from the Connection Details panel. Then copy and paste the password.

**Important:** You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

4. Click through the subsequent pages:

- Accept the terms and conditions.
- Do not add recovery options or two-factor authentication (because this is a temporary account).
- Do not sign up for free trials.

After a few moments, the Cloud Console opens in this tab.

**Note:** You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.

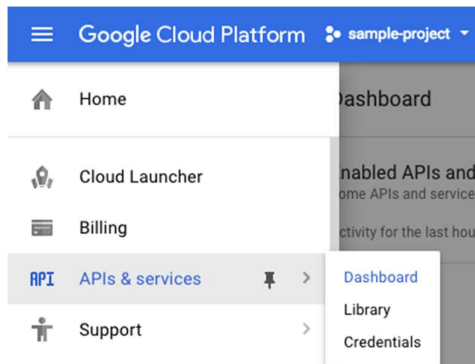


# Create an API key

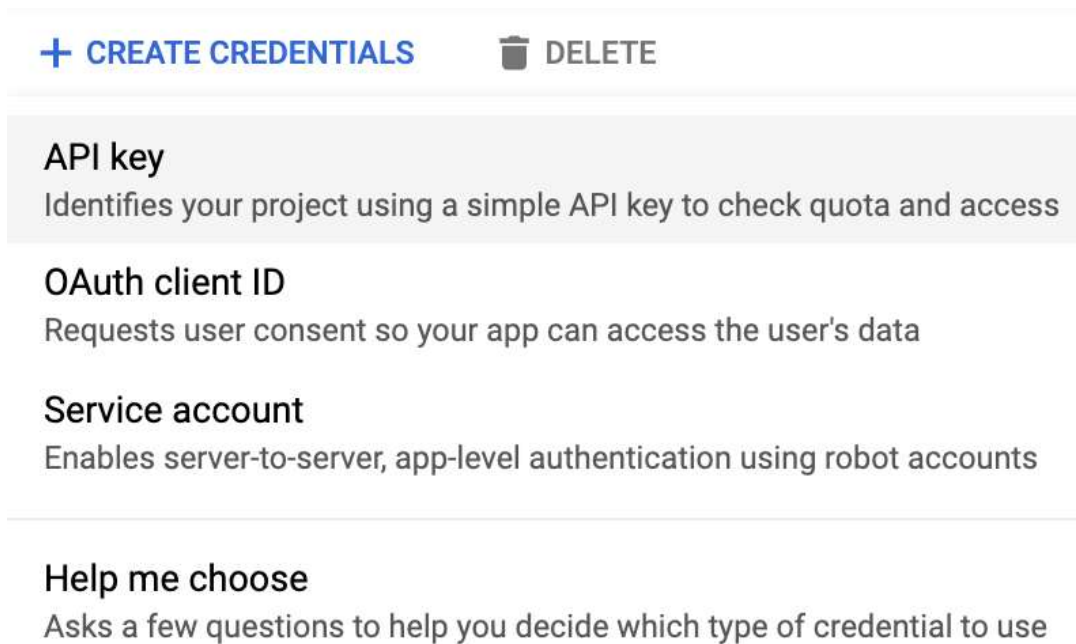
Since you'll be using `curl` to send a request to the Speech API, you'll need to generate an API key to pass in your request URL.

To create an API key, navigate to:

**APIs & services > Credentials:**



Click **Create credentials** and select **API key**



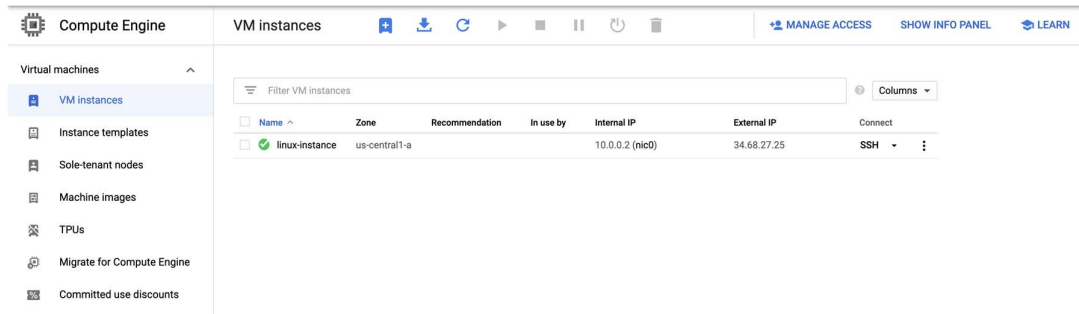
Next, copy the key you just generated. Click **Close**.

Click **Check my progress** to verify the objective.

Create an API Key  
Check my progress

Now save your key to an environment variable to avoid having to insert the value of your API key in each request.

In order to perform next steps please connect to the **linux-instance** provisioned for you via **SSH**:



Click on **SSH** button, you'll see an interactive shell, use this for performing next operations.

In the shell (SSH) run the following, replacing `<your_api_key>` with the key you just copied:

```
export API_KEY=<YOUR_API_KEY>
```

## Create your Speech API request

**Note:** This lab uses a pre-recorded file that's available on Cloud Storage: `gs://cloud-samples-tests/speech/brooklyn.flac`. You can listen to this file before sending it to the Speech API [here](#).

Build your request to the Speech API in a `request.json` file:

```
touch request.json
```

Open the file using your preferred command line editor (`nano`, `vim`, `emacs`) or `gcloud`. Add the following to your `request.json` file, using the `uri` value of the sample raw audio file:

```
{
  "config": {
    "encoding": "FLAC",
    "languageCode": "en-US"
  },
  "audio": {
    "uri": "gs://cloud-samples-tests/speech/brooklyn.flac"
  }
}
```

The request body has a `config` and `audio` object.

In `config`, you tell the Speech API how to process the request:

- The `encoding` parameter tells the API which type of audio encoding you're using while the file is being sent to the API. `FLAC` is the encoding type for `.raw` files (here is [documentation](#) for encoding types for more details).
- `languageCode` will default to English if left out of the request.  
There are other parameters you can add to your `config` object, but `encoding` is the only required one.

In the `audio` object, you pass the API the uri of the audio file which is stored in Cloud Storage for this lab.

Click **Check my progress** to verify the objective.

Create your Speech API request

Check my progress

Now you're ready to call the Speech API!

# Call the Speech API

Pass your request body, along with the API key environment variable, to the Speech API with the following `curl` command (all in one single command line):

```
curl -s -X POST -H "Content-Type: application/json" --data-binary @request.json \
"https://speech.googleapis.com/v1/speech:recognize?key=${API_KEY}" > result.json
```

Your response is stored in a file named as `result.json`. In order to see the contents of file you can use:

```
cat result.json
```

The response returned by the `curl` command look something like this:

```
{
  "results": [
    {
      "alternatives": [
        {
          "transcript": "how old is the Brooklyn Bridge",
          "confidence": 0.98267895
        }
      ]
    }
  ]
}
```

The `transcript` value will return the Speech API's text transcription of your audio file, and the `confidence` value indicates how sure the API is that it has accurately transcribed your audio.

Notice that you called the `syncrecognize` method in our request above. The Speech API supports both synchronous and asynchronous speech to text transcription. In this example a complete audio file was used, but you can also use the `syncrecognize` method to perform streaming speech to text transcription while the user is still speaking.

Click **Check my progress** to verify the objective.

Call the Speech API

Check my progress



# Speech to text transcription in different languages

Are you multilingual? The Speech API supports speech to text transcription in over 100 languages! You can change the `language_code` parameter in `request.json`. You can find a list of supported languages [here](#).

Let's try a French audio file (listen to it [here](#) if you'd like a preview).

Edit your `request.json` and change the content to the following:

```
{
  "config": {
    "encoding": "FLAC",
    "languageCode": "fr"
  },
  "audio": {
    "uri": "gs://speech-language-samples/fr-sample.flac"
  }
}
```

Now call the Speech API by running the `curl` command again.

You should see the following response:

```
{
  "results": [
    {
      "alternatives": [
        {
          "transcript": "maître corbeau sur un arbre perché tenait en son bec un fromage",
          "confidence": 0.9710122
        }
      ]
    }
  ]
}
```

This is a sentence from a popular French [children's tale](#). If you've got audio files in another language, you can try adding them to Cloud Storage and changing the `languageCode` parameter in your request.

API restrictions and usage limits on Cloud Speech-to-Text are documented here: <https://cloud.google.com/speech-to-text/quotas>

# Congratulations!

You've learned how to perform speech to text transcription with the Speech API. In this lab you passed the API the Cloud Storage URI of your audio file. Alternatively, you can pass a base64 encoded string of your audio content.

## What was covered

- Passing the Speech API a Cloud Storage URI of an audio file
- Creating a Speech API request and calling the API with curl
- Calling the Speech API with audio files in different languages



## Finish your quest

This self-paced lab is part of the Qwiklabs [Intro to ML: Language Processing](#) Quest. A Quest is a series of related labs that form a learning path. Completing a Quest earns you a badge to recognize your achievement. You can make your badge (or badges) public and link to them in your online resume or social media account. Enroll in the above Quest and get immediate completion credit if you've taken this lab. [See other available Qwiklabs Quests](#).

## Next steps / learn more

- Check out the Speech API [tutorials](#) in the documentations.
- Try out the [Vision API](#) and [Natural Language API](#)!

## Google Cloud Training & Certification

...helps you make the most of Google Cloud technologies. [Our classes](#) include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. [Certifications](#) help you validate and prove your skill and expertise in Google Cloud technologies.

Manual Last Updated February 19, 2021

Lab Last Tested February 19, 2021

Copyright 2021 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.