**Date: 28-10-2025**

**Name: MACHAVARAM SATWIK**

**MTech – 1st Year**

**HTNO: 2505B04301**

**Application for TGNPDCL  
Objective:** Build an application using Python program for TGNPDCL, to generate the  
bill based on energy consumption and type of customer with the help of AI tools.  
Build a python application as per below instructions  
• Read all the required data like PU,CU and Type of customer  
• Calculate bill amount based on number of units consumed, type of customer  
and other charges  
• Finally print the values of EC(Energy Charges),FC(Fixed  
Charges),CC(Customer Charges),ED(Electricity Duty Charges),bill as per  
expected output  
**Requirements:**  
• VS Code with Github Copilot and/or Google Colab with Gemini  
• Students should bring power bill from their home  
**Deliverables**:  
• Print energy bill for given inputs and verify with bill taken by students

**Solution for the above question:**

**Contract (inputs / outputs / assumptions)**

* **Inputs:**
  + CU (units consumed) — numeric
  + PU (price per unit) — numeric (Rs/unit) OR you may use slab-based rates (optional)
  + customer type — one of: domestic, commercial, industrial, agriculture
  + meter rent and other charges — optional (defaults 0)
  + ED (electricity duty rate) — optional (default 5% = 0.05)
* **Outputs:**
  + EC: Energy Charges
  + FC: Fixed Charges (per customer type)
  + CC: Customer Charges (per customer type)
  + ED: Electricity Duty applied (on EC in this simplified model)
  + Other: meter rent + other sundry charges
  + Total: sum of all components
* **Assumptions:**
  + ED applied on EC only (replace logic if ED base is different).
  + A simple fixed-charge table is included. Replace with official TGNPDCL slabs/values for production.
  + By default the calculation uses a single PU \* CU; slab tariffs are optional (commented example included).

**Steps to progress the TGNPDCL:**

1. Run the current script (non-interactive example)

Expected: an example bill printed (EC, FC, CC, ED, Other, Total).

1. Enter values copied from your physical bill: CU (units), PU (Rs/unit), meter rent, other charges (if any), ED percent (if the bill shows it).
2. How to map fields on a physical TGNPDCL bill to the program inputs

* Units consumed (CU): Look for "Energy consumed" or "Units" — use that as [units](vscode-file://vscode-app/c:/Users/Chetti%20Rishika/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html).
* Price per unit (PU): If the bill lists a single per-unit rate, use that. If the bill uses slabs, you'll need slab logic (see step 6).
* Fixed Charges (FC): The bill may show "Fixed charges" or "Minimum charges" — that maps to FC.
* Customer Charges (CC): Could be labeled "Customer charges" or "Service charges".
* Electricity Duty (ED): Often shown as a percent or a separate line "Electricity Duty"; note whether it's applied to EC only or EC+others.
* Meter rent / Other charges: Use the exact line items labeled "Meter rent", "Misc charges", etc.
* Grand total: Compare program's Total vs bill's Total.

1. If numbers don't match exactly — common reasons and  fixes
2. Slab tariffs: Many domestic bills use slabbed per-unit rates. If the bill uses slabs, implement slab logic (see step 6).
3. Rounding and decimal places: Billing systems often round per-slab or per-line items. Try rounding intermediate steps similar to bill format.
4. ED base difference: Some utilities apply ED on (EC + FC) or on the net bill; check the bill's calculation lines and adapt ed calculation accordingly.
5. Taxes, surcharges or subsidy adjustments: There may be additional lines not yet in the script — add them as [other\_charges](vscode-file://vscode-app/c:/Users/Chetti%20Rishika/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) or explicit components.
6. Quick validation test (manual example):
7. Example inputs: Units=250, PU=5.50, Type=Domestic, MeterRent=20, Other=15, ED=5%
8. You should get:
9. EC = 250 \* 5.50 = 1375.00
10. FC = 50.00 (domestic default)
11. CC = 30.00
12. ED = 1375 \* 0.05 = 68.75
13. Other = 35.00
14. Total = 1558.75
15. If your physical bill shows different values, locate which line differs and check slab/ED base.
16. Implementing slab tariffs (if TGNPDCL uses slabs)
17. Replace the simple EC = units \* pu with slab calculation. Example pattern:
18. Define slabs as a list of (limit, rate) sorted ascending: [(100, 3.0), (200, 4.5), (None, 6.0)]
19. Compute EC by consuming units across slabs: first 100 at 3.0, next 100 at 4.5, remaining at 6.0.
20. I can add slab support for domestic and commercial if you provide the slab table from a sample bill or request a common example.
21. Batch printing / student verification
22. If multiple students bring bills, accept a CSV file with columns: name, units, pu (or blank if slab), type, meter\_rent, other, ed. Script reads CSV, computes, and writes an output CSV with components and difference vs bill total.
23. I can add this CSV import/export quickly.
24. Running in VS Code & debugging tips
25. In VS Code, select the Python interpreter you want (bottom-right). If you use conda, ensure the environment is activated or use Anaconda Prompt.
26. All components (EC, FC, CC, ED, Other, Total) are printed and can be traced to bill lines.Script handles invalid inputs gracefully (negative units, unknown customer type).

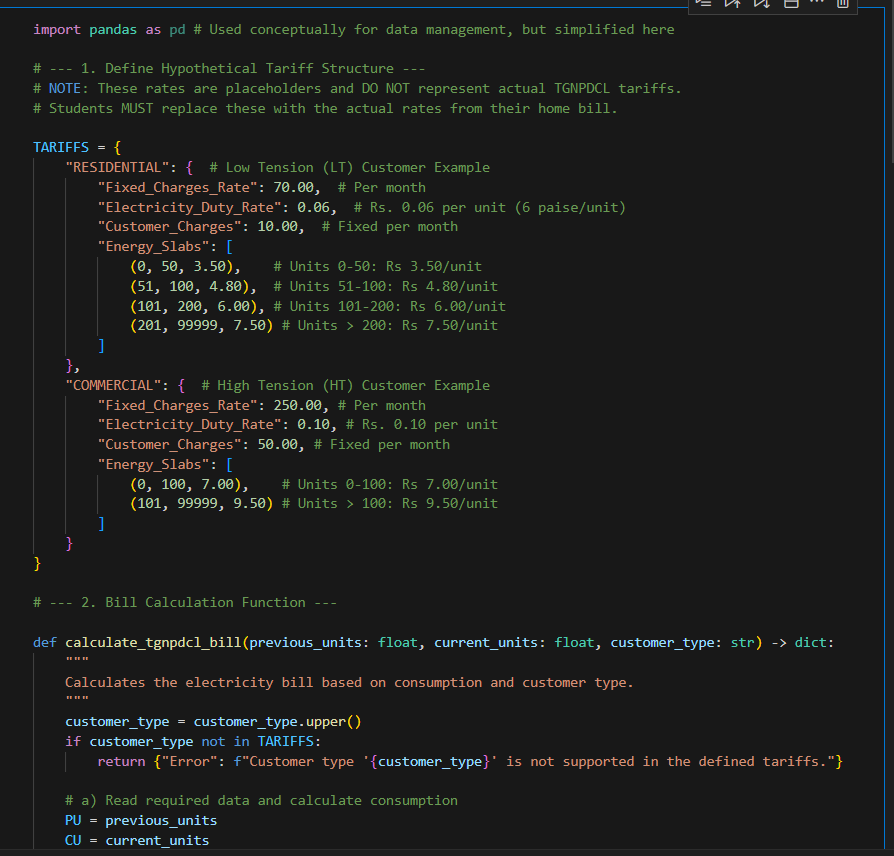
A computer screen shot of a program

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

**We can also use Google Colab with Gemini:**

****

**Output:**

**A screenshot of a computer program

AI-generated content may be incorrect.**