```
!pip install gTTs
```

```
Collecting gTTs
  Downloading gTTS-2.5.1-py3-none-any.whl (29 kB)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.10/dist-packages (from gTTs) (2.31.0)
Requirement already satisfied: click<8.2,>=7.1 in /usr/local/lib/python3.10/dist-packages (from gTTs) (8.1.7)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->gTTs) (3.3.2
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->gTTs) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->gTTs) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->gTTs) (2024.2.2)
Installing collected packages: gTTs
Successfully installed gTTs-2.5.1
```

```
from gtts import gTTS
from IPython.display import Audio, display
import os
from pprint import pprint
```

```
!pip install playsound
```

```
Collecting playsound
  Downloading playsound-1.3.0.tar.gz (7.7 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: playsound
  Building wheel for playsound (setup.py) ... done
  Created wheel for playsound: filename=playsound-1.3.0-py3-none-any.whl size=7020 sha256=f512a06ed1c05d2717b66ff7554800d83656b546ee9c09
  Stored in directory: /root/.cache/pip/wheels/90/89/ed/2d643f4226fc8c7c9156fc28abd8051e2d2c0de37ae51ac45c
Successfully built playsound
Installing collected packages: playsound
Successfully installed playsound-1.3.0
```

```
text="Computer engineering is the intersection of computer science and electrical engineering."
```

```
tts = gTTS(text)
```

```
tts.save("output.mp3")
tts.save("temp.mp3")
```

```
display(Audio("temp.mp3"))
transcribed_text = tts.text
```

```
def calculate_error_percentage(original, transcribed):
 original = original.lower()
 transcribed = transcribed.lower()
 error_count = sum(c1 != c2 for c1, c2 in zip(original, transcribed))
 total_chars = len(original)
 accuracy_percentage = ((total_chars - error_count) / total_chars) * 100 # Accuracy is (correct / total) * 100
 return accuracy_percentage
```

```
accuracy_percentage = calculate_error_percentage(text, transcribed_text)
print(f"Original Text: {text}")
print(f"Transcribed Text: {transcribed_text}")
print(f"Accuracy: {accuracy_percentage:.2f}%")
```

```
Original Text: Computer engineering is the intersection of computer science and electrical engineering.
Transcribed Text: Computer engineering is the intersection of computer science and electrical engineering.
Accuracy: 100.00%
```