



UNIVERSITAS
MUHAMMADIYAH
CIREBON



APLIKASI KONVERSI SUHU DENGAN TKINTER: MEMBANGUN APLIKASI KONVERSI SUHU

DISUSUN OLEH :

IBNU ALWAN (220511014)

FAKULTAS TEKNIK
PRODI TEKNIK INFORMATIKA
UNIVERSITAS MUHAMMADIYAH CIREBON

2023

KATA PENGANTAR

Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang, segala puji hanya bagi-Nya. Shalawat dan salam senantiasa tercurah kepada Nabi Muhammad SAW, utusan Allah yang diutus sebagai rahmat bagi seluruh alam.

Selamat datang dalam aplikasi konversi suhu sederhana ini! Aplikasi ini dirancang menggunakan framework GUI (Graphical User Interface) tkinter pada bahasa pemrograman Python. Tujuan utama dari aplikasi ini adalah memberikan kemudahan dalam melakukan konversi suhu antar satuan seperti Celsius, Fahrenheit, dan Kelvin.

Aplikasi ini dirancang dengan antarmuka yang sederhana dan intuitif, memungkinkan pengguna untuk dengan mudah melakukan konversi suhu tanpa kesulitan. Dengan menggunakan elemen-elemen seperti “Entry” untuk memasukkan nilai suhu, “OptionMenu” untuk memilih satuan konversi, dan “Label” untuk menampilkan hasil konversi, aplikasi ini dapat dioperasikan dengan mudah bahkan oleh pengguna yang kurang berpengalaman sekalipun.

Cara penggunaan aplikasi ini sangatlah mudah. Pengguna hanya perlu memasukkan nilai suhu yang ingin dikonversi, memilih satuan konversi yang diinginkan, dan mengklik tombol "Konversi". Hasil konversi suhu akan ditampilkan dengan jelas di bawah tombol konversi, memberikan pengalaman yang ramah dan informatif

Adapun pilihan satuan konversi yang disediakan mencakup konversi antara Celsius, Fahrenheit, dan Kelvin. Dengan ini, pengguna dapat dengan bebas mengubah suhu dari satu satuan ke satuan lainnya sesuai kebutuhan.

Saya berharap aplikasi ini dapat memberikan kontribusi positif dalam mempermudah kegiatan sehari-hari yang melibatkan perubahan suhu. Selamat menggunakan aplikasi konversi suhu ini, dan jangan ragu untuk memberikan masukan atau saran guna meningkatkan kualitas aplikasi ini di masa mendatang.

Terima kasih atas kepercayaan Anda untuk menggunakan aplikasi konversi suhu ini. Semoga aplikasi ini dapat memenuhi kebutuhan Anda dengan baik.

Cirebon, 26 November 2023

Penulis

DAFTAR ISI

KATA PENGANTAR	ii
DAFTAR ISI	iii
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan Aplikasi	1
II. TAMPILAN APLIKASI	2
2.1 Judul Aplikasi	2
2.2 Input Pengguna	2
2.3 Unit Suhu	3
2.4 Hasil Konversi	3
2.5 Tombol Konversi	3
2.6 Menu Pilihan Unit Suhu	4
III. FUNGSIONALITAS APLIKASI	5
3.1 Inisialisasi Aplikasi	5
3.2 Pembuatan Widget	5
3.3 Pembaruan Deskripsi Input	6
3.4 Konversi Suhu	7
3.5 Penanganan Kesalahan	7
IV. IMPLEMENTASI TKINTER	8
4.1 Pembuatan Objek Tkinter	8
4.2 Pembuatan Objek Aplikasi	8
4.3 Menjalankan Aplikasi	9
V. PENUTUP	10
5.1 Kesimpulan	10
5.2 Source Code dan Output	10

I. PENDAHULUAN

1.1 Latar Belakang

Dalam kehidupan sehari-hari, suhu adalah parameter krusial yang memengaruhi berbagai aspek, termasuk cuaca, kesehatan, dan proses industri. Dengan adanya beragam satuan suhu seperti Celsius, Fahrenheit, dan Kelvin, seringkali diperlukan konversi suhu untuk memahami informasi secara komprehensif.

Ketidakseragaman satuan suhu dapat menjadi sumber kebingungan dan potensi kesalahan pengukuran. Keberagaman ini menciptakan kebutuhan untuk alat konversi suhu yang efisien, memungkinkan interpretasi yang lebih mudah dan aplikasi informasi terkait suhu.

Aplikasi Konversi Suhu menggunakan Tkinter bertujuan memberikan solusi praktis dan ramah pengguna terkait konversi suhu. Tkinter dipilih sebagai basis GUI karena keberlanjutan dan kemudahan penggunaannya, memungkinkan pengguna dari berbagai latar belakang untuk dengan mudah mengakses alat konversi suhu ini.

Dengan mengintegrasikan fleksibilitas konversi antara berbagai satuan suhu utama, aplikasi ini diharapkan dapat memberikan dukungan praktis untuk kebutuhan pengguna sehari-hari. Melalui pendekatan yang sederhana namun efektif, Aplikasi Konversi Suhu diharapkan dapat menjadi alat yang bermanfaat untuk meningkatkan pemahaman dan penerapan informasi suhu dalam berbagai konteks.

1.2 Tujuan Aplikasi

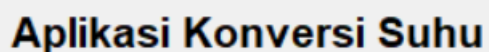
Tujuan utama aplikasi ini adalah memberikan antarmuka pengguna yang intuitif dan sederhana untuk melakukan konversi suhu. Dengan menyediakan pilihan unit suhu yang umum digunakan, pengguna dapat dengan mudah mengonversi suhu sesuai kebutuhan mereka.

II. TAMPILAN APLIKASI

2.1 Judul Aplikasi

Widget label judul aplikasi ('self.title_label') digunakan untuk menampilkan teks informatif "Aplikasi Konversi Suhu" dengan menggunakan font Arial berukuran 16 dalam gaya tebal. Label ini ditempatkan dalam objek master (jendela utama) dengan menggunakan metode 'grid'. Pengaturan kolom 'column' diset menjadi 0, sehingga label ini akan berada di kolom pertama, dan 'columnspan' diset menjadi 3, memungkinkan label ini mencakup tiga kolom sekaligus. Pengaturan ini memberikan efek visual pada label, membuatnya tampak lebih menonjol dan mencakup ruang yang lebih besar. Code:

```
1 # Judul aplikasi
2 self.title_label = tk.Label(self.master, text="Aplikasi Konversi Suhu", font=("Arial", 16, "bold"))
3 self.title_label.grid(row=0, column=0, columnspan=3, pady=10)
```



2.2 Input Pengguna

Widget entry ('self.input_entry') digunakan sebagai wadah untuk pengguna memasukkan nilai suhu yang akan dikonversi. Dengan panjang (width) sebanyak 15 karakter, entry ini memberikan ruang yang cukup untuk memasukkan nilai suhu dengan nyaman. Entry ditempatkan pada kolom pertama ('column=0') dan baris kedua ('row=1'), dengan memberikan padding pada sisi kiri dan kanan (padx=10) serta atas dan bawah ('pady=10'). Code :

```
1 # Entry untuk input pengguna
2 self.input_entry = tk.Entry(self.master, width=15)
3 self.input_entry.grid(row=1, column=0, padx=10, pady=10)
```



2.3 Unit Suhu

Widget label unit suhu ('Label') dibuat untuk menampilkan informasi tentang unit suhu yang akan dikonversi. Label ini terhubung dengan variabel 'self.current_unit'. Dengan menggunakan 'grid', label ditempatkan di baris 1, kolom 1. Parameter 'padx' dan 'pady' memberikan padding di sekitar label.

```
1 # Label unit suhu
2 self.label_unit = tk.Label(self.master, textvariable=self.current_unit)
3 self.label_unit.grid(row=1, column=1, padx=10, pady=10)
```

 Celsius

2.4 Hasil Konversi

Widget label hasil konversi ('Label') dibuat untuk menampilkan hasil dari konversi suhu. Dengan menggunakan 'grid', label ditempatkan di baris 2, kolom 0, dan merentang ke kolom 3. Parameter 'padx' dan 'pady' memberikan padding di sekitar label.

```
1 # Label hasil konversi
2 self.label_result = tk.Label(self.master, text="Hasil:")
3 self.label_result.grid(row=2, column=0, columnspan=3, padx=10, pady=10)
```

Hasil:

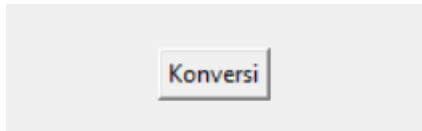
2.5 Tombol Konversi

Widget tombol konversi ('Button') dibuat untuk memicu proses konversi suhu. Dengan menggunakan 'grid', tombol ditempatkan di baris 3, kolom 0, dan merentang ke kolom 3. Parameter 'pady' memberikan padding di sekitar tombol. Perhatikan bahwa 'command=self.convert_temperature' menghubungkan tombol dengan metode 'convert_temperature' yang akan dijalankan saat tombol ditekan.

```

1 # Tombol konversi
2 self.convert_button = tk.Button(self.master, text="Konversi", command=self.convert_temperature)
3 self.convert_button.grid(row=3, column=0, columnspan=3, pady=10)
4

```



2.6 Menu Pilihan Unit Suhu

Widget menu dropdown ('OptionMenu') dibuat untuk memberikan opsi kepada pengguna untuk memilih unit suhu yang diinginkan. Objek 'StringVar' ('self.unit_menu') digunakan untuk menyimpan nilai default dan nilai terpilih dari menu. Opsi menu disediakan dalam bentuk daftar ('self.unit_menu_options'). Dengan menggunakan 'grid', menu dropdown ditempatkan di baris 4, kolom 0, dan merentang ke kolom 3. Parameter 'pady' memberikan padding di sekitar menu dropdown. 'command=self.update_input_description' menghubungkan menu dropdown dengan metode 'update_input_description' yang akan dijalankan saat pilihan dalam menu berubah.

```

1 # Menu pilihan untuk unit suhu
2 self.unit_menu = tk.StringVar(value="Pilih Pengkonversian")
3 self.unit_menu_options = ["Pilih Pengkonversian", "Celsius ke Fahrenheit", "Fahrenheit ke Celsius",
4                           "Celsius ke Kelvin", "Fahrenheit ke Kelvin", "Kelvin ke Celsius",
5                           "Kelvin ke Fahrenheit"]
6
7 self.unit_menu_dropdown = tk.OptionMenu(self.master, self.unit_menu, *self.unit_menu_options, command=self.update_input_description)
8 self.unit_menu_dropdown.grid(row=4, column=0, columnspan=3, pady=10)

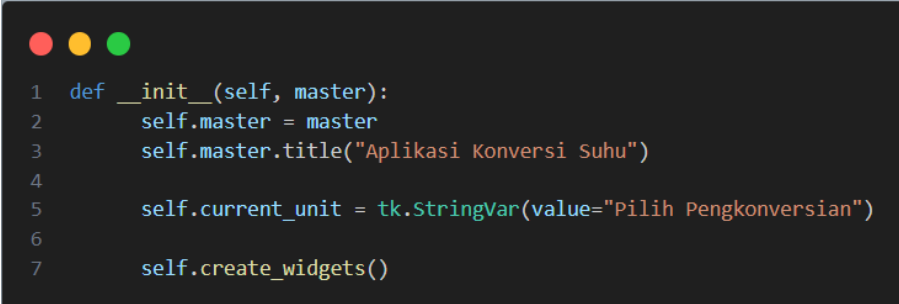
```



III. FUNGSIONALITAS APLIKASI

3.1 Inisialisasi Aplikasi

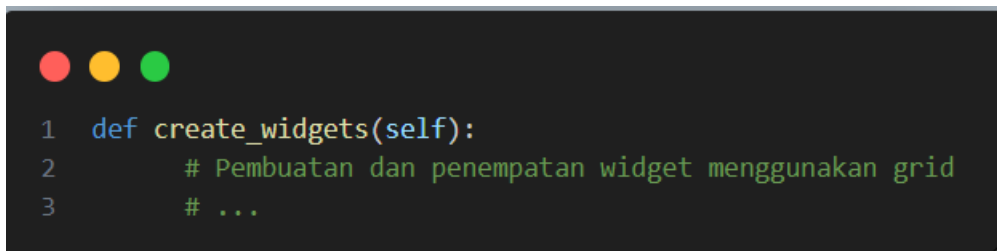
Dalam fase inisialisasi aplikasi, langkah-langkah utama termasuk:



```
1 def __init__(self, master):
2     self.master = master
3     self.master.title("Aplikasi Konversi Suhu")
4
5     self.current_unit = tk.StringVar(value="Pilih Pengkonversian")
6
7     self.create_widgets()
```

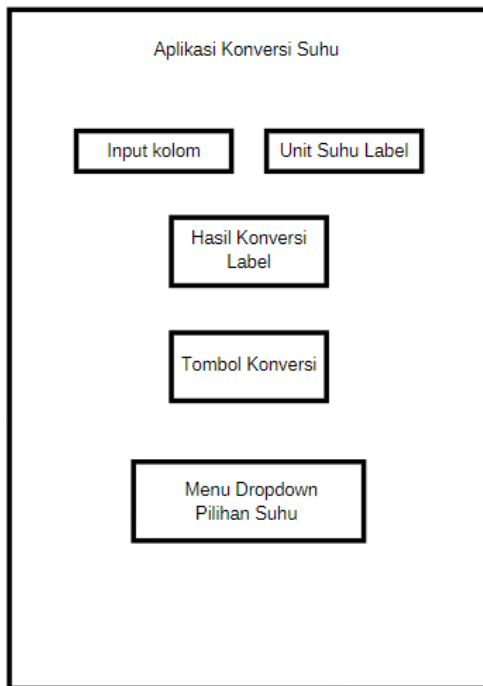
- `__init__`: Metode ini merupakan konstruktor untuk kelas `TemperatureConverterApp`.
- `self.master`: Variabel ini menyimpan objek master, yaitu jendela utama aplikasi Tkinter.
- `self.master.title`: Menetapkan judul jendela utama menjadi "Aplikasi Konversi Suhu".
- `self.current_unit`: Variabel `StringVar` yang digunakan untuk menyimpan dan memanipulasi nilai variabel untuk label unit suhu.
- `self.create_widgets()`: Metode ini dipanggil untuk membuat dan menempatkan widget dalam antarmuka pengguna.

3.2 Pembuatan Widget



```
1 def create_widgets(self):
2     # Pembuatan dan penempatan widget menggunakan grid
3     # ...
```

- `create_widgets`: Metode ini bertanggung jawab untuk membuat dan menempatkan elemen-elemen antarmuka pengguna seperti label, entry, tombol, dan menu dropdown.
- Widget-widget tersebut akan ditempatkan menggunakan metode `grid` untuk pengaturan dalam grid antarmuka pengguna.



Dalam diagram ini:

- Input Pengguna: Widget entry yang digunakan untuk memasukkan suhu yang akan dikonversi.
- Unit Suhu Label: Widget label yang menampilkan unit suhu yang akan dikonversi.
- Hasil Konversi Label: Widget label yang menampilkan hasil konversi suhu.
- Tombol Konversi: Widget tombol yang memicu proses konversi saat ditekan.
- Menu Dropdown Pilihan Suhu: Widget menu dropdown yang memberikan opsi unit suhu untuk konversi.

Diagram ini membantu memvisualisasikan posisi dan hubungan antara widget-widget dalam antarmuka pengguna aplikasi konversi suhu.

3.3 Pembaruan Deskripsi Input

```

1 def update_input_description(self, event=None):
2     selected_option = self.unit_menu.get()
3     if selected_option != "Pilih Pengkonversian":
4         input_description = selected_option.split(" ke ")[0]
5         self.current_unit.set(f"{input_description}")
6     else:
7         self.current_unit.set("Pilih Pengkonversian")
8

```

- `update_input_description`: Metode ini dipanggil ketika ada perubahan pada pilihan menu dropdown.
- Pilihan yang dipilih diambil dari `self.unit_menu`, dan label unit suhu diperbarui sesuai dengan pilihan tersebut.

3.4 Konversi Suhu

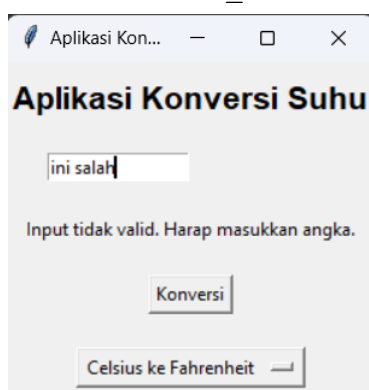
```
1 def convert_temperature(self):
2     try:
3         temperature = float(self.input_entry.get())
4         result = 0.0
5
6         current_option = self.unit_menu.get()
7
8         if current_option == "Pilih Pengkonversian":
9             self.label_result.config(text="Pilih pengkonversian terlebih dahulu.")
10            self.current_unit.set("")
11            return
12        # Logika konversi suhu
13        # ...
14
15        result_text = f"Hasil: {result:.2f} {unit_symbol}"
16        self.label_result.config(text=result_text)
17    except ValueError:
18        self.label_result.config(text="Input tidak valid. Harap masukkan angka.")
19        self.current_unit.set("")
20
```

- `convert_temperature`: Metode ini dipanggil saat tombol konversi ditekan.
- Input suhu dari pengguna diambil dari `self.input_entry` dan dicoba untuk dikonversi ke float. Jika berhasil, logika konversi suhu dijalankan, dan hasilnya ditampilkan pada label hasil konversi.
- Jika input tidak valid, pesan kesalahan ditampilkan pada label hasil konversi.

3.5 Penanganan Kesalahan

```
1 except ValueError:
2     self.label_result.config(text="Input tidak valid. Harap masukkan angka.")
3     self.current_unit.set("")
4
```

- Bagian dari metode `convert_temperature` yang menangani kesalahan jika input yang dimasukkan oleh pengguna tidak valid.
- Pesan kesalahan ditampilkan pada label hasil konversi, dan variabel `self.current_unit` diatur kembali ke nilai awal untuk label unit suhu.




IV. IMPLEMENTASI TKINTER

4.1 Pembuatan Objek Tkinter

```
1 if __name__ == "__main__":
2     root = tk.Tk()
3     app = TemperatureConverterApp(root)
4     root.mainloop()
5
```

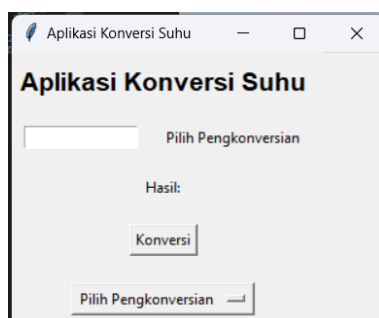
- ``__name__ == "__main__":` Ini memeriksa apakah skrip dijalankan sebagai skrip utama.
- ``tk.Tk()`:` Membuat objek utama Tkinter sebagai jendela root.
- ``TemperatureConverterApp(root)`:` Membuat objek aplikasi ``TemperatureConverterApp`` dengan root sebagai master.
- ``root.mainloop()`:` Menjalankan loop utama Tkinter untuk menampilkan antarmuka pengguna dan menangani interaksi pengguna.

 Aplikasi Konversi Suhu

4.2 Pembuatan Objek Aplikasi

```
1 class TemperatureConverterApp:
2     def __init__(self, master):
3         self.master = master
4         self.master.title("Aplikasi Konversi Suhu")
5
6         self.current_unit = tk.StringVar(value="Pilih Pengkonversian")
7
8         self.create_widgets()
```

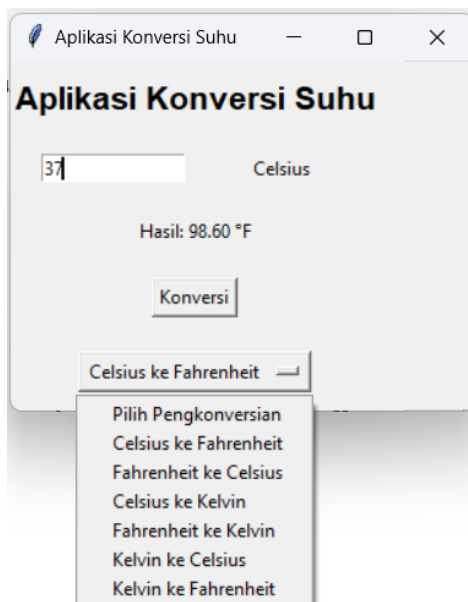
- ``TemperatureConverterApp`:` Kelas utama aplikasi konversi suhu.
- ``__init__`:` Metode konstruktor yang dipanggil saat objek aplikasi dibuat.
- ``self.master`:` Menyimpan objek master (root) untuk aplikasi Tkinter.
- ``self.current_unit`:` StringVar yang menyimpan dan memanipulasi nilai variabel untuk label unit suhu.
- ``self.create_widgets()`:` Metode untuk membuat dan menempatkan widget dalam antarmuka pengguna.



4.3 Menjalankan Aplikasi

```
1 if __name__ == "__main__":
2     root = tk.Tk()
3     app = TemperatureConverterApp(root)
4     root.mainloop()
5
```

- Blok ini berfungsi untuk menjalankan aplikasi jika skrip dijalankan sebagai skrip utama.
- `root = tk.Tk()`: Membuat objek root Tkinter.
- `app = TemperatureConverterApp(root)`: Membuat objek aplikasi dengan root sebagai master.
- `root.mainloop()`: Memulai loop utama Tkinter untuk menampilkan antarmuka pengguna dan menanggapi interaksi pengguna



V. PENUTUP

5.1 Kesimpulan

Melalui perjalanan ini, kita telah menjelajahi langkah-langkah rinci dalam pembuatan aplikasi konversi suhu menggunakan Tkinter. Dari perencanaan antarmuka pengguna hingga implementasi fungsionalitas konversi suhu, setiap tahap memunculkan pemahaman yang lebih mendalam tentang konsep pemrograman berbasis GUI. Kesimpulan yang dapat diambil dari proyek ini adalah sebagai berikut:

Pertama, Tkinter membuktikan dirinya sebagai alat yang tangguh dan sederhana dalam pembuatan antarmuka pengguna. Kelebihan integrasi dengan Python membuatnya menjadi pilihan yang kuat, terutama untuk proyek-proyek skala kecil hingga menengah.

Kedua, penggunaan desain antarmuka pengguna yang bersih dan intuitif memberikan pengalaman yang lebih baik bagi pengguna akhir. Dengan judul aplikasi yang jelas, entry untuk input yang terdefinisi dengan baik, dan label hasil konversi yang informatif, pengguna dapat dengan mudah berinteraksi dengan aplikasi.

Ketiga, pemahaman tentang konsep pengonversian suhu dari satu unit ke unit lainnya menjadi kunci. Implementasi fungsi konversi suhu memerlukan pemahaman yang kuat tentang formula matematika yang mendasari konversi antar satuan suhu.

5.2 Source Code dan Output

```
import tkinter as tk

class TemperatureConverterApp:
    def __init__(self, master):
        self.master = master
        self.master.title("Aplikasi Konversi Suhu")

        self.current_unit = tk.StringVar(value="Pilih Pengkonversian")

        self.create_widgets()

    def create_widgets(self):
        # Judul aplikasi
        self.title_label = tk.Label(self.master, text="Aplikasi Konversi Suhu", font=("Arial", 16, "bold"))
        self.title_label.grid(row=0, column=0, columnspan=3, pady=10)

        # Entry untuk input pengguna
        self.input_entry = tk.Entry(self.master, width=15)
        self.input_entry.grid(row=1, column=0, padx=10, pady=10)

        # Label unit suhu
        self.label_unit = tk.Label(self.master, textvariable=self.current_unit)
```

```

self.label_unit.grid(row=1, column=1, padx=10, pady=10)

# Label hasil konversi
self.label_result = tk.Label(self.master, text="Hasil:")
self.label_result.grid(row=2, column=0, columnspan=3, padx=10,
pady=10)

# Tombol konversi
self.convert_button = tk.Button(self.master, text="Konversi",
command=self.convert_temperature)
self.convert_button.grid(row=3, column=0, columnspan=3, pady=10)

# Menu pilihan untuk unit suhu
self.unit_menu = tk.StringVar(value="Pilih Pengkonversian")
self.unit_menu_options = ["Pilih Pengkonversian", "Celsius ke
Fahrenheit", "Fahrenheit ke Celsius",
"Celsius ke Kelvin", "Fahrenheit ke Kelvin",
"Kelvin ke Celsius",
"Kelvin ke Fahrenheit"]

self.unit_menu_dropdown = tk.OptionMenu(self.master, self.unit_menu,
*self.unit_menu_options, command=self.update_input_description)
self.unit_menu_dropdown.grid(row=4, column=0, columnspan=3, pady=10)

def update_input_description(self, event=None):
    selected_option = self.unit_menu.get()
    if selected_option != "Pilih Pengkonversian":
        input_description = selected_option.split(" ke ")[0]
        self.current_unit.set(f"{input_description}")
    else:
        self.current_unit.set("Pilih Pengkonversian")

def convert_temperature(self):
    try:
        temperature = float(self.input_entry.get())
        result = 0.0

        current_option = self.unit_menu.get()

        if current_option == "Pilih Pengkonversian":
            self.label_result.config(text="Pilih pengkonversian terlebih
dahulu.")
            self.current_unit.set("")
            return

        if "Celsius ke Fahrenheit" in current_option:
            result = (temperature * 9/5) + 32
            unit_symbol = "°F"

```

```

elif "Fahrenheit ke Celsius" in current_option:
    result = (temperature - 32) * 5/9
    unit_symbol = "°C"
elif "Celsius ke Kelvin" in current_option:
    result = temperature + 273.15
    unit_symbol = "K"
elif "Fahrenheit ke Kelvin" in current_option:
    result = (temperature - 32) * 5/9 + 273.15
    unit_symbol = "K"
elif "Kelvin ke Celsius" in current_option:
    result = temperature - 273.15
    unit_symbol = "°C"
elif "Kelvin ke Fahrenheit" in current_option:
    result = (temperature - 273.15) * 9/5 + 32
    unit_symbol = "°F"

result_text = f"Hasil: {result:.2f} {unit_symbol}"
self.label_result.config(text=result_text)
except ValueError:
    self.label_result.config(text="Input tidak valid. Harap masukkan angka.")
    self.current_unit.set("")

if __name__ == "__main__":
    root = tk.Tk()
    app = TemperatureConverterApp(root)
    root.mainloop()

```

