



[125]

✓ 10s

Cell 1: install libraries

```
!pip install SpeechRecognition==3.8.1 pydub gTTS==2.3.2 numpy
!apt-get install -y ffmpeg # pydub needs ffmpeg to convert audio
```

▼

```
Requirement already satisfied: SpeechRecognition==3.8.1 in /usr/local/lib/python3.12/dist-packages (3.8.1)
Requirement already satisfied: pydub in /usr/local/lib/python3.12/dist-packages (0.25.1)
Requirement already satisfied: gTTS==2.3.2 in /usr/local/lib/python3.12/dist-packages (2.3.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.12/dist-packages (2.0.2)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.12/dist-packages (from gTTS==2.3.2) (2.32.4)
Requirement already satisfied: click<8.2,>=7.1 in /usr/local/lib/python3.12/dist-packages (from gTTS==2.3.2) (8.1.8)
Requirement already satisfied: charset_normalizer<4,>=2 in /usr/local/lib/python3.12/dist-packages (from requests<3,>=2.27->gTTS==2.3.2) (3.4.4)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.12/dist-packages (from requests<3,>=2.27->gTTS==2.3.2) (3.11)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.12/dist-packages (from requests<3,>=2.27->gTTS==2.3.2) (2.5.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.12/dist-packages (from requests<3,>=2.27->gTTS==2.3.2) (2025.10.5)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ffmpeg is already the newest version (7:4.4.2-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.
```



[126]

✓ 0s



```
from google.colab import output
from IPython.display import Javascript, display

def record_audio():
    display(Javascript("""
const sleep = time => new Promise(resolve => setTimeout(resolve, time));
const startButton = document.createElement('button');
startButton.textContent = '🎤 Start Recording';
document.body.appendChild(startButton);

const stopButton = document.createElement('button');
```





Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM
Disk[126]
✓ 0s

```
const stopButton = document.createElement('button');
stopButton.textContent = '🛑 Stop Recording';
stopButton.disabled = true;
document.body.appendChild(stopButton);

let recorder, audio;

startButton.onclick = async () => {
  startButton.disabled = true;
  stopButton.disabled = false;
  const stream = await navigator.mediaDevices.getUserMedia({ audio: true });
  recorder = new MediaRecorder(stream);
  const chunks = [];
  recorder.ondataavailable = e => chunks.push(e.data);
  recorder.onstop = e => {
    audio = new Blob(chunks);
    const reader = new FileReader();
    reader.readAsDataURL(audio);
    reader.onloadend = () => {
      const base64Audio = reader.result.split(',')[1];
      google.colab.kernel.invokeFunction('upload_audio', [base64Audio], {});
    };
  };
  recorder.start();
};

stopButton.onclick = () => {
  stopButton.disabled = true;
  recorder.stop();
  alert("Recording saved as 'recorded.wav'");
};
""")
```



{ } Variables Terminal



✓ 21:39 Python 3



Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM Disk 

```
ffmpeg version 4.4.2-0ubuntu0.22.04.1 Copyright (c) 2000-2021 the FFmpeg developers
built with gcc 11 (Ubuntu 11.2.0-19ubuntu1)
configuration: --prefix=/usr --extra-version=0ubuntu0.22.04.1 --toolchain=hardened --libdir=/usr/lib/x86_64-linux-gnu --incdir=/usr/include/x86_64-linux-g
libavutil      56. 70.100 / 56. 70.100
libavcodec     58.134.100 / 58.134.100
libavformat    58. 76.100 / 58. 76.100
libavdevice    58. 13.100 / 58. 13.100
libavfilter    7.110.100 / 7.110.100
libswscale     5.  9.100 / 5.  9.100
libswresample  3.  9.100 / 3.  9.100
libpostproc   55.  9.100 / 55.  9.100
Input #0, matroska,webm, from 'recorded.wav':
Metadata:
  encoder      : Chrome
Duration: 00:00:04.30, start: 0.000000, bitrate: 129 kb/s
Stream #0:0(eng): Audio: opus, 48000 Hz, mono, fltp (default)
Stream mapping:
  Stream #0:0 -> #0:0 (opus (native) -> pcm_s16le (native))
Press [q] to stop, [?] for help
Output #0, wav, to 'converted.wav':
Metadata:
  ISFT        : Lavf58.76.100
Stream #0:0(eng): Audio: pcm_s16le ([1][0][0][0] / 0x0001), 48000 Hz, mono, s16, 768 kb/s (default)
Metadata:
  encoder      : Lavc58.134.100 pcm_s16le
size=      405kB time=00:00:04.29 bitrate= 771.8kbits/s speed= 210x
video:0kB audio:405kB subtitle:0kB other streams:0kB global headers:0kB muxing overhead: 0.018808%
```

[129]

✓ 1s

```
import speech_recognition as sr
import os

r = sr.Recognizer()
```

{ Variables Terminal



✓ 21:39 Python 3



Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM Disk 

```
[126]
✓ 0s
import base64

def _upload_audio(b64):
    audio_bytes = base64.b64decode(b64)
    with open("recorded.wav", "wb") as f:
        f.write(audio_bytes)
    print("File saved: recorded.wav")

output.register_callback('upload_audio', _upload_audio)

record_audio()
```



⋮ Start Recording Stop Recording

```
[127]
✓ 1s
# Cell 3: play the recorded audio
from IPython.display import Audio, display
import os
if os.path.exists('recorded.wav'):
    display(Audio('recorded.wav', autoplay=False))
else:
    print("No 'recorded.wav' found. Run the recorder cell first.")
```

▶ 0:00 / 0:04 🔊 ⋮

```
[128]
✓ 0s
# Convert recorded audio to real WAV using ffmpeg
!ffmpeg -y -i recorded.wav converted.wav
```

{ } Variables Terminal



✓ 21:39 Python 3



Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM Disk

[129]
✓ 1s

```
if not os.path.exists("converted.wav"):
    raise FileNotFoundError("converted.wav not found. Run the ffmpeg convert cell first.")

with sr.AudioFile("converted.wav") as source:
    audio = r.record(source)

try:
    text = r.recognize_google(audio)
    print("Recognized Text:", text)
except Exception as e:
    print("Speech Recognition Error:", e)
    text = ""
```

▼ Recognized Text: medical timer

[130]
✓ 0s

```
# ★ CELL 5: INTENT RECOGNITION (Hindi + English + Time + Links)
import datetime
import random
import pytz
from IPython.display import HTML, display

def process_command(command):
    cmd = command.lower().strip()

    # No input
    if cmd == "":
        return "NO_INPUT", "I didn't catch that, please speak again."

    # ===== TIME COMMAND =====
    if "time" in cmd or "samay" in cmd:
        india = pytz.timezone("Asia/Kolkata")
```

{ } Variables Terminal



✓ 21:39 Python 3



Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM
Disk[130]
✓ 0s

```
india = pytz.timezone("Asia/Kolkata")
now = datetime.datetime.now(india).strftime("%I:%M %p")
return "TELL_TIME", f"India mein samay ho raha hai: {now}"

# ===== GOOGLE OPEN =====
if "google" in cmd:
    url = "https://www.google.com"
    display(HTML(f'<a href="{url}" target="_blank">Click here to open Google</a>'))
    return "OPEN_GOOGLE", "Opening Google. Click the link above."

# ===== YOUTUBE OPEN =====
if "youtube" in cmd:
    url = "https://www.youtube.com"
    display(HTML(f'<a href="{url}" target="_blank">Click here to open YouTube</a>'))
    return "OPEN_YOUTUBE", "Opening YouTube. Click the link above."

# ===== ENGLISH JOKES (Random) =====
if ("joke" in cmd) and ("hindi" not in cmd):
    english_jokes = [
        "Why don't skeletons fight each other? Because they don't have the guts.",
        "Why did the computer go to the doctor? Because it had a virus!",
        "I told my laptop I needed a break. It instantly shut down!",
        "Why was the math book sad? Because it had too many problems!",
        "Why don't eggs tell jokes? They'd crack each other up!",
        "Parallel lines have so much in common. It's a shame they never meet.",
        "What do you call fake spaghetti? An impasta!",
        "Why was the broom late? It swept in!",
        "Why don't bananas ever get lonely? Because they hang out in bunches!"
    ]
    return "ENGLISH_JOKE", random.choice(english_jokes)

# ===== HINDI JOKES (Random) =====
if ("hindi joke" in cmd) or ("joke in hindi" in cmd) or ("hindi mein joke" in cmd):
```

{ } Variables Terminal



✓ 21:39 Python 3



Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM
Disk[130]
✓ 0s

```
# ===== HINDI JOKES (Random) =====
if ("hindi joke" in cmd) or ("joke in hindi" in cmd) or ("hindi mein joke" in cmd):
    hindi_jokes = [
        "एक आदमी डॉक्टर के पास गया और बोला - डॉक्टर साहब, मुझे रात को सपने में चूहे क्रिकेट खेलते हुए दिखते हैं। डॉक्टर बोला - कोई दवाई दे देता हूँ। आदमी बोला - लेकिन कल म",
        "टीचर: दुनिया का सबसे पुराना तेल कौन सा है? छात्र: कच्चा तेल, क्योंकि पकाने वाला तो बाद में आया!",
        "पप्पू ATM में गया। मशीन ने लिखा: कृपया कार्ड डालें। पप्पू बोला: कार्ड तो डाल दिया, अब पैसे तुम डालो!",
        "पति: तुम मुझसे कितना प्यार करती हो? पत्नी: जान से भी ज्यादा। पति: तो जान कब देने वाली हो? पत्नी: जब तुम कहोगे... पति: अच्छा तो लाइट चालू कर दो!",
        "बच्चा: मम्मी मैं बड़ा होकर एलोन मस्क बनूँगा। मम्मी: पहले EVS की कॉपी दिखा!",
        "गोलू: मेरे फोन में नेटवर्क नहीं आ रहा। चाचा: एयरटेल डाल। गोलू: चाचा ये फोन है, भेलपुरी वाला ठेला नहीं!",
        "डॉक्टर: आपको आराम की जरूरत है। 7 दिन कुछ मत सोचिए। मरीज: डॉक्टर, फीस कौन देगा? डॉक्टर: अगला!",
        "राजू: भाई मेरी घड़ी बंद हो गई। पप्पू: तो खोल दे, घुटन हो रही होगी!",
        "एक आदमी भगवान से बोला: हे प्रभु, नौकरी दिला दो! भगवान: बेटा, पहले LinkedIn खोल। आदमी: प्रभु, पासवर्ड भूल गया!"
    ]
    return "HINDI_JOKE", random.choice(hindi_jokes)

# ===== GREETING =====
if "hello" in cmd or "hi" in cmd or "namaste" in cmd:
    return "GREETING", "Hello! How can I help you?"

# ===== DEFAULT UNKNOWN =====
return "UNKNOWN", "Sorry, I am not trained for that command yet."

# Run intent detection
intent, reply_text = process_command(text)
print("Intent:", intent)
print("Reply:", reply_text)
```

Intent: TELL_TIME
Reply: India mein samay ho raha hai: 09:39 PM

{ Variables Terminal



✓ 21:39 Python 3



Untitled1.ipynb ☆ ☁

File Edit View Insert Runtime Tools Help



Share



Commands + Code + Text ▶ Run all

✓ RAM Disk

[131]
✓ 0s

```
# CELL 6: TEXT-TO-SPEECH (Assistant replies using voice)
from gtts import gTTS
from IPython.display import Audio, display

def speak(text):
    """
    Convert the assistant's reply text into speech (mp3)
    and play it inside Google Colab.
    """
    tts = gTTS(text)
    tts.save("reply.mp3")
    display(Audio("reply.mp3", autoplay=True))

# Speak the assistant's reply
speak(reply_text)
```

▶ 0:05 / 0:05 ⋮

{ } Variables Terminal



✓ 21:39 Python 3