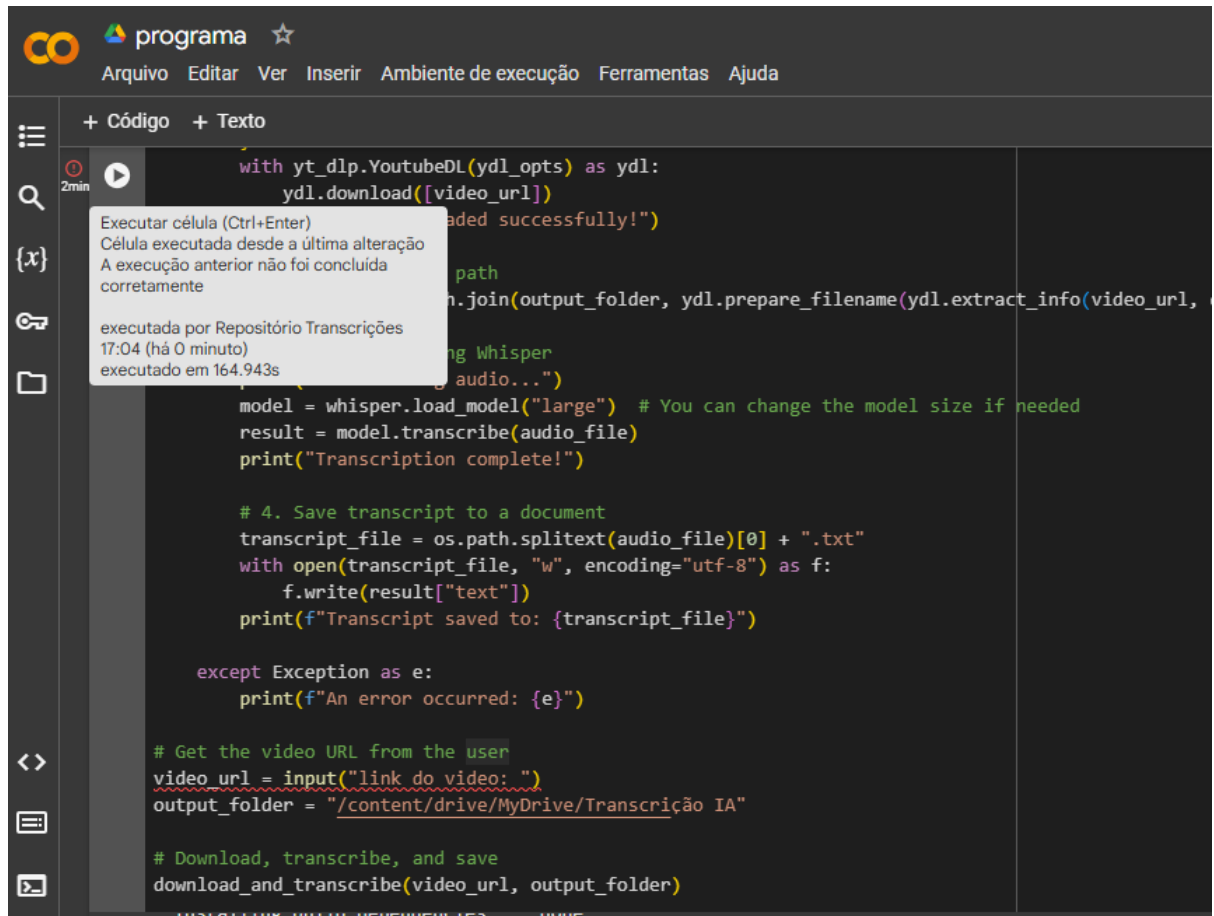


1. Acesse o código - transcrever videos do youtube - que está dentro dessa mesma pasta
- 2.



The screenshot shows the Google Colab web interface. At the top, there's a menu bar with 'Arquivo', 'Editar', 'Ver', 'Inserir', 'Ambiente de execução', 'Ferramentas', and 'Ajuda'. Below the menu, there's a toolbar with '+ Código' and '+ Texto'. The main area contains a Python script for transcribing YouTube videos. A tooltip is visible over the 'Executar célula' button, showing the following text:

```
Executar célula (Ctrl+Enter)
Célula executada desde a última alteração
A execução anterior não foi concluída
corretamente
executada por Repositório Transcrições
17:04 (há 0 minuto)
executado em 164.943s
```

```
with yt_dlp.YoutubeDL(ydl_opts) as ydl:
    ydl.download([video_url])
    print("Downloaded successfully!")

# Save the transcript to a document
path = os.path.join(output_folder, ydl.prepare_filename(ydl.extract_info(video_url,
    download=True), 'audio...'))

# Load the Whisper model
model = whisper.load_model("large") # You can change the model size if needed
result = model.transcribe(audio_file)
print("Transcription complete!")

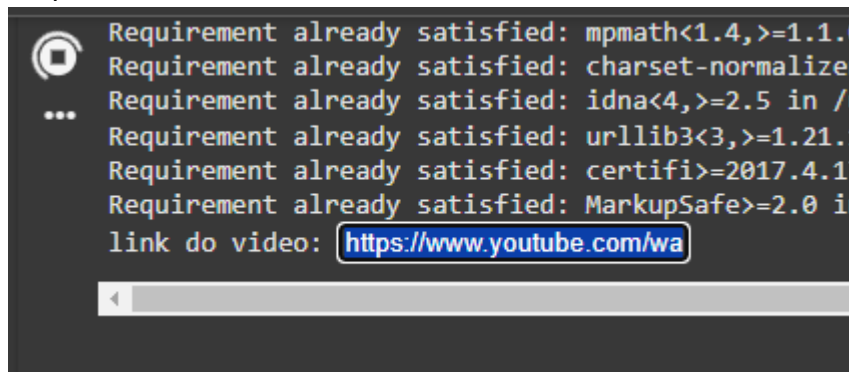
# 4. Save transcript to a document
transcript_file = os.path.splitext(audio_file)[0] + ".txt"
with open(transcript_file, "w", encoding="utf-8") as f:
    f.write(result["text"])
print(f"Transcript saved to: {transcript_file}")

except Exception as e:
    print(f"An error occurred: {e}")

# Get the video URL from the user
video_url = input("link do video: ")
output_folder = "/content/drive/MyDrive/Transcrição IA"

# Download, transcribe, and save
download_and_transcribe(video_url, output_folder)
```

Clique em Executar Célula



The screenshot shows the Google Colab terminal output. The output displays the requirements for the libraries and the video URL entered by the user:

```
Requirement already satisfied: mpmath<1.4,>=1.1.0
Requirement already satisfied: charset-normalizer
Requirement already satisfied: idna<4,>=2.5 in /u
Requirement already satisfied: urllib3<3,>=1.21.1
Requirement already satisfied: certifi>=2017.4.17
Requirement already satisfied: MarkupSafe>=2.0 in
link do video: https://www.youtube.com/wa
```

3. - Adicione o link do youtube ou da playlist para ser transcrita.

4.

```
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>ope
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>ope
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages (from torch>ope
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.10/dist-packages (from tor
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (fro
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-package
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requ
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (fro
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (fro
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from j
link do video: https://www.youtube.com/watch?v=C4YnJADAlw0&ab\_channel=LamaPadmaSamten-AltV%C3%ADo
Downloading audio...
Audio downloaded successfully!
Transcribing audio...
17% | 494M/2.88G [00:03<00:20, 123MiB/s]
```

Aguarde a finalização

5. Confira na pasta - Transcrição IA se o arquivo de áudio e o txt estão disponíveis.
6. Retiro estes arquivos para a sua pasta e finalize a transcrição.

Qualquer duvida, falar com Bruno  
48 - 991486176

Codigo guardado (nao alterar)

```
# Install necessary libraries
!pip install yt-dlp
!pip install git+https://github.com/openai/whisper.git

import yt_dlp
import whisper
import os

def download_and_transcribe(video_url, output_folder):
    """Downloads YouTube audio, transcribes it, and saves the
    transcript.

    Args:
        video_url: The URL of the YouTube video.
        output_folder: The path to the folder where the audio and
        transcript should be saved.
    """
    try:
        # 1. Download audio using yt-dlp
        print("Downloading audio...")
        os.makedirs(output_folder, exist_ok=True)
        ydl_opts = {
            'format': 'bestaudio[ext=m4a]',
            'outtmpl': os.path.join(output_folder, '%(title)s.%(ext)s'),
            'extract-audio': True,
            'audio-format': 'm4a',
            'quiet': True
        }
        with yt_dlp.YoutubeDL(ydl_opts) as ydl:
            ydl.download([video_url])
        print("Audio downloaded successfully!")

        # 2. Get audio file path
        audio_file = os.path.join(output_folder,
            ydl.prepare_filename(ydl.extract_info(video_url, download=False)))

        # 3. Transcribe using Whisper
        print("Transcribing audio...")
        model = whisper.load_model("large") # Removed weights_only
        here
        result = model.transcribe(audio_file)
```

```
print("Transcription complete!")

# 4. Save transcript to a document
transcript_file = os.path.splitext(audio_file)[0] + ".txt"
with open(transcript_file, "w", encoding="utf-8") as f:
    f.write(result["text"])
print(f"Transcript saved to: {transcript_file}")

except Exception as e:
    print(f"An error occurred: {e}")

# Get the video URL from the user
video_url = input("link do video: ")
output_folder = "/content/drive/MyDrive/#Transcrição IA"

# Download, transcribe, and save
download_and_transcribe(video_url, output_folder)
```