LANGUAGE TRANSLATOR

- Source Code :

```
from tkinter import *  #version-tk 8.6
from tkinter import ttk
import googletrans    # version - 4.0.0
from googletrans import Translator, LANGUAGES
from gtts import gTTS   # version -2.3.2
import gtts.lang
import playsound  # version-0.2.13
import os

def translate():
        text = source_text.get(1.0, END).strip()
        src_lang = source_lang.get()
        dst_lang = dest_lang.get()
        translator = Translator()
        translation = translator.translate(text, src=src_lang,
        dest=dst_lang)
        translated_text.delete(1.0, END) translated_text.insert(END,
        translation.text)

def clear():
        source_text.delete(1.0, END)
        translated_text.delete(1.0, END)

# Create the main application window
root = Tk()
root.title("Language Translator")

# Create a frame for the content
  frame = Frame(root)
  frame.pack(pady=30)
```

```
code_language = gtts.lang.tts_langs() # Getting all the languages supported by gtts
del code_language['zh-TW']
del code_language['zh']

# Create source text label and textbox
source_text_label = Label(frame, text="Source Text", font=("Arial", 14))
source_text_label.grid(row=0, column=0, padx=10, pady=10, sticky="w") # Sticky='w' means stretch to west

source_text = Text(frame, font=("Arial", 12), height=5, width=40)
source_text.grid(row=1, column=0, padx=10, pady=5)

# Create source language label and drop-down menu
source_lang_label = Label(frame, text="Source Language", font=("Arial", 14))
source_lang_label.grid(row=2, column=0, padx=10, pady=10, sticky="w")
source_lang = ttk.Combobox(frame, values=list(code_language.values()), font=("Arial",12))
source_lang.grid(row=3, column=0, padx=10, pady=5)
source_lang.set("English")

# Create destination language label and drop-down menu
dest_lang_label = Label(frame, text="Destination Language", font=("Arial", 14))
dest_lang_label.grid(row=4, column=0, padx=10, pady=10, sticky="w")
dest_lang = ttk.Combobox(frame, values=list(code_language.values()), font=("Arial", 12))
dest_lang.grid(row=5, column=0, padx=10, pady=5)
dest_lang.set("Hindi") gtts.lang.tts_langs().values()
```

```
 def speaker():
      def get_language_codes():
                langs = gtts.lang.tts_langs()
                language_codes = {}
                for lang, name in langs.items():
                    language_codes[name.lower()] = lang # names of the
                    languages are converted to lowercase and used as key

      return language_codes

      text = translated_text.get(1.0, END)
      language_codes = get_language_codes()
      converted_voice = gTTS(text=text,
      lang=str(language_codes[dest_lang.get().lower()]))
      # lang will be the lowercase version of the selected language by
        user
      converted_voice.save("voice.mp3")
      playsound.playsound("voice.mp3")
      os.remove("voice.mp3")


# Create translate button
  translate_button = Button(frame, text="Translate", font=("Arial",
  14), command=translate)
  translate_button.grid(row=6, column=0, padx=10, pady=10)

# Create translated text label and textbox
  translated_text_label = Label(frame, text="Translated Text",
  font=("Arial", 14))
  translated_text_label.grid(row=7, column=0, padx=10, pady=10,
  sticky="w")
  translated_text = Text(frame, font=("Arial", 12), height=5,
  width=40)
  translated_text.grid(row=8, column=0, padx=10, pady=5)
```

```
# Create clear button
clear_button = Button(frame, text="CLEAR", command=clear)
clear_button.grid(row=9, column=0, padx=10, pady=10)

# Create speaker button
Speaker_button = Button(frame, text="SPEAKER",
command=speaker)
Speaker_button.grid(row=10, column=0, padx=10, pady=10)

# Start the application
root.mainloop()
```