

Assessment Submission Form

Student Number (If this is group work, please include the student numbers of all group participants)	G/H1024/169
Assessment Title	Trinity Final Assessment
Module Code	M604
Module Title	Advanced Programming
Module Tutor	William Morrison
Date Submitted	21 st September, 2023 21/09/2023

Declaration of Authorship

I declare that all material in this assessment is my own work except where there is clear acknowledgement and appropriate reference to the work of others.

I fully understand that the unacknowledged inclusion of another person's writings or ideas or works in this work may be considered plagiarism and that, should a formal investigation process confirms the allegation, I would be subject to the penalties associated with plagiarism, as per GISMA Business School, University of Applied Sciences' regulations for academic misconduct.

Signed Awaiz Date 21/09/2023

GISMA UNIVERSITY OF APPLIED SCIENCES

POTSDAM, GERMANY

Instructor Name : William Baker Morrison

Course Title : Advanced Programming (M604)

Subject : Use Of Tkinter GUI

Submission Date : 21st September , 2023

Submitted by : AWAIS RAFI

Student Id : GH1024169

Introduction:

SolarBuddy is a CivTech firm based in United States Of America that specializes in green energy solutions primarily focusing on power generation harnessing energy from the sun using photovoltaic plates installed in homes and industries with the government shifting a certain percentage of domestic consumption on renewables such as solar.

What is Solar Power ?

The sun is a natural lamp for planet earth that formed billions of years ago after the big bang . The sun provides critical energy in the form of photons , plants absorb them to grow , humans need them to maintain a healthy body , earth needs them to maintain a healthy livable ecosystem . Photovoltaic plates absorb photons and convert them into electrical power , depending on the design and material used , either monocrystalline or polycrystalline , single crystal for mono and more than one crystal for poly . The power generated varies with design. The world record efficiency is 47.1 (%) held by National Renewable Energy Laboratory (NREL).

Solar Calculations:

To be able to determine the number of solar plates needed , a rough estimate of the power consumption needs to be evaluated . The Unit of Power is Watts , named after a Illustrious Scottish Inventor James Watt (1736-1819).

The power consumption is evaluated as the number of light bulbs/LEDS , Television , refrigerator or any other gadget consuming power , however in real practical terms the number of units of electricity consumed monthly is sufficient to get a rough estimate , such data is readily available by the local power supplier or even disclosed by consent by the homeowner .

The power generated by a solar plate can be calculated as follows:

Legend:

S.P.C= Solar Plate Capacity H.Sunlight = Hours of Sunlight S.P = Solar Power Generated
 $S.P.C * H.Sunlight = S.P \text{ Generated (Kilo Watt Hour)}$ $250 (KW) * 3(H) = 750 (KWH)$

By rough estimate a solar plate generates 750 (KWH) of power using a 250 (KW) solar plate with 3 Hours of Sunlight.

Considering the above calculations an entire city can be modelled, instead of just 01 Solar plate, the calculations can be carried out for several hundred plates as well.

Objective:

The objective of the assignment is to use graphical user interface of python programming language mainly tkinter.

A CIVTECH dataset with power generated by various power sources in USA will be used to display the results in a interactive manner for the user.

What is tkinter ?

Python is object oriented programming language . However, in practice a programming language needs to have a GUI (Graphical User Interface) so that the software coded can be visually appealing for users and can be easily accessed and understood by almost anyone without any technical knowledge.

Tkinter Code:

Step – 01:

The Initial Step is to Import all the necessary Libraries needed to code the tkinter app.

```
import tkinter as tk
import matplotlib.pyplot as plt
import pandas as pd
from matplotlib.backends.backend_tkagg import (FigureCanvasTkAgg,
NavigationToolbar2Tk)
from PIL import Image, ImageTk
from tkinter import Text
```

Step – 02:

Pandas Library is used to Import the DataSet

```
df = pd.read_csv(
    r"C:\Users\USER\Desktop\Renewable Energy\renewable electricity by country
- US states renewable electricity.csv")
pd.set_option('display.max_columns', None)
print(df)
```

Step – 03:

The tkinter module is imported into the console using tk.Tk() function

```
window = tk.Tk()
window.configure()
window.title("Solar Buddy")
window.geometry("800x800")
```

```
fig, ax = plt.subplots()
plt.grid()
```

Step – 04:

An image is uploaded onto the tkinter app for visually pleasing aesthetics for the user interacting with the tkinter app.

```
solar_image = Image.open(r"C:\Users\USER\Desktop\Renewable Energy\Solar
Panel.webp")
obj = ImageTk.PhotoImage(solar_image)
```

Step-05:

The image is placed onto the tkinter frame using pack().

```
lbl_1 = tk.Label(image=obj)
lbl_1.image=obj

lbl_1.pack()

frame = tk.Frame(window)
label = tk.Label(text="Solar Power Generated")
label.config(font=("Arial"))
label.pack()
frame.pack()

canvas = FigureCanvasTkAgg(fig, master=window)
canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=1)
```

Step-06:

The dataset is extracted and entries are displayed onto the Frame. The try and exception technique is used to check exception handling for the app as seen below.

```
def plotting():
    x = df['US STATE'].head(10)
    y = df['renewables'].head(10)
    plt.figure(figsize=(12, 12))
    plt.figure(figsize=(12, 12))
    ax.plot(x, y)
    canvas.draw()

#Exception Handling Test
try:
    print(dff)
except:
    print("dff is not defined")

#####

def solar_power_calculation(solar_plate,solar_power):
    return solar_plate * solar_power

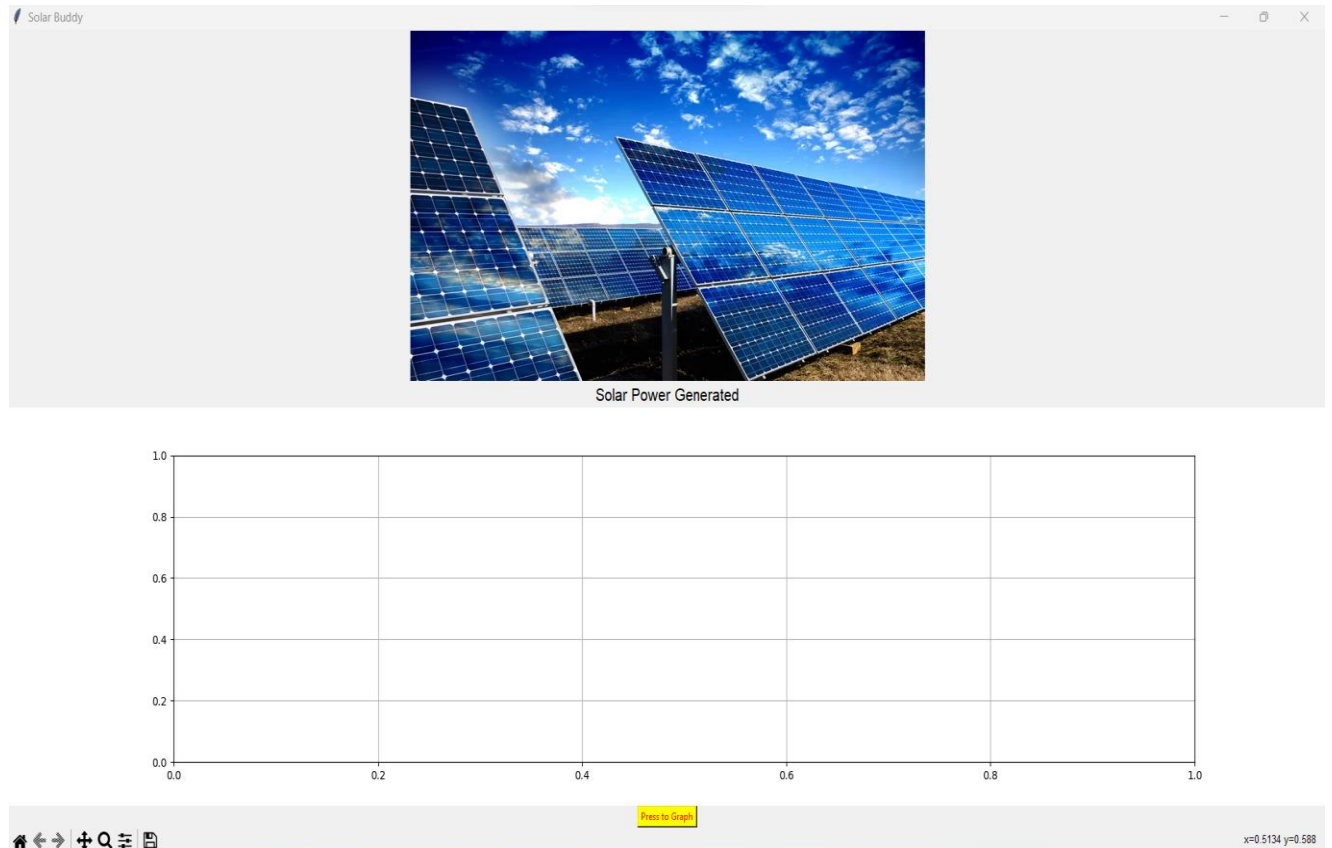
#####
###
b = tk.Button(window, text="Press to Graph",
command=plotting,fg='red',bg='yellow')
b.pack()

toolbar = NavigationToolbar2Tk(canvas, window, pack_toolbar=False)
toolbar.update()
toolbar.pack(side=tk.BOTTOM, fill=tk.X)

window.mainloop()
```

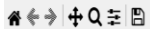
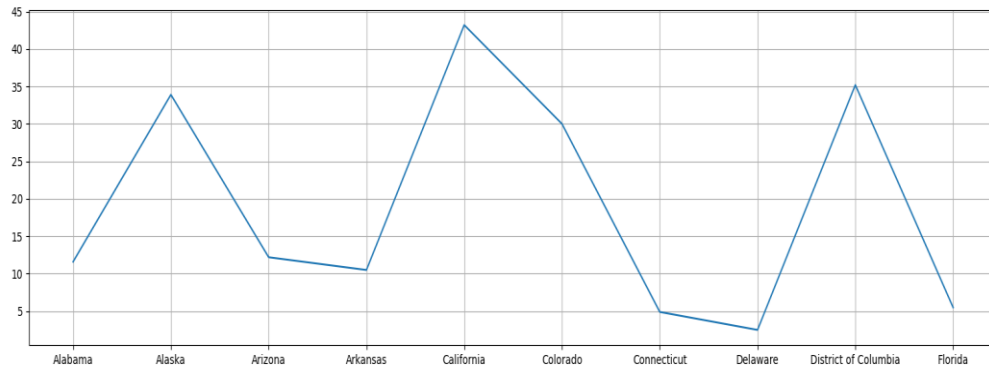
Tkinter App :

Following is the Snippet of the tkinter App , as it can be noticed a Image has been placed at the top . A Empty Grid is placed right below with a PressToGraph Button at the bottom to Initiate Graph plotting.



Tkinter Plot:

Press to graph plots the entire graph on the grid .



Press to Graph

Github Code Link :

<https://github.com/rafiaw01/M604-Advanced-Programming/upload/main>