A simple translation application using the Tkinter GUI library and the `deep_translator` module.

Tkinter: A standard Python library for building graphical user interfaces (GUIs).
deep_translator: A third-party library used for language translation via services like Google Translate.

```
from tkinter import *
from tkinter import ttk
from deep_translator import GoogleTranslator
```

`tkinter` is imported for GUI development.
`ttk` is part of `tkinter` and provides themed widgets (like `Combobox`).
`GoogleTranslator` from `deep_translator` is imported to handle translations.

```
def change(text="Type", src="english", dest="hindi"):
    translated = GoogleTranslator(source=src, target=dest).translate(text)
    return translated
```

This function translates the provided text from the source language (`src`) to the destination language (`dest`) using Google Translate. The default text is "Type," with "english" as the source and "hindi" as the target.

```
def data():
    s = combo_sor.get().lower()
    d = combo_dest.get().lower()
    msg = Sor_txt.get(1.0, END)
    textget = change(text=msg, src=s, dest=d)
    dest_txt.delete(1.0, END)
    dest_txt.insert(END, textget)
```

- **`combo_sor.get()`**: This method retrieves the current value selected in the `Combobox` widget `combo_sor` (the dropdown where the user selects the source language).
    - For example, if the user selects "English" from the dropdown, `combo_sor.get()` returns the string `"english"`.
- **`.lower()`**: This converts the retrieved string to lowercase to ensure the function works consistently, regardless of the case the user selects. For example, whether the user selects "English" or "english," it will always be returned as `"english"`.
  After this step, the variable `s` holds the source language (in lowercase), which will later be used for translation.

- Similar to the source language, this line retrieves the destination language selected by the user from the `combo_dest` dropdown.

- **`combo_dest.get()`**: Fetches the selected value from the destination language dropdown (e.g., `"hindi"` if the user selects Hindi).

- **`.lower()`**: Converts it to lowercase (e.g., "HINDI" would be converted to `"hindi"`).

  After this, `d` contains the target language for translation.

- **`Sor_txt.get(1.0, END)`: This method retrieves the text entered by the user in the "Source Text" text box (`Sor_txt`).**

  - **`1.0`: Refers to the starting position in the text box (the beginning of the text).**

  - **`END`: Represents the end of the text in the text box.**

  The `get()` method extracts all the text, starting from the first character (position `1.0`) to the last (position `END`). The result is stored in the variable `msg`, which is the text that will be translated.

  Note: The `Text` widget in Tkinter allows multiline input, so this method is suitable for extracting longer input than a single line of text.

  The `change()` function is called with the following arguments:

- **`text=msg`: The `msg` variable contains the source text that the user wants to translate.**
- **`src=s`: The source language (`s`) is passed as the source language for translation.**
- **`dest=d`: The destination language (`d`) is passed as the target language for translation.**

Inside the `change()` function:

- **The `GoogleTranslator` class from `deep_translator` is used to perform the translation. It takes the source and destination languages, as well as the text to be translated, and returns the translated text.**
- **The translated result is stored in the variable `textget`.**

`dest_txt.delete(1.0, END)`: Before inserting the translated text, the existing content in the destination text box (`dest_txt`) is cleared.

- **`1.0`: Indicates the starting position of the text.**
- **`END`: Refers to the end of the text (all the content).**

`dest_txt.insert(END, textget)`: The translated text stored in `textget` is inserted at the end of the destination text box (`dest_txt`), displaying the result of the translation.

```
root = Tk()

root.title("Translator")

root.geometry("500x700")

root.config(bg='red')
```

Creates a Tkinter window (`root`) with a title and specific dimensions (500x700).

Sets the background color of the window to red.

.