**Purpose of the Script:**

The python script calculates the DC power output of a solar panel system installed at ASU’s Photovoltaic Reliability Lab (PRL), it shows the estimate of one typical year (2024) and here the output is in the form of graph.

**Libraries and Modules Used:**

**1. Pandas:** Used for storing the weather data in tables to extract columns ‘ghi’ and ‘temp\_air’ and for the time indexed data.

**2. matplotlib and matplotlib.pyplot:** For graphical representation such as to create a canvas, to draw the power output and to save the figure as an image file.

**matplotlib.use('Agg')** – Used for non-GUI backend to avoid tkinter error

**3. pvlib.pvsystem.pvwatts\_dc:** For calculating power output from irradiance and temperature using ‘pvwatts\_dc’ function

**4. pvlib.location:** To create a location object to represent the physical location of ASU PRL

**5. pvlib.iotools:** Used to fetch solar irradiance and weather data from sources TMY2/3

**Script Breakdown:**

**1.** Importing all the modules

**A screen shot of a computer code

AI-generated content may be incorrect.**

**2.** Defined the geographical coordinates (latitude & longitude) of ASU PRL for fetching location-specific weather data.

**A number on a black background

AI-generated content may be incorrect.**

**3.** Created a location object.

****

**4.** For fetching Typical Meteorological Year (TMY) data for the given coordinates used only two items tmy\_data (the actual weather) and metadata.

****

**5.** Print the first 5 rows of the tmy\_data DataFrame in the console.

****

**6.** Since TMY data doesn't come labelled as "2024", we manually set the time index to match each hour in 2024 (from Jan 1 to Dec 31).

****

**7.** ghi = Global Horizontal Irradiance in W/m² → how much sunlight hits a flat surface and temp\_air = ambient air temperature in Celsius → affects panel performance.

**A black background with white text

AI-generated content may be incorrect.**

**8.** Inspects the first few values of the two most important variables used for calculating solar panel power output.

****

**9.** pdc0 = DC system size in watts → tells us the panel capacity under ideal conditions and gamma\_pdc = temperature loss factor: the system loses 0.3% output per °C above 25°C.

**A black background with white text

AI-generated content may be incorrect.**

**10.** Used the PVWatts model for estimating DC power output using the Irradiance, Temperature, system size and loss factor.

****

**11.** Prints the first 5 values of the calculated DC power output.

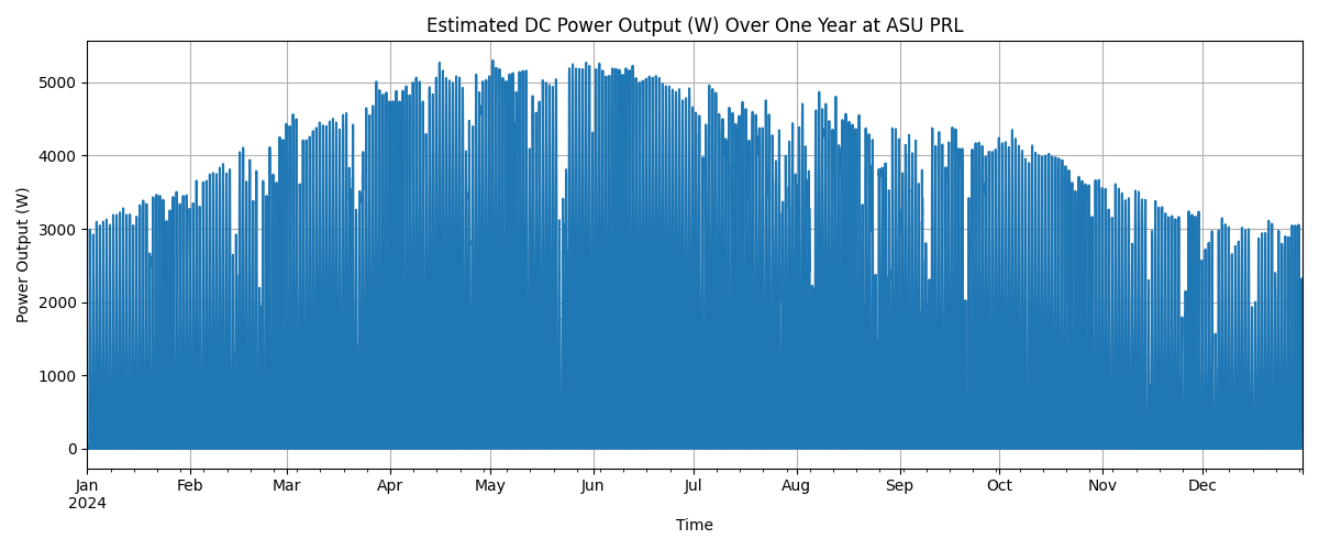
****

**12.** It creates a new blank figure figsize=(12, 5) sets the width = 12 inches and height = 5 inches. It uses Pandas .plot() function which is used to plot the dc\_power series where X-axis: time (DatetimeIndex from tmy\_data) and Y-axis: power output (in watts). ‘plt.xlabel(‘Time’)’ it adds a label to the X-axis. ‘plt.ylabel('Power Output (W)')’ adds a label to the Y-axis. ‘plt.grid(True)’ adds a grid to the plot background. ‘plt.tight\_layout()’ to adjust the spacing between the elements.

**A computer screen with text on it

AI-generated content may be incorrect.**

**Graphical Output:**

****

The Graph shows the DC power output (W) estimation over one year at ASU PRL starting from Jan 2024 till December 2024.

**X-Axis:**

It represents the entire year of 2024 where each tick marks a month and this is based on hourly TMY data, so the chart contains 8760 data points.

**Y-Axis:**

It shows the DC power output of the solar panels for each our, where the values range from 0W to just above 5000W which is basically the system capacity pdc0 = 5000 watts.

The graph shows the hike in power output from January to June, then it decreases from July to December. This is because of longer sunnier days in spring/summer and shorter, weaker sun in winter months.

Here the shape is influenced by the Irradiance (ghi), temperature (temp\_air), time of day, season and clouds/haze.