**Dokumentáció: Python Tkinter Átváltó Program**

***Készítette: Bankó Olivér és Hoffer Tamás***

**Áttekintés**

Ez a program egy grafikus felhasználói felületet (GUI) biztosít a mértékegységek közötti átváltás elvégzésére a Python tkinter könyvtárának segítségével. A felhasználó választhat az elérhető mértékegységek közül (hossz, térfogat, tömeg), megadhat egy értéket, majd az alkalmazás kiszámolja az átváltott értéket és megjeleníti.

**Használat**

1. **Indítás:** A program elindítása után a felhasználó az első legördülő menüből választhatja ki az átváltás típusát (hossz, térfogat, tömeg).
2. **Érték megadása:** A felhasználó megadja a kívánt értéket az "Bemenet" mezőben.
3. **Átváltás:** A "Átszámítás" gombra kattintva a program elvégzi az átváltást és megjeleníti az eredményt a "Kimenet" mezőben.

**Program felépítése**

**1. Tkinter és szükséges könyvtárak importálása**

A tkinter könyvtár biztosítja a GUI elemeket, mint például a gombok, szövegmezők és legördülő menük.

**2. Átváltási logika**

A program három típusú átváltást kezel:

* **Hossz**
* **Térfogat**
* **Tömeg**

Mindegyik típushoz tartozik egy dictionary, amely az alapértelmezett mértékegységekhez tartozó átváltási tényezőket tartalmazza.

**3. A convert\_units() függvény**

A convert\_units() függvény felelős az átváltás végrehajtásáért. A program először ellenőrzi, hogy a bemenet egy szám-e, majd a kiválasztott átváltási típustól függően végzi el a számítást.

A program főablakát a root változó tartalmazza. A mértékegységek kiválasztásához szükséges legördülő menüket és egyéb elemeket a Frame tárolja, és azokat a grid elrendezéssel pozicionáljuk.

A GUI tartalmazza:

* **Legördülő menü:** A felhasználó választhatja ki az átváltás típusát (Hossz, Térfogat, Tömeg).
* **Bemeneti mezők:** Az első mezőbe írhatjuk be az átváltani kívánt értéket, a második mezőbe pedig az eredményt jeleníti meg a program.
* **Gomb:** Az "Átszámítás" gomb a kiválasztott átváltási típus és mértékegységek alapján kiszámolja az eredményt.

**5. A update\_converter() függvény**

A update\_converter() függvény frissíti az átváltási opciókat a felhasználó választása alapján. Ha a felhasználó egy adott típusú átváltást választ (például hosszúságot), akkor az azt követő mértékegységek jelennek meg.

**Program futtatása**

A program végén a root.mainloop() hívás indítja el az alkalmazást. Ez folyamatosan figyeli az eseményeket és reagál a felhasználói interakciókra.

**6. Kód**

#tkinter importálása a programba

import tkinter as tk

from tkinter import \*

# Számolás gomb funkciói

def convert\_units():

    try:

        value = float(bemenet.get())

        from\_unit = clicked.get()

        to\_unit = clicked2.get()

        # Átváltási számolások

        length\_factors = {

            "Milliméter": 0.001,

            "Centiméter": 0.01,

            "Méter": 1,

            "Kilométer": 1000

        }

        volume\_factors = {

            "Milliliter": 0.001,

            "Centiliter": 0.01,

            "Liter": 1,

            "Köbméter": 1000

        }

        weight\_factors = {

            "Gramm": 0.001,

            "Dekagramm": 0.01,

            "Kilogramm": 1,

            "Tonna": 1000

        }

        # Mennyiség váltó típusának kiválasztása

        if selected\_converter.get() == "Hossz":

            factors = length\_factors

        elif selected\_converter.get() == "Térfogat":

            factors = volume\_factors

        elif selected\_converter.get() == "Tömeg":

            factors = weight\_factors

        else:

            raise ValueError("Nem támogatott átváltó típus!")

        # Műveletek

        from\_factor = factors[from\_unit]

        to\_factor = factors[to\_unit]

        result = value \* (from\_factor / to\_factor)

    except ValueError:

        print("Adjon meg egy számot!")

        return

#végösszeg kerekítése, nullázása, "readonly" funkció

    result = round(result, 8)

    bevitel.config(state="normal")

    bevitel.delete(0, END)

    bevitel.insert(0, result)

    bevitel.config(state="readonly")

# Frame a különböző átváltókhoz

def update\_converter(selected\_converter):

    for widget in converter\_frame.winfo\_children():

        widget.grid\_forget()

#bemeneti mezők pozícionálása keretrendszerben

    bemenet.grid(column=0, row=0, padx=10, pady=5)

    bevitel.grid(column=0, row=1, padx=10, pady=5)

    drop.grid(column=1, row=0, padx=10, pady=5)

    drop2.grid(column=1, row=1, padx=10, pady=5)

    calculate\_button.grid(column=1, row=2, pady=10)

    if selected\_converter == "Hossz":

        options = length\_options

    elif selected\_converter == "Térfogat":

        options = volume\_options

    elif selected\_converter == "Tömeg":

        options = weight\_options

    else:

        return

    drop["menu"].delete(0, "end")

    drop2["menu"].delete(0, "end")

    for option in options:

        drop["menu"].add\_command(label=option, command=tk.\_setit(clicked, option))

        drop2["menu"].add\_command(label=option, command=tk.\_setit(clicked2, option))

    clicked.set(options[0])

    clicked2.set(options[0])

# Konzolalkalmazás alapjai

root = tk.Tk()

root.minsize(width=500, height=250)

root.title("Átváltó Program")

root.config(padx=40, pady=50, background="white")

# Átváltó lehetőségek

options = ["Válasszon egy átváltót", "Hossz", "Térfogat", "Tömeg"]

selected\_converter = StringVar()

selected\_converter.set(options[0])

#főoldal alapbeállítása, pozícionálása

main\_menu = OptionMenu(root, selected\_converter, \*options, command=update\_converter)

main\_menu.grid(row=0, column=0, padx=10, pady=10)

converter\_frame = Frame(root)

converter\_frame.grid(row=1, column=0, padx=10, pady=10)

bemenet = tk.Entry(converter\_frame)

bevitel = tk.Entry(converter\_frame)

bevitel.config(state="readonly")

# Mérték lehetőségek

length\_options = ["Milliméter", "Centiméter", "Méter", "Kilométer"]

volume\_options = ["Milliliter", "Centiliter", "Liter", "Köbméter"]

weight\_options = ["Gramm", "Dekagramm", "Kilogramm", "Tonna"]

clicked = StringVar()

clicked.set(length\_options[0])  # alapértelmezett mértékegység a legördülő menüben

drop = OptionMenu(converter\_frame, clicked, \*length\_options)

clicked2 = StringVar()

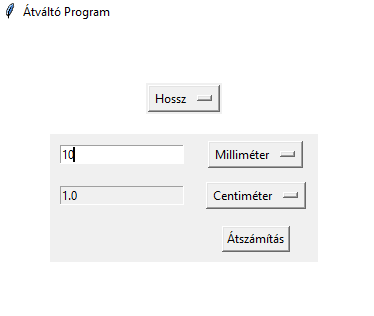
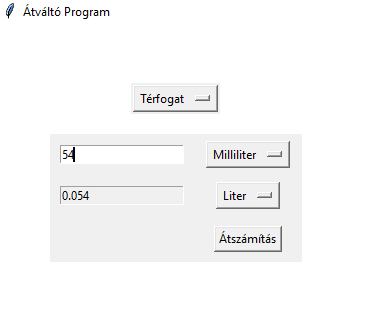
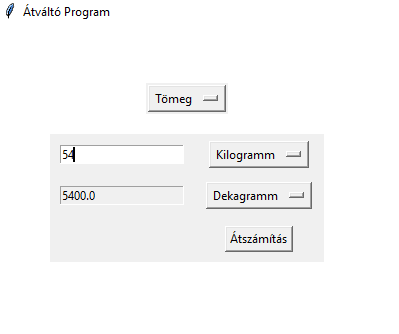
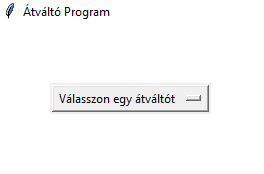
clicked2.set(length\_options[0])

drop2 = OptionMenu(converter\_frame, clicked2, \*length\_options)

calculate\_button = Button(converter\_frame, text="Átszámítás", command=convert\_units)

root.mainloop()

**7. Illusztráció**

**8. Továbbfejlesztési lehetőségek:**

* Többfajta érték és mértékegység
* Előzmények fül
* GUI szebb felülete