Hello there. My name is Kevin Lewis,Ð
and I'm a developer ad for here at Deepgram. And today, I'm gonna show you how to get a transcript from a Youtube video using Deepgram speech recognition Api. This project has three parts. In the first part, we'll be downloading a Youtube video to a local P three file. In the second part, will take that m mp three file and send it to Deepgram to get a nice and accurate transcript. And then finally, we'll take that transcript and store it in a text file on our computer. I've just gone ahead and installed the dependencies we're gonna use today and required them in this index Js file. So we're using the file system built in library. We have this package called Youtube P three download, which surprise surprise we'll download a Youtube video. We have the Deepgram node js s s sdk. And finally, we have F mp static F peg is an audio processing and manipulation utility, an F peg static that makes the executable available inside of our project. So let's crack on. We're ready to go. The first thing we're gonna do is download a Youtube video. So we've required Youtube p three download, Now we're gonna go ahead and initialize it. And we initialize it with a few settings, a link directly to our executable. Specifying that we want the Mp three file to be put alongside the index Js file. And finally, specifying that we want to prioritize the highest audio quality possible. Then to download it, we can just go ahead and provide a Youtube video Id. This is of a movie trailer. So this is all we need to download a Youtube video. There is something else we need to do here, though. We need to know when it's finished. Because we don't wanna send it to Deepgram until we have the cÐ
ompleted file. Fortunately, this Youtube p three download a package emiÐ
ts an event when the download has been completed. So we can listen for that event, Ð
finished. And what is returned once it is downloaded, is a object in this video parameter herÐ
e, which contains some metadata about the file. We really only care Ð
about the video file name. Because we'll need that later. What we'll do here is just console log the the video file name, has been downloadedÐ
. So we'll run this and we'll just check that that successfully downloads. Ð
Send you a short video, so it just took a moment there. But that is our m p threeÐ
file downloaded. Now in the next step, we wanna take this file and provide it to Deepgram. So the first thing we Ð
wanna do is take this required node Js S sdk and initialize it. And we'll initializÐ
e it with our Api key. Which I have stored here in an environment variable. Ð
Then we're ready to ask for a pre recorded transcription. We need to provide the file name hereÐ
, which we stored video, file name. And you can provide any features here that youÐ
want. We have a whole list of features in our documentation. I'm gonna use pun eight and I'm also going to use utterances, which will return whicÐ
h will return phrases as well as words. Then we're gonna go ahead here and console log the resulÐ
t. And I'm just gonna use our, captions generator here two web V to make it a little easier to see inside of thÐ
e terminal. So that is now redo downloading the M p three file going off to Deepgram, coming back and therÐ
e is our brand new transcription. So the final step is to save this text right here in ouÐ

r terminal to a text file. So we won't console log it. Instead, we're gonna go aheÐ
ad and save it. All we need to do here is use the file system module, write file sync. The firsÐ
t thing we need to provide is the file name, and the file name is going to be video file name. AÐ
nd then we'll just add to the end of that dot t x t. And we also want to proÐ
vide the text, So we'll just do to web Bt again. We'll run this one final time, aÐ
nd it's gonna go grab a brand new m p three for us. Go off to Deepgram and tÐ
hen provide our captions right here in this file. I hope you found thÐ
at interesting, and if you have any questions at all, Please feel free to reach ouÐ
t to us. We love answering questions and helping you build great projects with voice. Bye for now.