

Ardian Ferdy Firmansyah

Konversi Mata Uang Gui

Code

```
import requests
from tkinter import *
import tkinter as tk
from tkinter import ttk
import re

class RealTimeCurrencyConverter():
    def __init__(self,url):
        self.data = requests.get(url).json()
        self.currencies = self.data['rates']

    def convert(self, from_currency, to_currency, amount):
        initial_amount = amount
        if from_currency != 'USD' :
            amount = amount / self.currencies[from_currency]
        amount = round(amount * self.currencies[to_currency], 4)
        return amount

class App(tk.Tk):

    def __init__(self, converter):
        tk.Tk.__init__(self)
        self.title = 'Konversi Mata Uang'
        self.currency_converter = converter
        self.geometry("500x300")

        self.intro_label = Label(self, text = 'Konversi Mata Uang', anchor='n'
, fg = 'blue', relief = tk.RAISED, borderwidth = 3)
        self.intro_label.config(font = ('Courier',15,'bold'))
        self.date_label = Label(self, text = f"1 Dolar Amerika Sama Dengan = {
self.currency_converter.convert('USD','IDR',1)} Rupiah\n1 Euro Sama Dengan = {
self.currency_converter.convert('EUR','IDR',1)} Rupiah  \n Date : {self.curren
cy_converter.data['date']}", relief = tk.GROOVE, borderwidth = 5)
        self.intro_label.place(x = 10 , y = 5)
        self.date_label.place(x = 160, y= 50)
        valid = (self.register(self.restrictNumberOnly), '%d', '%P')
        self.amount_field = Entry(self,bd = 3, relief = tk.RIDGE, justify = tk
.CENTER,validate='key', validatecommand=valid)
        self.converted_amount_field_label = Label(self, text = '', fg = 'black
', bg = 'white', relief = tk.RIDGE, justify = tk.CENTER, width = 17, borderwid
th = 3)

        self.from_currency_variable = StringVar(self)
```

```

        self.from_currency_variable.set("IDR")
        self.to_currency_variable = StringVar(self)
        self.to_currency_variable.set("USD")

        font = ("Courier", 12, "bold")
        self.option_add('*TCombobox*Listbox.font', font)
        self.from_currency_dropdown = ttk.Combobox(self, textvariable=self.from_currency_variable, values=list(self.currency_converter.currencies.keys()), font = font, state = 'readonly', width = 12, justify = tk.CENTER)
        self.to_currency_dropdown = ttk.Combobox(self, textvariable=self.to_currency_variable, values=list(self.currency_converter.currencies.keys()), font = font, state = 'readonly', width = 12, justify = tk.CENTER)

        self.from_currency_dropdown.place(x = 30, y= 120)
        self.amount_field.place(x = 36, y = 150)
        self.to_currency_dropdown.place(x = 340, y= 120)
        #self.converted_amount_field.place(x = 346, y = 150)
        self.converted_amount_field_label.place(x = 346, y = 150)

        self.convert_button = Button(self, text = "Konversikan", fg = "black", command = self.perform)
        self.convert_button.config(font=('Courier', 10, 'bold'))
        self.convert_button.place(x = 225, y = 135)

    def perform(self):
        amount = float(self.amount_field.get())
        from_curr = self.from_currency_variable.get()
        to_curr = self.to_currency_variable.get()
        converted_amount = self.currency_converter.convert(from_curr,to_curr,amount)

        converted_amount = round(converted_amount, 2)
        self.converted_amount_field_label.config(text = str(converted_amount))

    def restrictNumberOnly(self, action, string):
        regex = re.compile(r"[0-9,]*?(\.)?[0-9,]*$")
        result = regex.match(string)
        return (string == "" or (string.count('.') <= 1 and result is not None))

if __name__ == '__main__':
    url = 'https://api.exchangerate-api.com/v4/latest/USD'
    converter = RealTimeCurrencyConverter(url)

    App(converter)
    mainloop()

```

Output

