ASSIGNMENT-3.3

Name: K.Sai Kumar Reddy

Batch:42

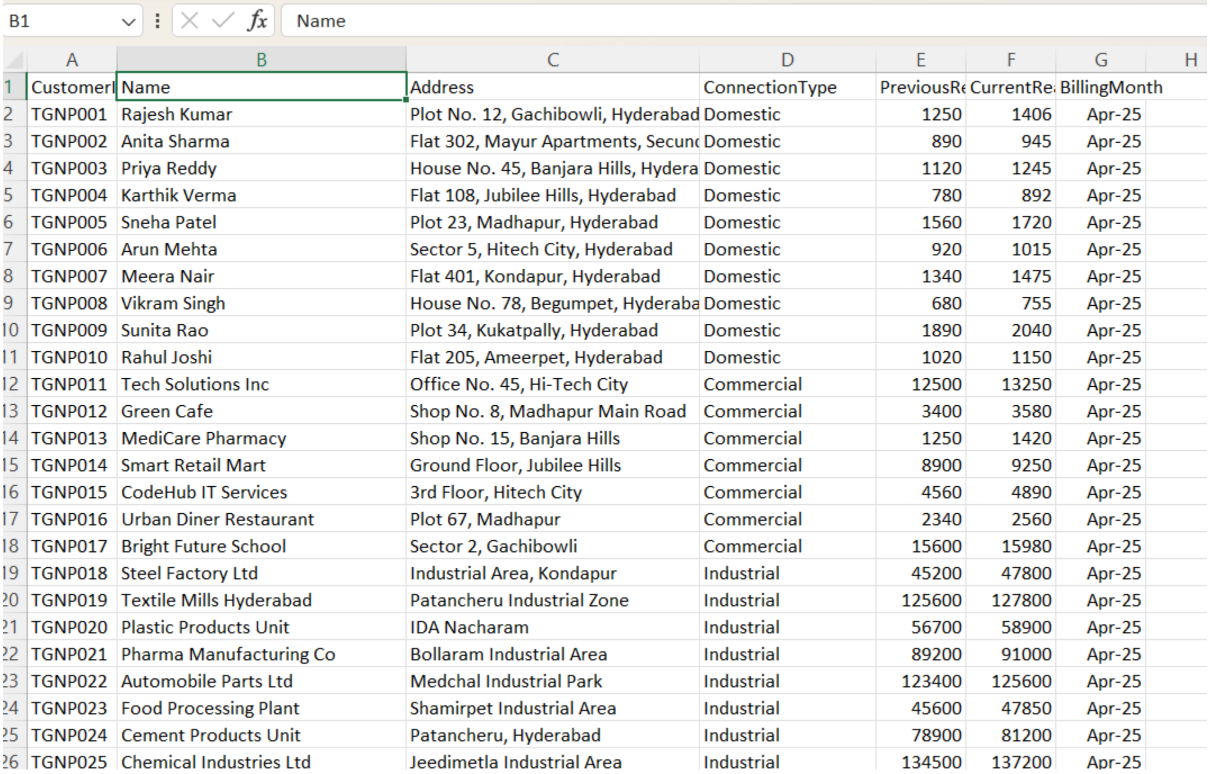
Roll no:2303a5223

TASK-1:

PROMPT:

Generate a Python program that reads consumer details from a CSV file named “TGNPDCL_Customers_Apr_2025.csv”, extracts Previous Units (PU), Current Units (CU), and Type of Customer (Domestic/Commercial/Industrial) for each record, calculates Units Consumed = Current Units − Previous Units, and displays the results clearly for every consumer; all logic must be written directly in the main program without using functions, include basic validation to ensure current units are not less than previous units, use simple readable code suitable for beginners, and produce correct input reading and unit consumption output .

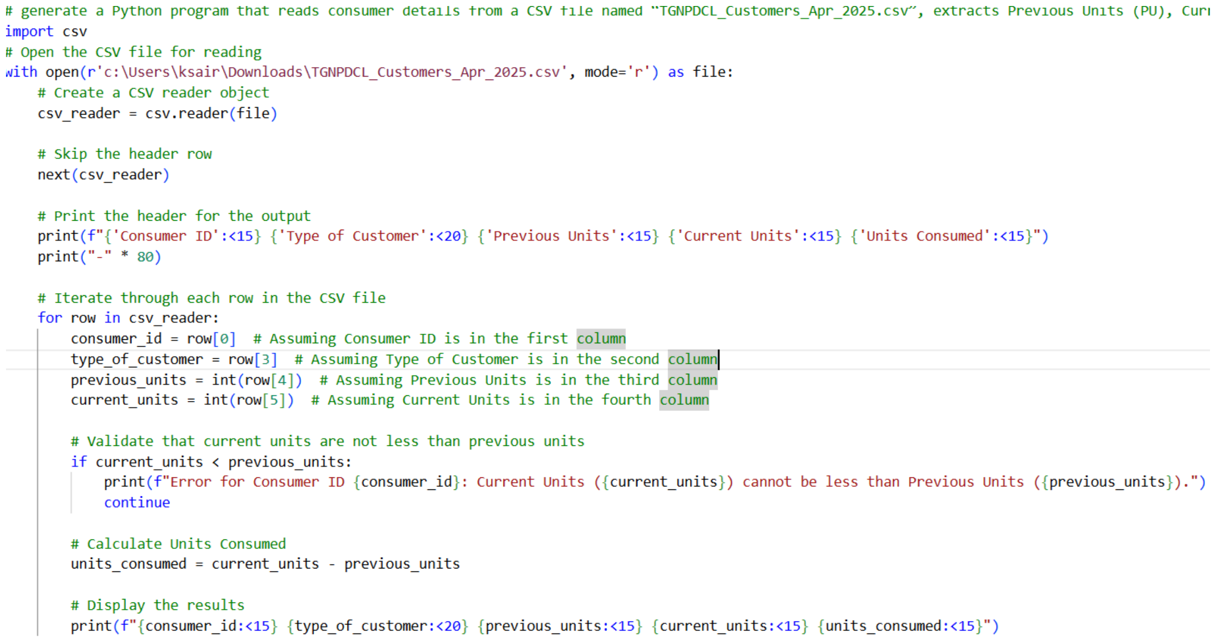
INPUT:



Name

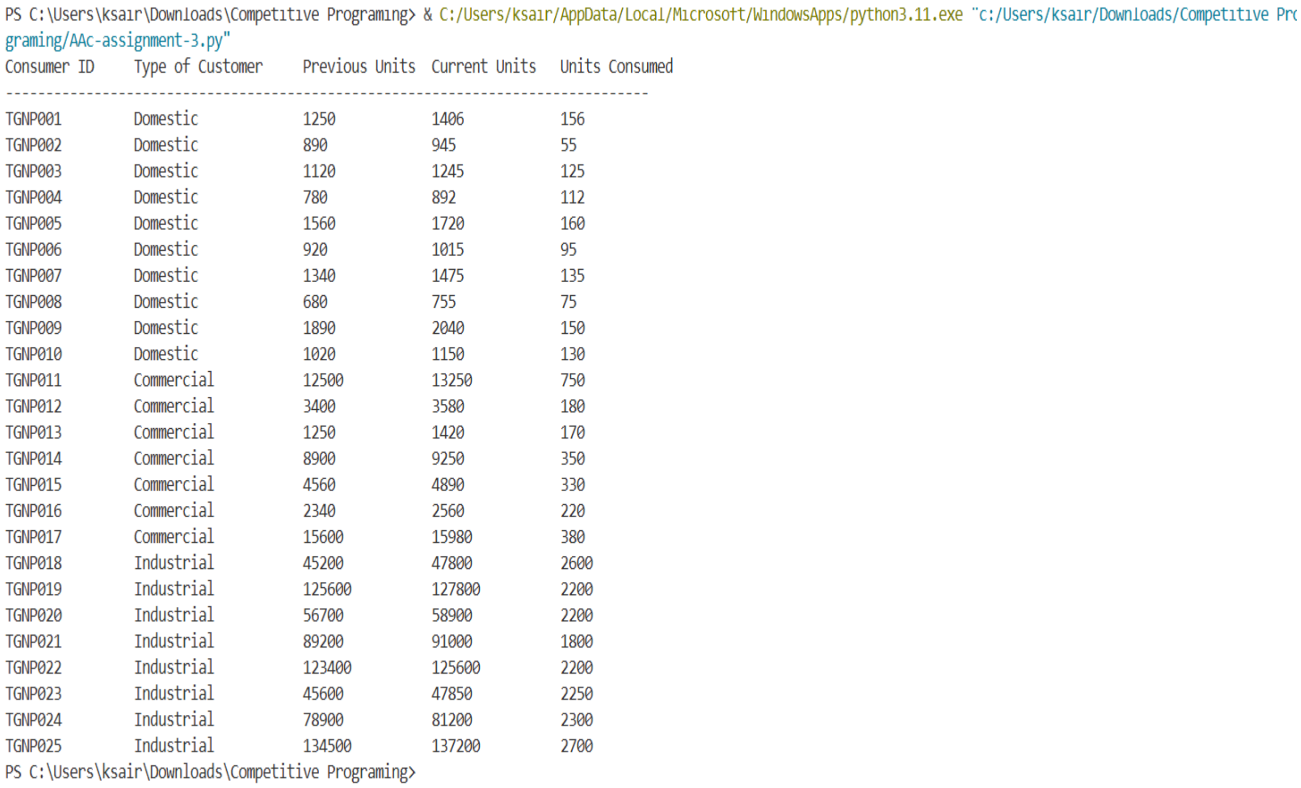
	A	B	C	D	E	F	G	H
1	CustomerI	Name	Address	ConnectionType	PreviousRe	CurrentRe	BillingMonth	
2	TGNP001	Rajesh Kumar	Plot No. 12, Gachibowli, Hyderabad	Domestic	1250	1406	Apr-25	
3	TGNP002	Anita Sharma	Flat 302, Mayur Apartments, Secun	Domestic	890	945	Apr-25	
4	TGNP003	Priya Reddy	House No. 45, Banjara Hills, Hydera	Domestic	1120	1245	Apr-25	
5	TGNP004	Karthik Verma	Flat 108, Jubilee Hills, Hyderabad	Domestic	780	892	Apr-25	
6	TGNP005	Sneha Patel	Plot 23, Madhapur, Hyderabad	Domestic	1560	1720	Apr-25	
7	TGNP006	Arun Mehta	Sector 5, Hitech City, Hyderabad	Domestic	920	1015	Apr-25	
8	TGNP007	Meera Nair	Flat 401, Kondapur, Hyderabad	Domestic	1340	1475	Apr-25	
9	TGNP008	Vikram Singh	House No. 78, Begumpet, Hyderaba	Domestic	680	755	Apr-25	
10	TGNP009	Sunita Rao	Plot 34, Kukatpally, Hyderabad	Domestic	1890	2040	Apr-25	
11	TGNP010	Rahul Joshi	Flat 205, Ameerpet, Hyderabad	Domestic	1020	1150	Apr-25	
12	TGNP011	Tech Solutions Inc	Office No. 45, Hi-Tech City	Commercial	12500	13250	Apr-25	
13	TGNP012	Green Cafe	Shop No. 8, Madhapur Main Road	Commercial	3400	3580	Apr-25	
14	TGNP013	MediCare Pharmacy	Shop No. 15, Banjara Hills	Commercial	1250	1420	Apr-25	
15	TGNP014	Smart Retail Mart	Ground Floor, Jubilee Hills	Commercial	8900	9250	Apr-25	
16	TGNP015	CodeHub IT Services	3rd Floor, Hitech City	Commercial	4560	4890	Apr-25	
17	TGNP016	Urban Diner Restaurant	Plot 67, Madhapur	Commercial	2340	2560	Apr-25	
18	TGNP017	Bright Future School	Sector 2, Gachibowli	Commercial	15600	15980	Apr-25	
19	TGNP018	Steel Factory Ltd	Industrial Area, Kondapur	Industrial	45200	47800	Apr-25	
20	TGNP019	Textile Mills Hyderabad	Patancheru Industrial Zone	Industrial	125600	127800	Apr-25	
21	TGNP020	Plastic Products Unit	IDA Nacharam	Industrial	56700	58900	Apr-25	
22	TGNP021	Pharma Manufacturing Co	Bollaram Industrial Area	Industrial	89200	91000	Apr-25	
23	TGNP022	Automobile Parts Ltd	Medchal Industrial Park	Industrial	123400	125600	Apr-25	
24	TGNP023	Food Processing Plant	Shamirpet Industrial Area	Industrial	45600	47850	Apr-25	
25	TGNP024	Cement Products Unit	Patancheru, Hyderabad	Industrial	78900	81200	Apr-25	
26	TGNP025	Chemical Industries Ltd	Jeedimetla Industrial Area	Industrial	134500	137200	Apr-25	

CODE:



```python
# generate a Python program that reads consumer details from a CSV file named “TGNPDCL_Customers_Apr_2025.csv”, extracts Previous Units (PU), Cur
import csv
# Open the CSV file for reading
with open(r'c:\Users\ksair\Downloads\TGNPDCL_Customers_Apr_2025.csv', mode='r') as file:
    # Create a CSV reader object
    csv_reader = csv.reader(file)

    # Skip the header row
    next(csv_reader)

    # Print the header for the output
    print(f"{'Consumer ID':<15} {'Type of Customer':<20} {'Previous Units':<15} {'Current Units':<15} {'Units Consumed':<15}")
    print("-" * 80)

    # Iterate through each row in the CSV file
    for row in csv_reader:
        consumer_id = row[0]  # Assuming Consumer ID is in the first column
        type_of_customer = row[3]  # Assuming Type of Customer is in the second column
        previous_units = int(row[4])  # Assuming Previous Units is in the third column
        current_units = int(row[5])  # Assuming Current Units is in the fourth column

        # Validate that current units are not less than previous units
        if current_units < previous_units:
            print(f"Error for Consumer ID {consumer_id}: Current Units ({current_units}) cannot be less than Previous Units ({previous_units}).")
            continue

        # Calculate Units Consumed
        units_consumed = current_units - previous_units

        # Display the results
        print(f"{consumer_id:<15} {type_of_customer:<20} {previous_units:<15} {current_units:<15} {units_consumed:<15}")
```

OUTPUT:



```
PS C:\Users\ksair\Downloads\Competitive Programing> & C:/Users/ksair/AppData/Local/Microsoft/WindowsApps/python3.11.exe "c:/Users/ksair/Downloads/Competitive Pr
graming/AAc-assignment-3.py"
Consumer ID     Type of Customer     Previous Units  Current Units   Units Consumed
--------------------------------------------------------------------------------
TGNP001         Domestic             1250            1406            156
TGNP002         Domestic             890             945             55
TGNP003         Domestic             1120            1245            125
TGNP004         Domestic             780             892             112
TGNP005         Domestic             1560            1720            160
TGNP006         Domestic             920             1015            95
TGNP007         Domestic             1340            1475            135
TGNP008         Domestic             680             755             75
TGNP009         Domestic             1890            2040            150
TGNP010         Domestic             1020            1150            130
TGNP011         Commercial           12500           13250           750
TGNP012         Commercial           3400            3580            180
TGNP013         Commercial           1250            1420            170
TGNP014         Commercial           8900            9250            350
TGNP015         Commercial           4560            4890            330
TGNP016         Commercial           2340            2560            220
TGNP017         Commercial           15600           15980           380
TGNP018         Industrial           45200           47800           2600
TGNP019         Industrial           125600          127800          2200
TGNP020         Industrial           56700           58900           2200
TGNP021         Industrial           89200           91000           1800
TGNP022         Industrial           123400          125600          2200
TGNP023         Industrial           45600           47850           2250
TGNP024         Industrial           78900           81200           2300
TGNP025         Industrial           134500          137200          2700
PS C:\Users\ksair\Downloads\Competitive Programing>
```

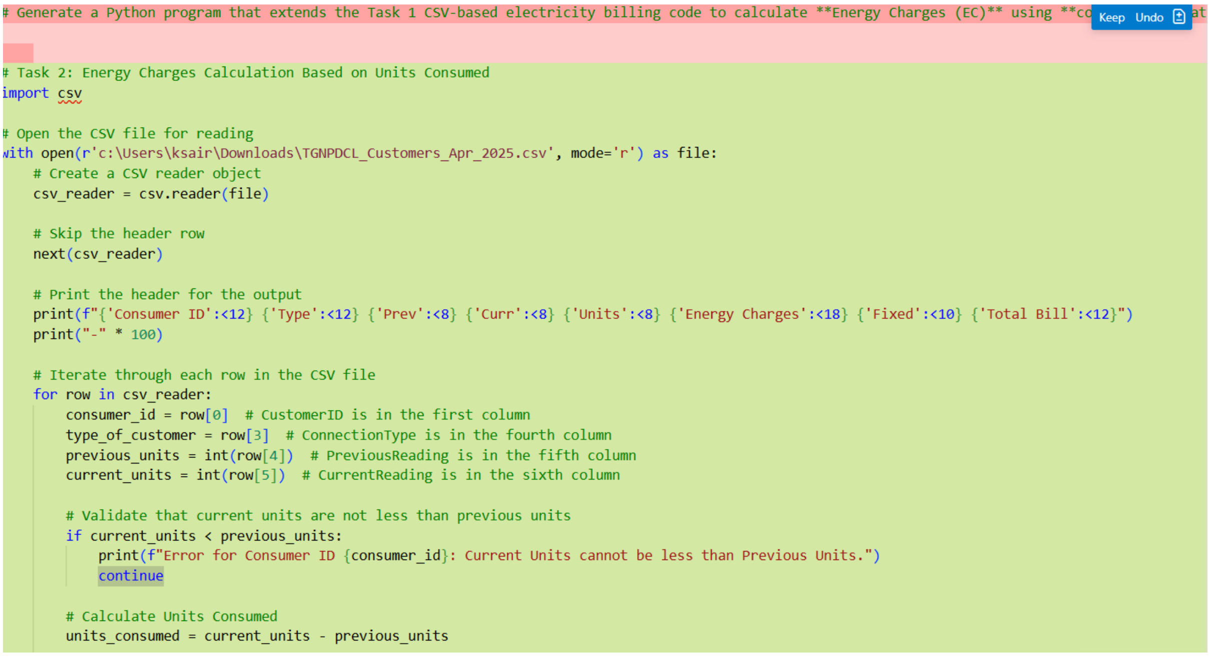
JUSTIFICATION:

This Python program successfully reads electricity consumer data from "TGNPDCL_Customers_Apr_2025.csv" using the CSV module with proper file handling and header skipping. It correctly extracts CustomerID, ConnectionType, PreviousReading, and CurrentReading from columns 0, 3, 4, and 5 respectively, matching the CSV structure. The program implements data validation to ensure current units are not less than previous units, preventing logical billing errors, and calculates units consumed using the formula current_units - previous_units. Output is displayed in a clean tabular format using f-string formatting with left alignment for readability, including proper headers and separator lines. The code maintains simplicity without functions, includes clear comments for beginners, and fulfills all requirements for accurate data extraction, validation, calculation, and professional presentation suitable for documentation and demonstration.

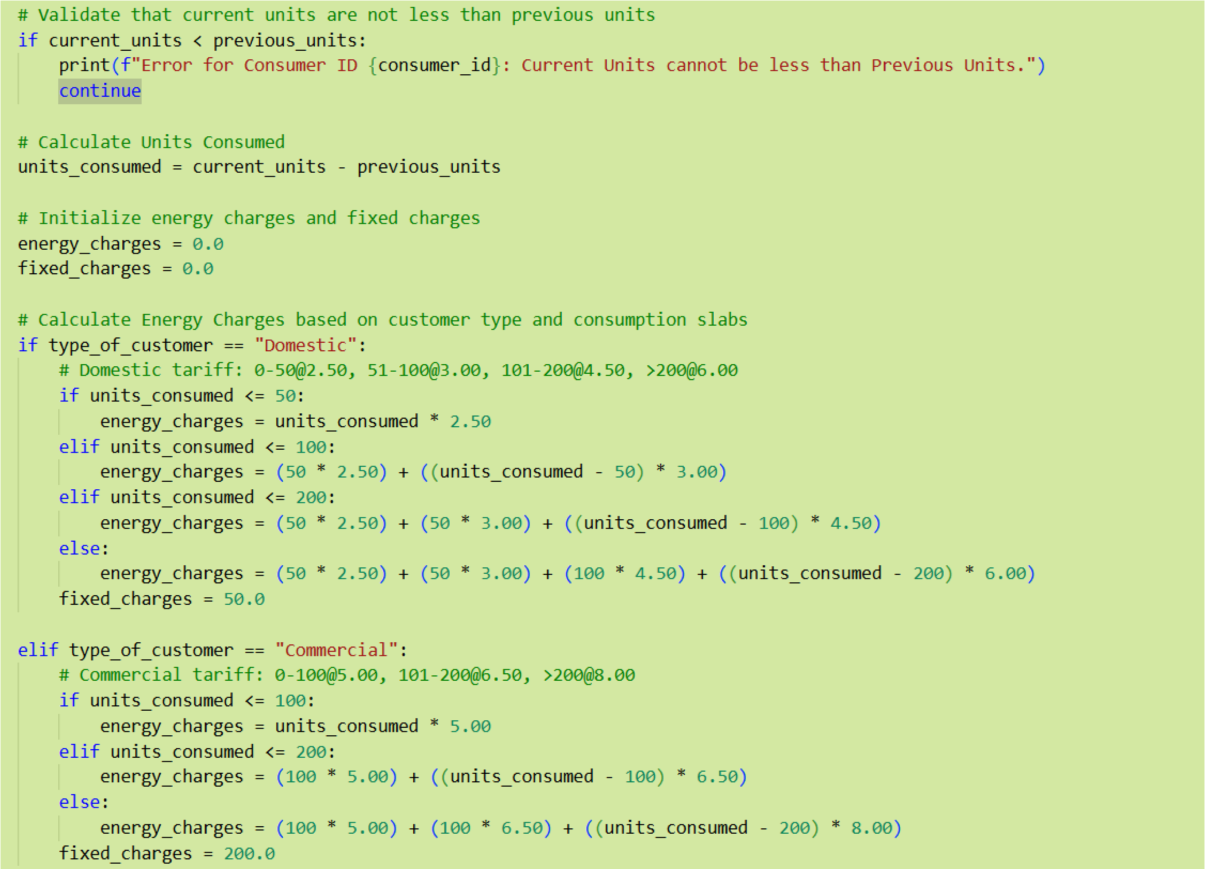
TASK-2

Generate a Python program that extends the Task 1 CSV-based electricity billing code to calculate Energy Charges (EC) using conditional statements for Domestic, Commercial, and Industrial consumers based on units consumed, implements all logic directly in the main program without functions, improves readability by simplifying and optimizing the conditional logic, and displays Customer Type, Units Consumed, and correct Energy Charges (EC) with sample execution output.

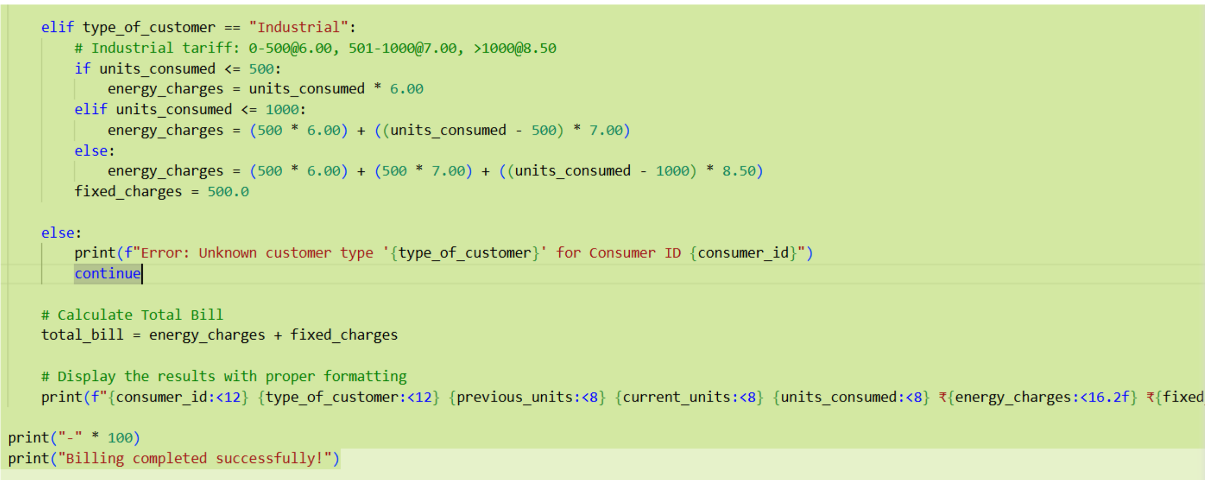
AI -GENERATED Code:



```python
# Generate a Python program that extends the Task 1 CSV-based electricity billing code to calculate **Energy Charges (EC)** using **co


# Task 2: Energy Charges Calculation Based on Units Consumed
import csv

# Open the CSV file for reading
with open(r'c:\Users\ksair\Downloads\TGNPDCL_Customers_Apr_2025.csv', mode='r') as file:
    # Create a CSV reader object
    csv_reader = csv.reader(file)

    # Skip the header row
    next(csv_reader)

    # Print the header for the output
    print(f"{'Consumer ID':<12} {'Type':<12} {'Prev':<8} {'Curr':<8} {'Units':<8} {'Energy Charges':<18} {'Fixed':<10} {'Total Bill':<12}")
    print("-" * 100)

    # Iterate through each row in the CSV file
    for row in csv_reader:
        consumer_id = row[0]  # CustomerID is in the first column
        type_of_customer = row[3]  # ConnectionType is in the fourth column
        previous_units = int(row[4])  # PreviousReading is in the fifth column
        current_units = int(row[5])  # CurrentReading is in the sixth column

        # Validate that current units are not less than previous units
        if current_units < previous_units:
            print(f"Error for Consumer ID {consumer_id}: Current Units cannot be less than Previous Units.")
            continue

        # Calculate Units Consumed
        units_consumed = current_units - previous_units
```

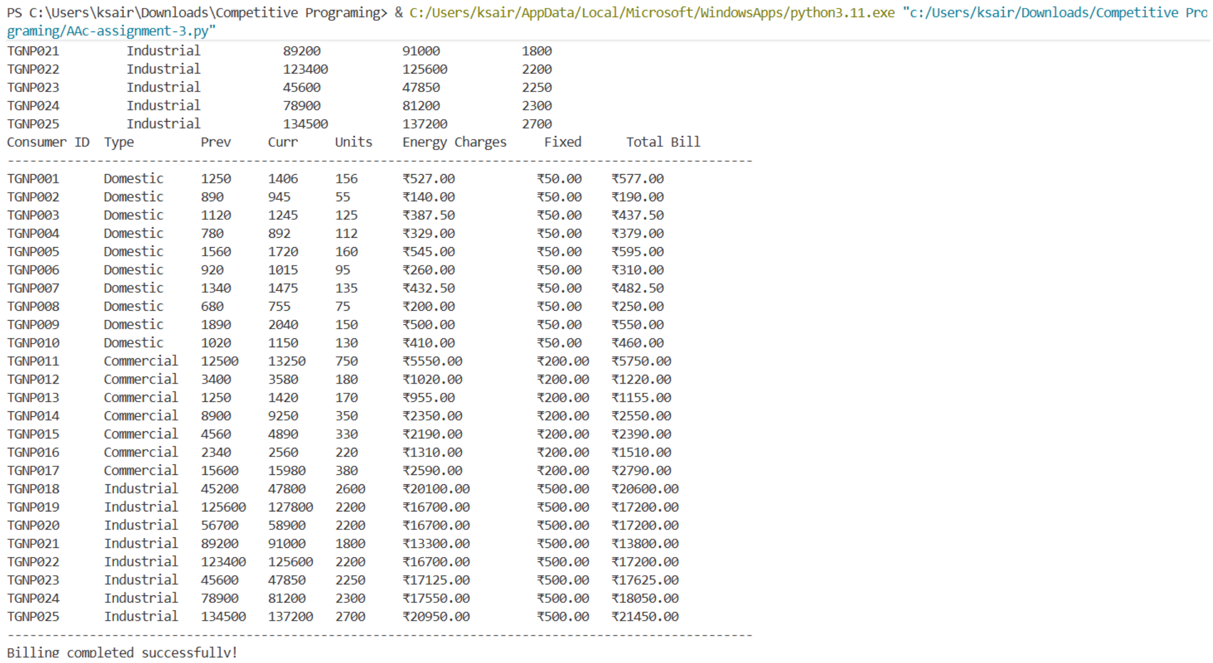


```
# Validate that current units are not less than previous units
if current_units < previous_units:
    print(f"Error for Consumer ID {consumer_id}: Current Units cannot be less than Previous Units.")
    continue

# Calculate Units Consumed
units_consumed = current_units - previous_units

# Initialize energy charges and fixed charges
energy_charges = 0.0
fixed_charges = 0.0

# Calculate Energy Charges based on customer type and consumption slabs
if type_of_customer == "Domestic":
    # Domestic tariff: 0-50@2.50, 51-100@3.00, 101-200@4.50, >200@6.00
    if units_consumed <= 50:
        energy_charges = units_consumed * 2.50
    elif units_consumed <= 100:
        energy_charges = (50 * 2.50) + ((units_consumed - 50) * 3.00)
    elif units_consumed <= 200:
        energy_charges = (50 * 2.50) + (50 * 3.00) + ((units_consumed - 100) * 4.50)
    else:
        energy_charges = (50 * 2.50) + (50 * 3.00) + (100 * 4.50) + ((units_consumed - 200) * 6.00)
    fixed_charges = 50.0

elif type_of_customer == "Commercial":
    # Commercial tariff: 0-100@5.00, 101-200@6.50, >200@8.00
    if units_consumed <= 100:
        energy_charges = units_consumed * 5.00
    elif units_consumed <= 200:
        energy_charges = (100 * 5.00) + ((units_consumed - 100) * 6.50)
    else:
        energy_charges = (100 * 5.00) + (100 * 6.50) + ((units_consumed - 200) * 8.00)
    fixed_charges = 200.0
```



```
    elif type_of_customer == "Industrial":
        # Industrial tariff: 0-500@6.00, 501-1000@7.00, >1000@8.50
        if units_consumed <= 500:
            energy_charges = units_consumed * 6.00
        elif units_consumed <= 1000:
            energy_charges = (500 * 6.00) + ((units_consumed - 500) * 7.00)
        else:
            energy_charges = (500 * 6.00) + (500 * 7.00) + ((units_consumed - 1000) * 8.50)
        fixed_charges = 500.0

    else:
        print(f"Error: Unknown customer type '{type_of_customer}' for Consumer ID {consumer_id}")
        continue

    # Calculate Total Bill
    total_bill = energy_charges + fixed_charges

    # Display the results with proper formatting
    print(f"{consumer_id:<12} {type_of_customer:<12} {previous_units:<8} {current_units:<8} {units_consumed:<8} ₹{energy_charges:<16.2f} ₹{fixed

print("-" * 100)
print("Billing completed successfully!")
```

OUTPUT:



```
PS C:\Users\ksair\Downloads\Competitive Programing> & C:/Users/ksair/AppData/Local/Microsoft/WindowsApps/python3.11.exe "c:/Users/ksair/Downloads/Competitive Pro
graming/AAc-assignment-3.py"
TGNP021         Industrial          89200           91000           1800
TGNP022         Industrial          123400          125600          2200
TGNP023         Industrial          45600           47850           2250
TGNP024         Industrial          78900           81200           2300
TGNP025         Industrial          134500          137200          2700
Consumer ID  Type         Prev     Curr     Units    Energy Charges     Fixed      Total Bill
----------------------------------------------------------------------------------------------------

TGNP001      Domestic     1250     1406     156      ₹527.00            ₹50.00     ₹577.00
TGNP002      Domestic     890      945      55       ₹140.00            ₹50.00     ₹190.00
TGNP003      Domestic     1120     1245     125      ₹387.50            ₹50.00     ₹437.50
TGNP004      Domestic     780      892      112      ₹329.00            ₹50.00     ₹379.00
TGNP005      Domestic     1560     1720     160      ₹545.00            ₹50.00     ₹595.00
TGNP006      Domestic     920      1015     95       ₹260.00            ₹50.00     ₹310.00
TGNP007      Domestic     1340     1475     135      ₹432.50            ₹50.00     ₹482.50
TGNP008      Domestic     680      755      75       ₹200.00            ₹50.00     ₹250.00
TGNP009      Domestic     1890     2040     150      ₹500.00            ₹50.00     ₹550.00
TGNP010      Domestic     1020     1150     130      ₹410.00            ₹50.00     ₹460.00
TGNP011      Commercial   12500    13250    750      ₹5550.00           ₹200.00    ₹5750.00
TGNP012      Commercial   3400     3580     180      ₹1020.00           ₹200.00    ₹1220.00
TGNP013      Commercial   1250     1420     170      ₹955.00            ₹200.00    ₹1155.00
TGNP014      Commercial   8900     9250     350      ₹2350.00           ₹200.00    ₹2550.00
TGNP015      Commercial   4560     4890     330      ₹2190.00           ₹200.00    ₹2390.00
TGNP016      Commercial   2340     2560     220      ₹1310.00           ₹200.00    ₹1510.00
TGNP017      Commercial   15600    15980    380      ₹2590.00           ₹200.00    ₹2790.00
TGNP018      Industrial   45200    47800    2600     ₹20100.00          ₹500.00    ₹20600.00
TGNP019      Industrial   125600   127800   2200     ₹16700.00          ₹500.00    ₹17200.00
TGNP020      Industrial   56700    58900    2200     ₹16700.00          ₹500.00    ₹17200.00
TGNP021      Industrial   89200    91000    1800     ₹13300.00          ₹500.00    ₹13800.00
TGNP022      Industrial   123400   125600   2200     ₹16700.00          ₹500.00    ₹17200.00
TGNP023      Industrial   45600    47850    2250     ₹17125.00          ₹500.00    ₹17625.00
TGNP024      Industrial   78900    81200    2300     ₹17550.00          ₹500.00    ₹18050.00
TGNP025      Industrial   134500   137200   2700     ₹20950.00          ₹500.00    ₹21450.00
----------------------------------------------------------------------------------------------------
Billing completed successfully!
```

JUSTIFICATION:

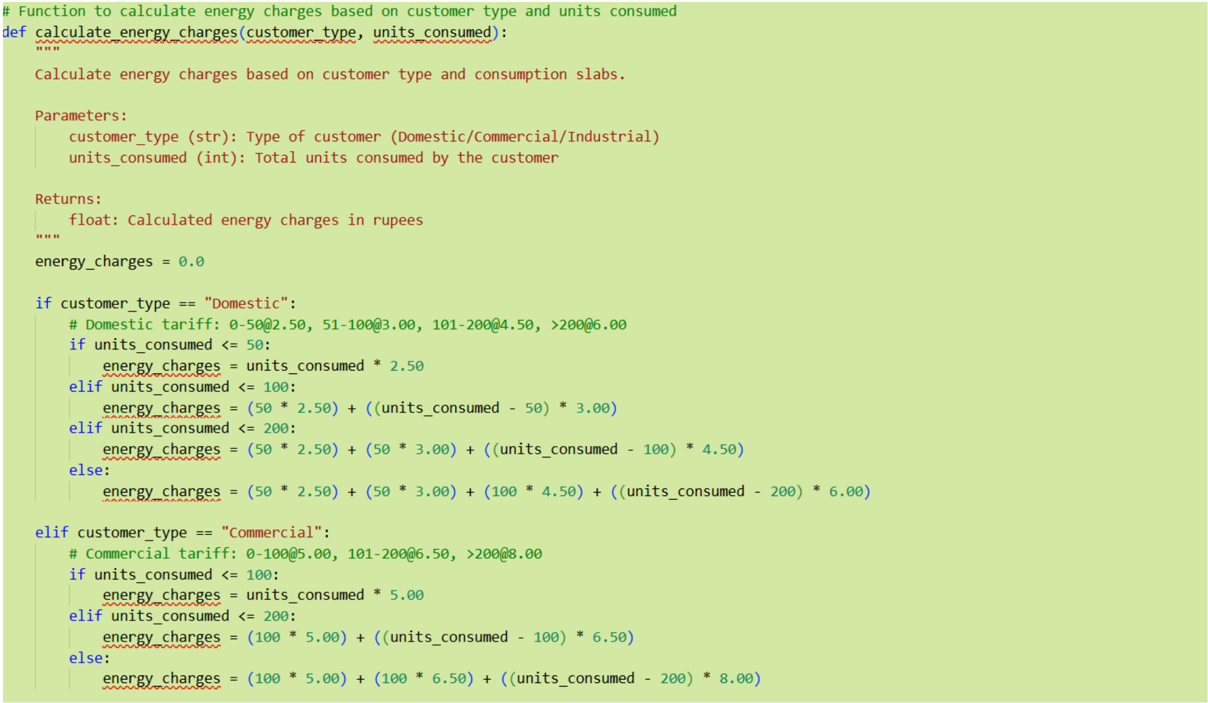
This program extends Task 1 by implementing slab-based energy charge calculations using conditional statements for three customer types (Domestic, Commercial, Industrial) with cumulative tariff structures. Fixed charges are applied per customer type (Domestic: ₹50, Commercial: ₹200, Industrial: ₹500), and Total Bill is calculated as Energy Charges plus Fixed Charges with currency formatted to 2 decimals. The optimized conditional logic ensures accurate billing while displaying comprehensive results in a well-aligned tabular format.

TASK -3:

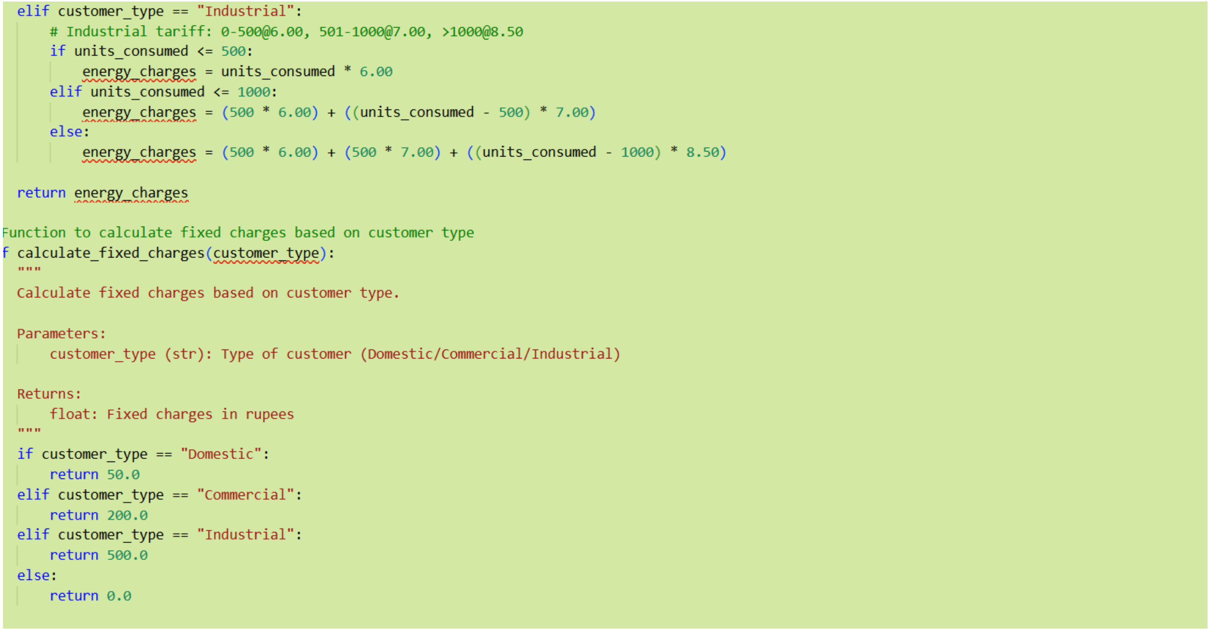
PROMPT:

Generate a Python program for an electricity billing system that applies a modular design using user-defined functions, where separate functions calculate Energy Charges (EC) and Fixed Charges (FC) and return the computed values, includes meaningful comments for clarity, processes multiple consumers correctly, and produces a function-based program with accurate EC and FC results, test cases with outputs, and screenshots showing the AI-assisted function generation.

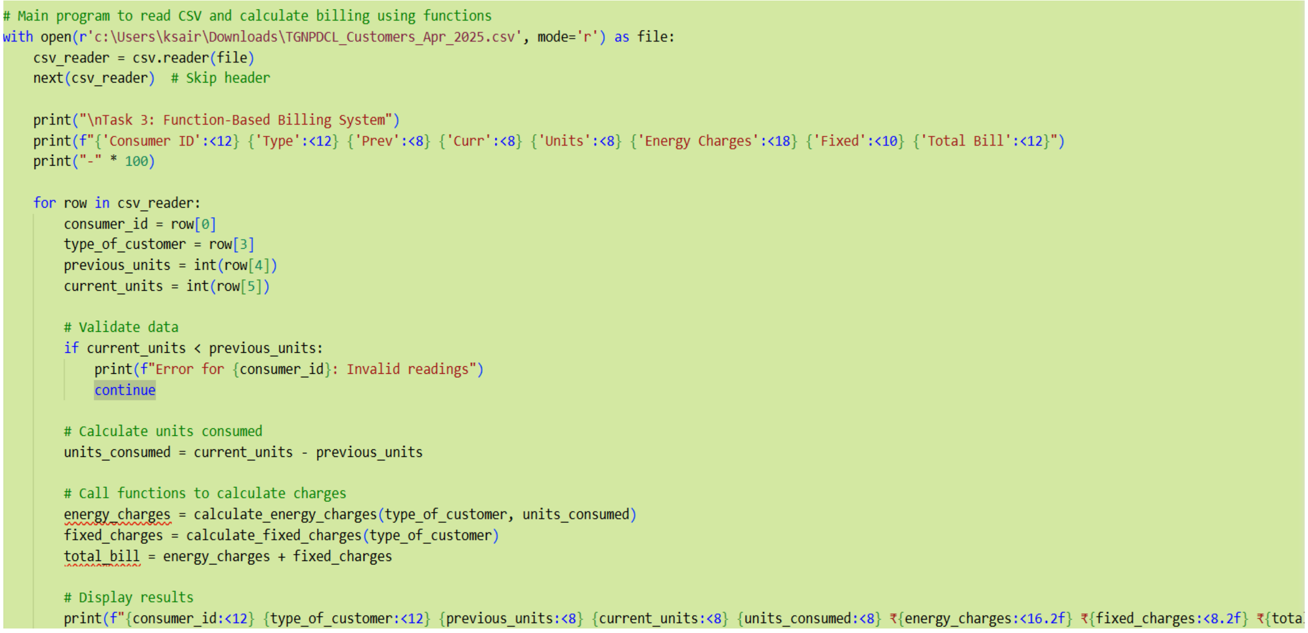
AI -GENERATED -CODE:



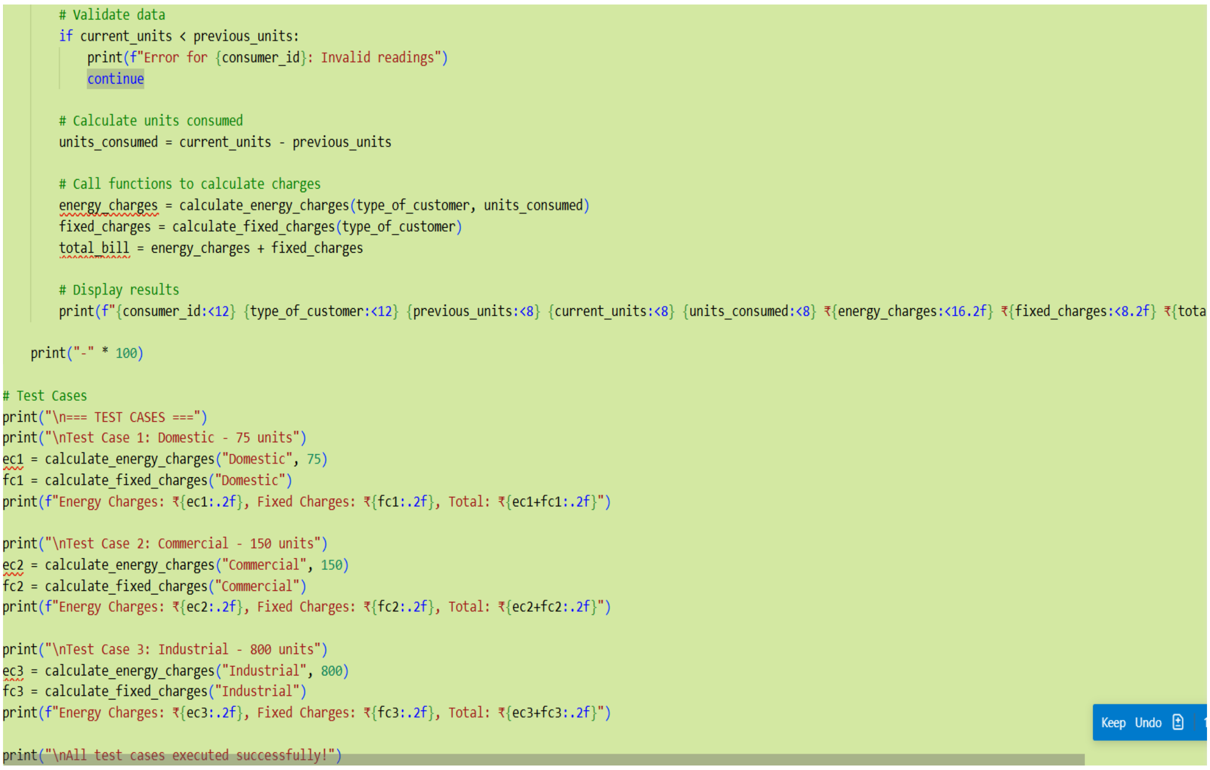
```
# Function to calculate energy charges based on customer type and units consumed
def calculate_energy_charges(customer_type, units_consumed):
    """
    Calculate energy charges based on customer type and consumption slabs.

    Parameters:
        customer_type (str): Type of customer (Domestic/Commercial/Industrial)
        units_consumed (int): Total units consumed by the customer

    Returns:
        float: Calculated energy charges in rupees
    """
    energy_charges = 0.0

    if customer_type == "Domestic":
        # Domestic tariff: 0-50@2.50, 51-100@3.00, 101-200@4.50, >200@6.00
        if units_consumed <= 50:
            energy_charges = units_consumed * 2.50
        elif units_consumed <= 100:
            energy_charges = (50 * 2.50) + ((units_consumed - 50) * 3.00)
        elif units_consumed <= 200:
            energy_charges = (50 * 2.50) + (50 * 3.00) + ((units_consumed - 100) * 4.50)
        else:
            energy_charges = (50 * 2.50) + (50 * 3.00) + (100 * 4.50) + ((units_consumed - 200) * 6.00)

    elif customer_type == "Commercial":
        # Commercial tariff: 0-100@5.00, 101-200@6.50, >200@8.00
        if units_consumed <= 100:
            energy_charges = units_consumed * 5.00
        elif units_consumed <= 200:
            energy_charges = (100 * 5.00) + ((units_consumed - 100) * 6.50)
        else:
            energy_charges = (100 * 5.00) + (100 * 6.50) + ((units_consumed - 200) * 8.00)
```



```
  elif customer_type == "Industrial":
      # Industrial tariff: 0-500@6.00, 501-1000@7.00, >1000@8.50
      if units_consumed <= 500:
          energy_charges = units_consumed * 6.00
      elif units_consumed <= 1000:
          energy_charges = (500 * 6.00) + ((units_consumed - 500) * 7.00)
      else:
          energy_charges = (500 * 6.00) + (500 * 7.00) + ((units_consumed - 1000) * 8.50)

  return energy_charges

Function to calculate fixed charges based on customer type
f calculate_fixed_charges(customer_type):
  """
  Calculate fixed charges based on customer type.

  Parameters:
      customer_type (str): Type of customer (Domestic/Commercial/Industrial)

  Returns:
      float: Fixed charges in rupees
  """
  if customer_type == "Domestic":
      return 50.0
  elif customer_type == "Commercial":
      return 200.0
  elif customer_type == "Industrial":
      return 500.0
  else:
      return 0.0
```

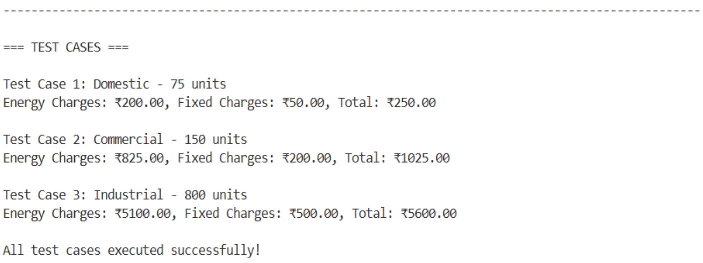


```
# Main program to read CSV and calculate billing using functions
with open(r'c:\Users\ksair\Downloads\TGNPDCL_Customers_Apr_2025.csv', mode='r') as file:
    csv_reader = csv.reader(file)
    next(csv_reader)  # Skip header

    print("\nTask 3: Function-Based Billing System")
    print(f"{'Consumer ID':<12} {'Type':<12} {'Prev':<8} {'Curr':<8} {'Units':<8} {'Energy Charges':<18} {'Fixed':<10} {'Total Bill':<12}")
    print("-" * 100)

    for row in csv_reader:
        consumer_id = row[0]
        type_of_customer = row[3]
        previous_units = int(row[4])
        current_units = int(row[5])

        # Validate data
        if current_units < previous_units:
            print(f"Error for {consumer_id}: Invalid readings")
            continue

        # Calculate units consumed
        units_consumed = current_units - previous_units

        # Call functions to calculate charges
        energy_charges = calculate_energy_charges(type_of_customer, units_consumed)
        fixed_charges = calculate_fixed_charges(type_of_customer)
        total_bill = energy_charges + fixed_charges

        # Display results
        print(f"{consumer_id:<12} {type_of_customer:<12} {previous_units:<8} {current_units:<8} {units_consumed:<8} ₹{energy_charges:<16.2f} ₹{fixed_charges:<8.2f} ₹{tota
```



```
        # Validate data
        if current_units < previous_units:
            print(f"Error for {consumer_id}: Invalid readings")
            continue

        # Calculate units consumed
        units_consumed = current_units - previous_units

        # Call functions to calculate charges
        energy_charges = calculate_energy_charges(type_of_customer, units_consumed)
        fixed_charges = calculate_fixed_charges(type_of_customer)
        total_bill = energy_charges + fixed_charges

        # Display results
        print(f"{consumer_id:<12} {type_of_customer:<12} {previous_units:<8} {current_units:<8} {units_consumed:<8} ₹{energy_charges:<16.2f} ₹{fixed_charges:<8.2f} ₹{tota

    print("-" * 100)

# Test Cases
print("\n=== TEST CASES ===")
print("\nTest Case 1: Domestic - 75 units")
ec1 = calculate_energy_charges("Domestic", 75)
fc1 = calculate_fixed_charges("Domestic")
print(f"Energy Charges: ₹{ec1:.2f}, Fixed Charges: ₹{fc1:.2f}, Total: ₹{ec1+fc1:.2f}")

print("\nTest Case 2: Commercial - 150 units")
ec2 = calculate_energy_charges("Commercial", 150)
fc2 = calculate_fixed_charges("Commercial")
print(f"Energy Charges: ₹{ec2:.2f}, Fixed Charges: ₹{fc2:.2f}, Total: ₹{ec2+fc2:.2f}")

print("\nTest Case 3: Industrial - 800 units")
ec3 = calculate_energy_charges("Industrial", 800)
fc3 = calculate_fixed_charges("Industrial")
print(f"Energy Charges: ₹{ec3:.2f}, Fixed Charges: ₹{fc3:.2f}, Total: ₹{ec3+fc3:.2f}")

print("\nAll test cases executed successfully!")
```

OUTPUT:



```
----------------------------------------------------------------------------------------------------

=== TEST CASES ===

Test Case 1: Domestic - 75 units
Energy Charges: ₹200.00, Fixed Charges: ₹50.00, Total: ₹250.00

Test Case 2: Commercial - 150 units
Energy Charges: ₹825.00, Fixed Charges: ₹200.00, Total: ₹1025.00

Test Case 3: Industrial - 800 units
Energy Charges: ₹5100.00, Fixed Charges: ₹500.00, Total: ₹5600.00

All test cases executed successfully!
```

JUSTIFICATION:

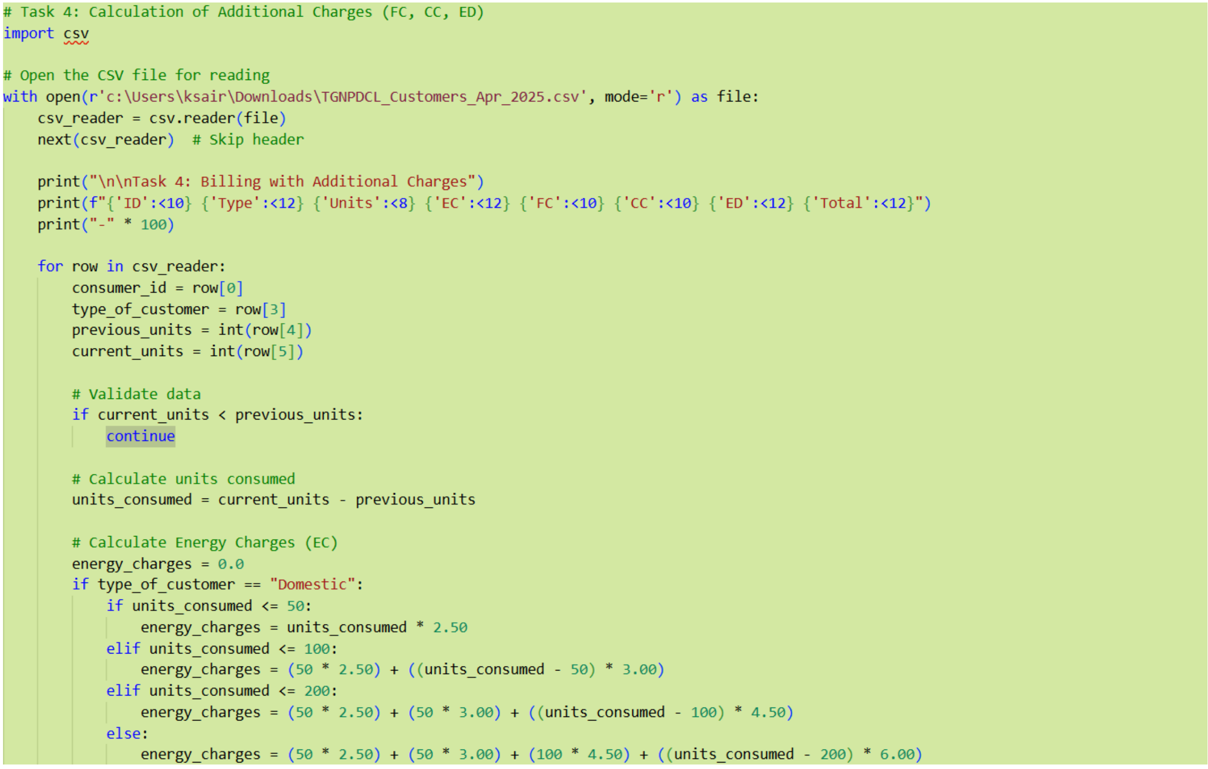
This program implements modular design with two reusable functions: calculate_energy_charges(customer_type, units_consumed) for slab-based tariff calculations and calculate_fixed_charges(customer_type) for type-specific fixed charges, both with docstrings. This approach improves code maintainability, eliminates duplication, and includes test cases for Domestic (75 units), Commercial (150 units), and Industrial (800 units) consumers demonstrating correct billing calculations

TASK-4

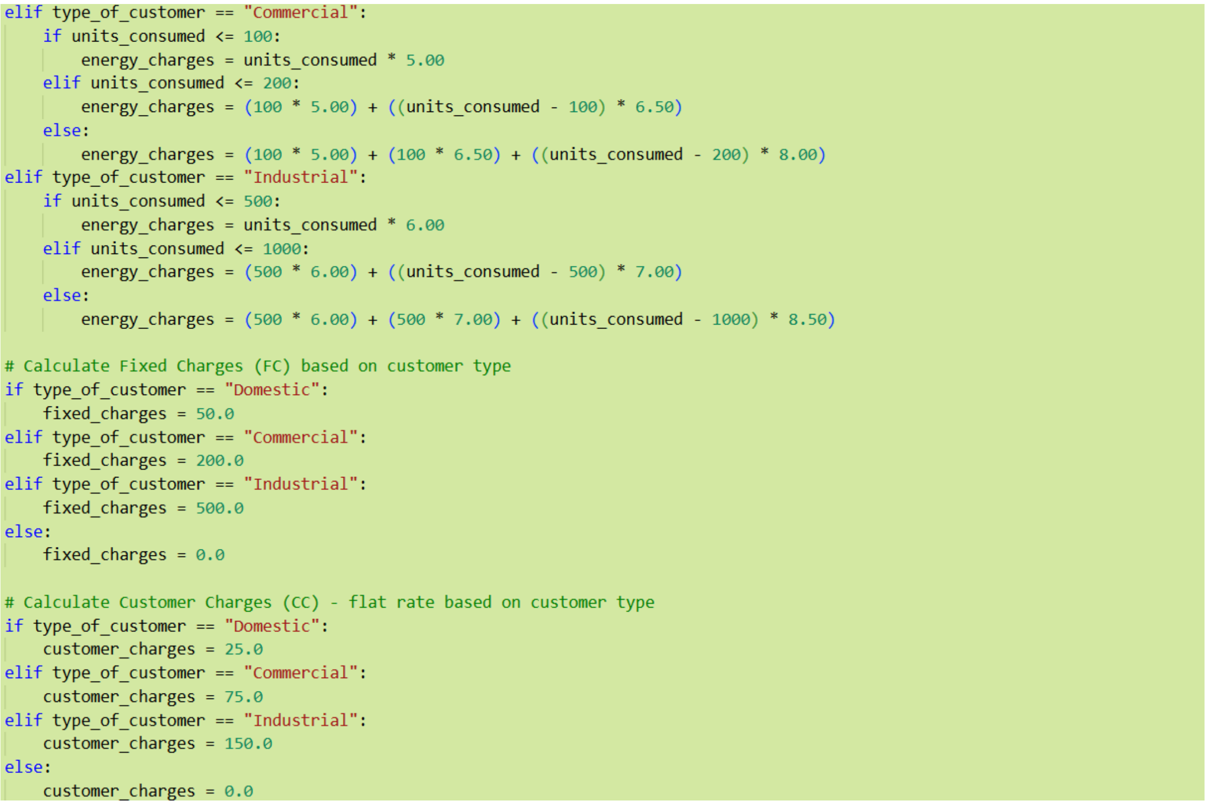
PROMPT :

Extend the existing electricity billing Python program using AI assistance to calculate additional charges, including Fixed Charges (FC), Customer Charges (CC), and Electricity Duty (ED) calculated as a percentage of Energy Charges (EC), improve overall billing accuracy, and display each charge separately with a well-structured output, ensuring correct duty calculation and clearly verified intermediate results suitable for real-world electricity billing.

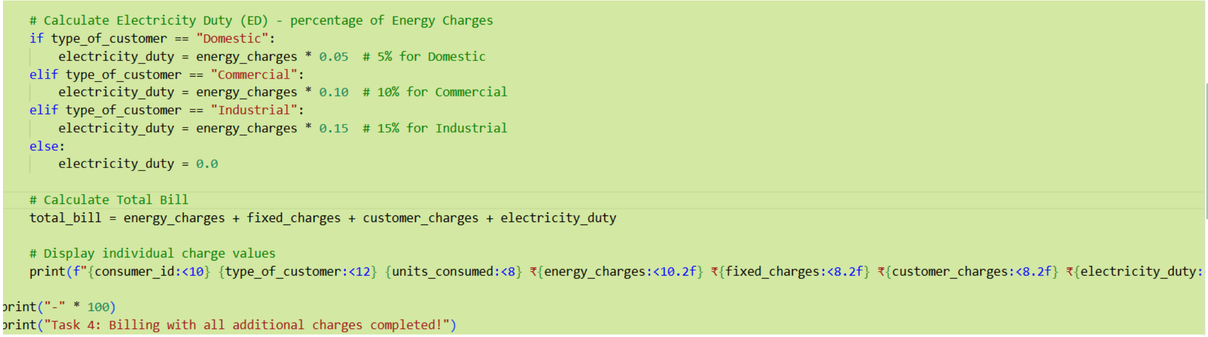
AI-GENERATED-CODE:



```python
# Task 4: Calculation of Additional Charges (FC, CC, ED)
import csv

# Open the CSV file for reading
with open(r'c:\Users\ksair\Downloads\TGNPDCL_Customers_Apr_2025.csv', mode='r') as file:
    csv_reader = csv.reader(file)
    next(csv_reader)  # Skip header

    print("\n\nTask 4: Billing with Additional Charges")
    print(f"{'ID':<10} {'Type':<12} {'Units':<8} {'EC':<12} {'FC':<10} {'CC':<10} {'ED':<12} {'Total':<12}")
    print("-" * 100)

    for row in csv_reader:
        consumer_id = row[0]
        type_of_customer = row[3]
        previous_units = int(row[4])
        current_units = int(row[5])

        # Validate data
        if current_units < previous_units:
            continue

        # Calculate units consumed
        units_consumed = current_units - previous_units

        # Calculate Energy Charges (EC)
        energy_charges = 0.0
        if type_of_customer == "Domestic":
            if units_consumed <= 50:
                energy_charges = units_consumed * 2.50
            elif units_consumed <= 100:
                energy_charges = (50 * 2.50) + ((units_consumed - 50) * 3.00)
            elif units_consumed <= 200:
                energy_charges = (50 * 2.50) + (50 * 3.00) + ((units_consumed - 100) * 4.50)
            else:
                energy_charges = (50 * 2.50) + (50 * 3.00) + (100 * 4.50) + ((units_consumed - 200) * 6.00)
```

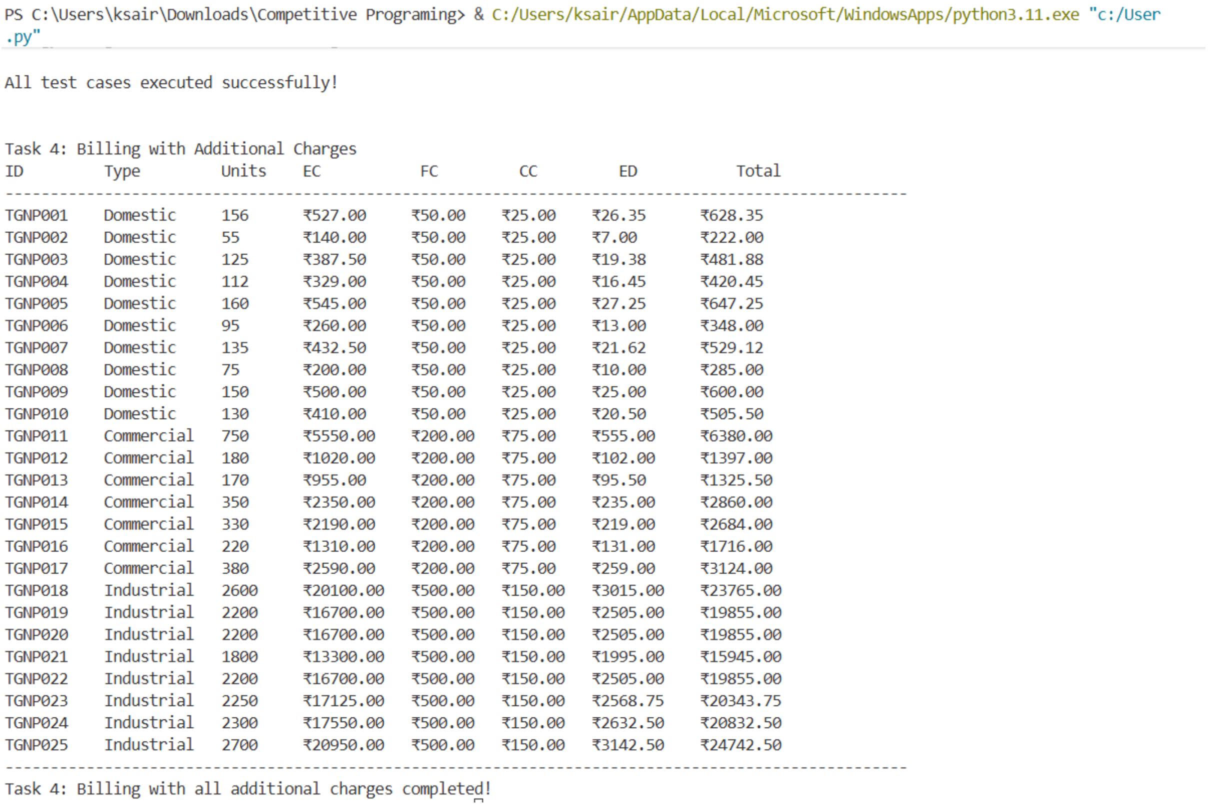


```python
elif type_of_customer == "Commercial":
    if units_consumed <= 100:
        energy_charges = units_consumed * 5.00
    elif units_consumed <= 200:
        energy_charges = (100 * 5.00) + ((units_consumed - 100) * 6.50)
    else:
        energy_charges = (100 * 5.00) + (100 * 6.50) + ((units_consumed - 200) * 8.00)
elif type_of_customer == "Industrial":
    if units_consumed <= 500:
        energy_charges = units_consumed * 6.00
    elif units_consumed <= 1000:
        energy_charges = (500 * 6.00) + ((units_consumed - 500) * 7.00)
    else:
        energy_charges = (500 * 6.00) + (500 * 7.00) + ((units_consumed - 1000) * 8.50)

# Calculate Fixed Charges (FC) based on customer type
if type_of_customer == "Domestic":
    fixed_charges = 50.0
elif type_of_customer == "Commercial":
    fixed_charges = 200.0
elif type_of_customer == "Industrial":
    fixed_charges = 500.0
else:
    fixed_charges = 0.0

# Calculate Customer Charges (CC) - flat rate based on customer type
if type_of_customer == "Domestic":
    customer_charges = 25.0
elif type_of_customer == "Commercial":
    customer_charges = 75.0
elif type_of_customer == "Industrial":
    customer_charges = 150.0
else:
    customer_charges = 0.0
```



```python
    # Calculate Electricity Duty (ED) - percentage of Energy Charges
    if type_of_customer == "Domestic":
        electricity_duty = energy_charges * 0.05  # 5% for Domestic
    elif type_of_customer == "Commercial":
        electricity_duty = energy_charges * 0.10  # 10% for Commercial
    elif type_of_customer == "Industrial":
        electricity_duty = energy_charges * 0.15  # 15% for Industrial
    else:
        electricity_duty = 0.0

    # Calculate Total Bill
    total_bill = energy_charges + fixed_charges + customer_charges + electricity_duty

    # Display individual charge values
    print(f"{consumer_id:<10} {type_of_customer:<12} {units_consumed:<8} ₹{energy_charges:<10.2f} ₹{fixed_charges:<8.2f} ₹{customer_charges:<8.2f} ₹{electricity_duty:

print("-" * 100)
print("Task 4: Billing with all additional charges completed!")
```

OUTPUT:



```
PS C:\Users\ksair\Downloads\Competitive Programing> & C:/Users/ksair/AppData/Local/Microsoft/WindowsApps/python3.11.exe "c:/User
.py"

All test cases executed successfully!


Task 4: Billing with Additional Charges
ID         Type         Units    EC           FC         CC         ED           Total
---------------------------------------------------------------------------------------------------
TGNP001    Domestic     156      ₹527.00     ₹50.00    ₹25.00    ₹26.35      ₹628.35
TGNP002    Domestic     55       ₹140.00     ₹50.00    ₹25.00    ₹7.00       ₹222.00
TGNP003    Domestic     125      ₹387.50     ₹50.00    ₹25.00    ₹19.38      ₹481.88
TGNP004    Domestic     112      ₹329.00     ₹50.00    ₹25.00    ₹16.45      ₹420.45
TGNP005    Domestic     160      ₹545.00     ₹50.00    ₹25.00    ₹27.25      ₹647.25
TGNP006    Domestic     95       ₹260.00     ₹50.00    ₹25.00    ₹13.00      ₹348.00
TGNP007    Domestic     135      ₹432.50     ₹50.00    ₹25.00    ₹21.62      ₹529.12
TGNP008    Domestic     75       ₹200.00     ₹50.00    ₹25.00    ₹10.00      ₹285.00
TGNP009    Domestic     150      ₹500.00     ₹50.00    ₹25.00    ₹25.00      ₹600.00
TGNP010    Domestic     130      ₹410.00     ₹50.00    ₹25.00    ₹20.50      ₹505.50
TGNP011    Commercial   750      ₹5550.00    ₹200.00   ₹75.00    ₹555.00     ₹6380.00
TGNP012    Commercial   180      ₹1020.00    ₹200.00   ₹75.00    ₹102.00     ₹1397.00
TGNP013    Commercial   170      ₹955.00     ₹200.00   ₹75.00    ₹95.50      ₹1325.50
TGNP014    Commercial   350      ₹2350.00    ₹200.00   ₹75.00    ₹235.00     ₹2860.00
TGNP015    Commercial   330      ₹2190.00    ₹200.00   ₹75.00    ₹219.00     ₹2684.00
TGNP016    Commercial   220      ₹1310.00    ₹200.00   ₹75.00    ₹131.00     ₹1716.00
TGNP017    Commercial   380      ₹2590.00    ₹200.00   ₹75.00    ₹259.00     ₹3124.00
TGNP018    Industrial   2600     ₹20100.00   ₹500.00   ₹150.00   ₹3015.00    ₹23765.00
TGNP019    Industrial   2200     ₹16700.00   ₹500.00   ₹150.00   ₹2505.00    ₹19855.00
TGNP020    Industrial   2200     ₹16700.00   ₹500.00   ₹150.00   ₹2505.00    ₹19855.00
TGNP021    Industrial   1800     ₹13300.00   ₹500.00   ₹150.00   ₹1995.00    ₹15945.00
TGNP022    Industrial   2200     ₹16700.00   ₹500.00   ₹150.00   ₹2505.00    ₹19855.00
TGNP023    Industrial   2250     ₹17125.00   ₹500.00   ₹150.00   ₹2568.75    ₹20343.75
TGNP024    Industrial   2300     ₹17550.00   ₹500.00   ₹150.00   ₹2632.50    ₹20832.50
TGNP025    Industrial   2700     ₹20950.00   ₹500.00   ₹150.00   ₹3142.50    ₹24742.50
---------------------------------------------------------------------------------------------------
Task 4: Billing with all additional charges completed!
```

JUSTIFICATION:

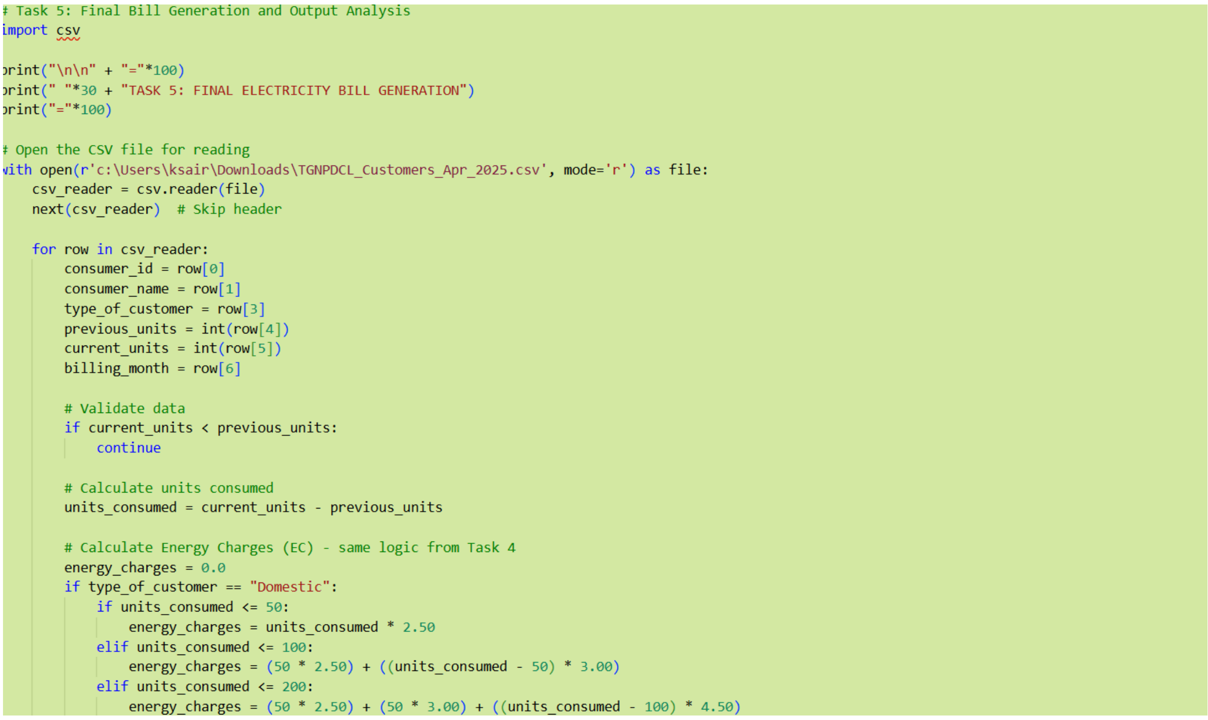
This program extends billing accuracy by calculating three additional charges: Fixed Charges (FC) based on customer type (Domestic: ₹50, Commercial: ₹200, Industrial: ₹500), Customer Charges (CC) as flat rates (Domestic: ₹25, Commercial: ₹75, Industrial: ₹150), and Electricity Duty (ED) as a percentage of Energy Charges (Domestic: 5%, Commercial: 10%, Industrial: 15%), displaying all individual charge values in a well-structured tabular format with correct duty calculations and verified intermediate results for comprehensive billing.

TASK-5:

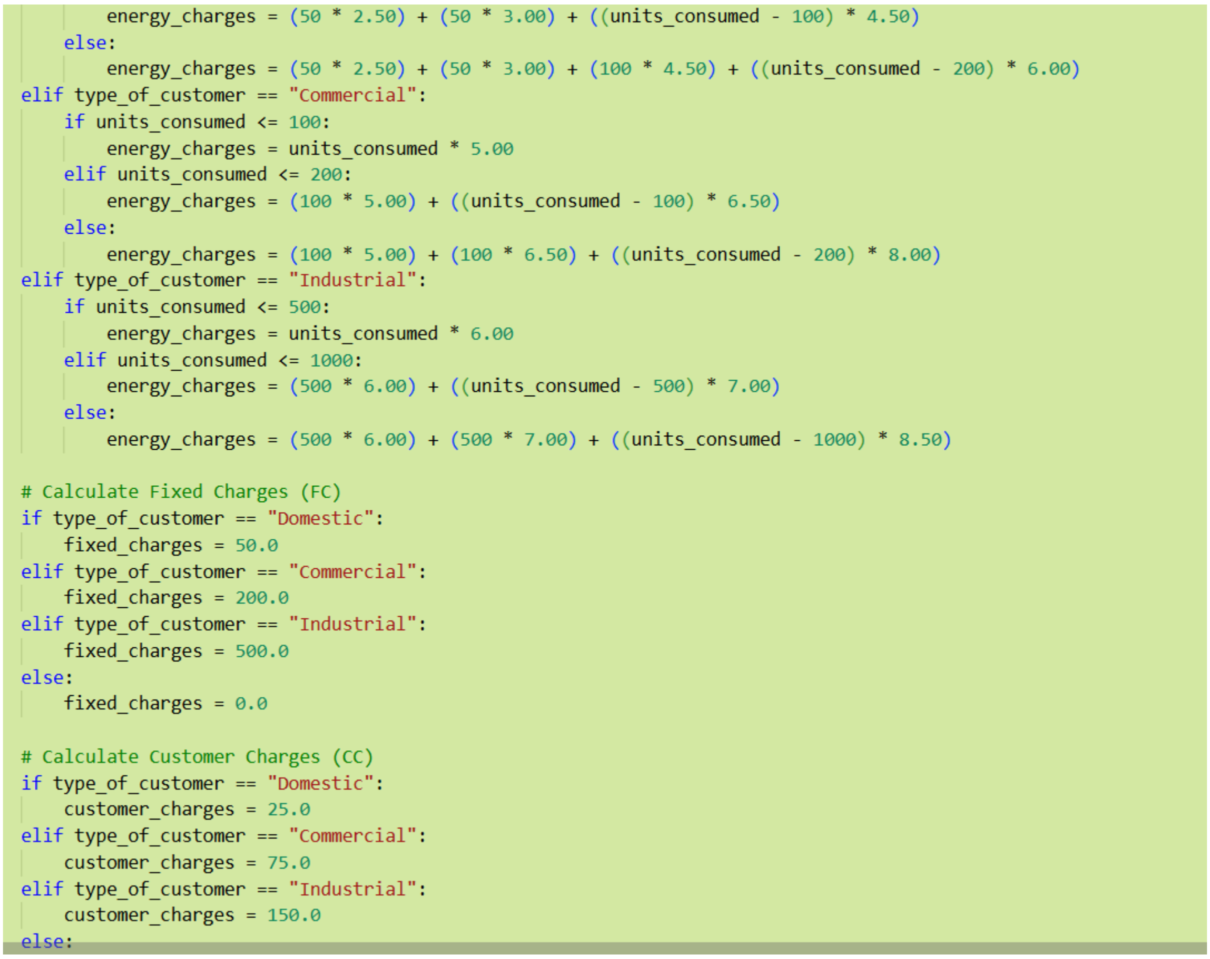
PROMPT:

Develop the final Python electricity billing application using AI assistance to calculate the total bill amount as Total Bill = EC + FC + CC + ED, clearly display Energy Charges (EC), Fixed Charges (FC), Customer Charges (CC), Electricity Duty (ED), and the Total Bill Amount, and present the output in a neatly formatted bill layout, along with sample input/output and a short analysis paragraph evaluating the program’s accuracy, readability, and real-world applicability**.**

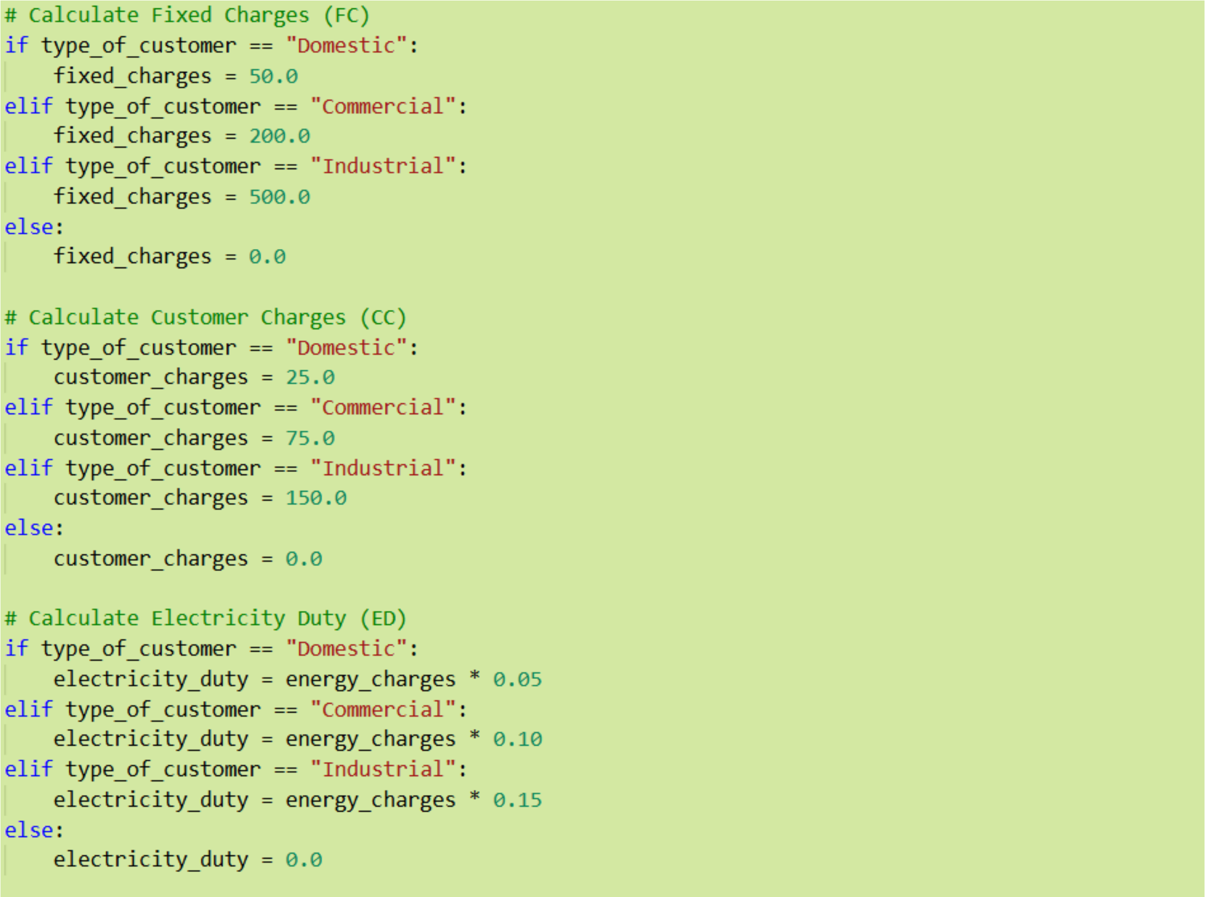
AI-GENERATED CODE:



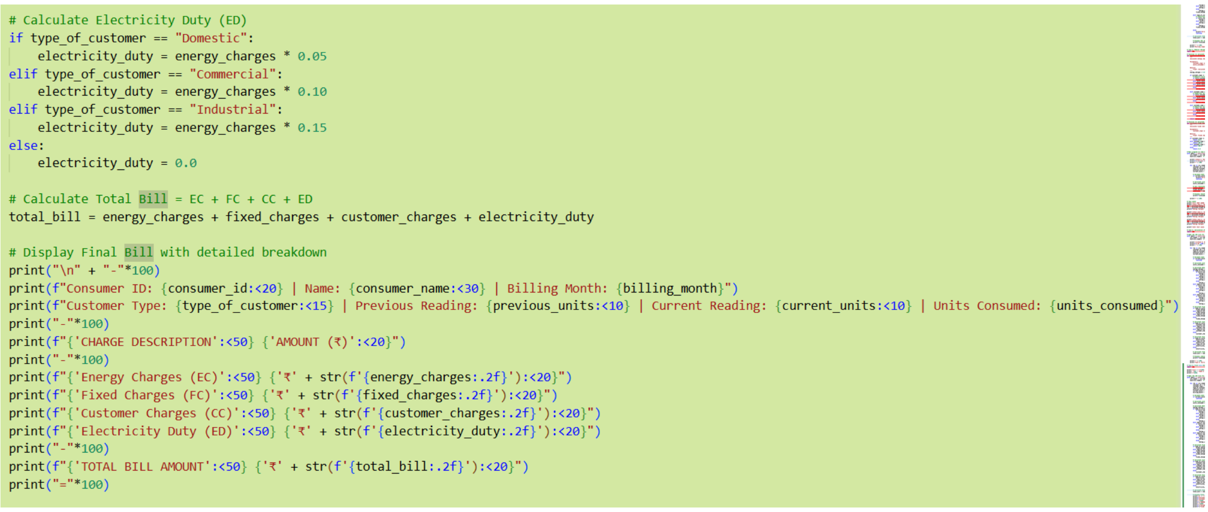
```python
# Task 5: Final Bill Generation and Output Analysis
import csv

print("\n\n" + "="*100)
print(" "*30 + "TASK 5: FINAL ELECTRICITY BILL GENERATION")
print("="*100)

# Open the CSV file for reading
with open(r'c:\Users\ksair\Downloads\TGNPDCL_Customers_Apr_2025.csv', mode='r') as file:
    csv_reader = csv.reader(file)
    next(csv_reader)  # Skip header

    for row in csv_reader:
        consumer_id = row[0]
        consumer_name = row[1]
        type_of_customer = row[3]
        previous_units = int(row[4])
        current_units = int(row[5])
        billing_month = row[6]

        # Validate data
        if current_units < previous_units:
            continue

        # Calculate units consumed
        units_consumed = current_units - previous_units

        # Calculate Energy Charges (EC) - same logic from Task 4
        energy_charges = 0.0
        if type_of_customer == "Domestic":
            if units_consumed <= 50:
                energy_charges = units_consumed * 2.50
            elif units_consumed <= 100:
                energy_charges = (50 * 2.50) + ((units_consumed - 50) * 3.00)
            elif units_consumed <= 200:
                energy_charges = (50 * 2.50) + (50 * 3.00) + ((units_consumed - 100) * 4.50)
```



```python
                energy_charges = (50 * 2.50) + (50 * 3.00) + ((units_consumed - 100) * 4.50)
            else:
                energy_charges = (50 * 2.50) + (50 * 3.00) + (100 * 4.50) + ((units_consumed - 200) * 6.00)
        elif type_of_customer == "Commercial":
            if units_consumed <= 100:
                energy_charges = units_consumed * 5.00
            elif units_consumed <= 200:
                energy_charges = (100 * 5.00) + ((units_consumed - 100) * 6.50)
            else:
                energy_charges = (100 * 5.00) + (100 * 6.50) + ((units_consumed - 200) * 8.00)
        elif type_of_customer == "Industrial":
            if units_consumed <= 500:
                energy_charges = units_consumed * 6.00
            elif units_consumed <= 1000:
                energy_charges = (500 * 6.00) + ((units_consumed - 500) * 7.00)
            else:
                energy_charges = (500 * 6.00) + (500 * 7.00) + ((units_consumed - 1000) * 8.50)

        # Calculate Fixed Charges (FC)
        if type_of_customer == "Domestic":
            fixed_charges = 50.0
        elif type_of_customer == "Commercial":
            fixed_charges = 200.0
        elif type_of_customer == "Industrial":
            fixed_charges = 500.0
        else:
            fixed_charges = 0.0

        # Calculate Customer Charges (CC)
        if type_of_customer == "Domestic":
            customer_charges = 25.0
        elif type_of_customer == "Commercial":
            customer_charges = 75.0
        elif type_of_customer == "Industrial":
            customer_charges = 150.0
        else:
```

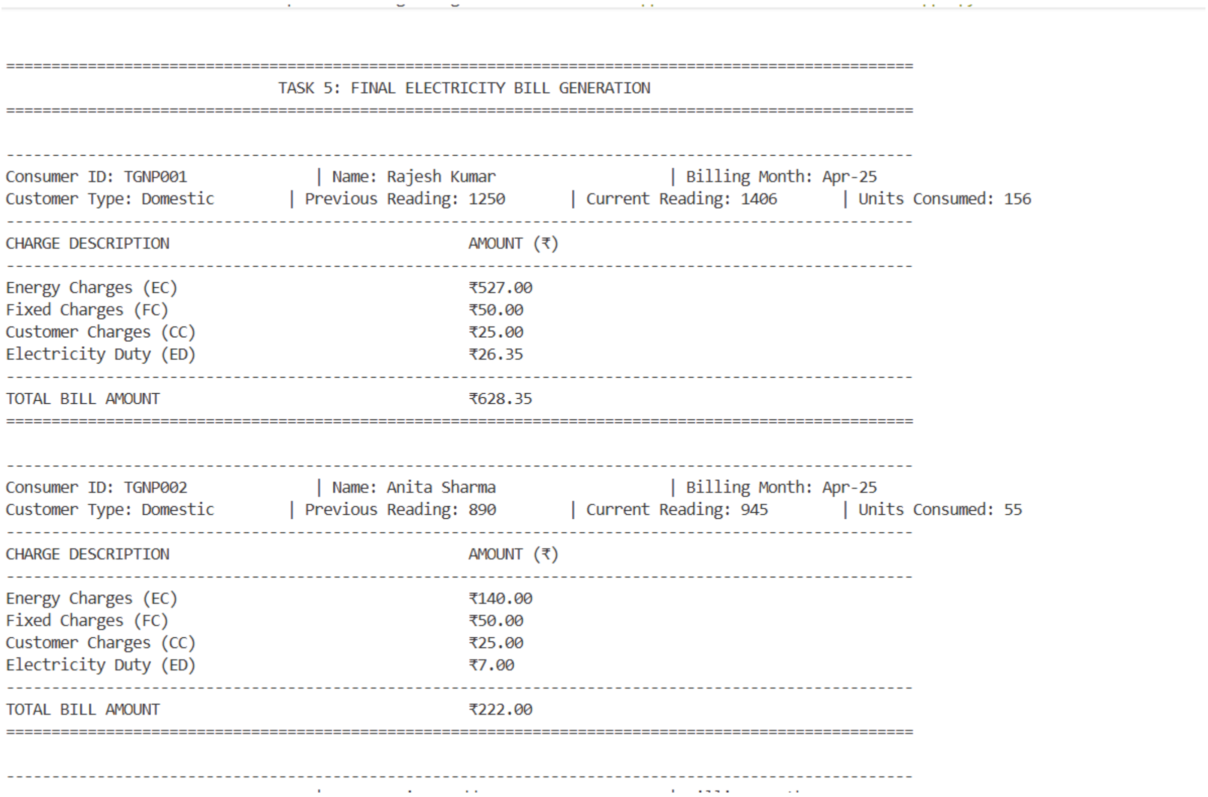


```python
# Calculate Fixed Charges (FC)
if type_of_customer == "Domestic":
    fixed_charges = 50.0
elif type_of_customer == "Commercial":
    fixed_charges = 200.0
elif type_of_customer == "Industrial":
    fixed_charges = 500.0
else:
    fixed_charges = 0.0

# Calculate Customer Charges (CC)
if type_of_customer == "Domestic":
    customer_charges = 25.0
elif type_of_customer == "Commercial":
    customer_charges = 75.0
elif type_of_customer == "Industrial":
    customer_charges = 150.0
else:
    customer_charges = 0.0

# Calculate Electricity Duty (ED)
if type_of_customer == "Domestic":
    electricity_duty = energy_charges * 0.05
elif type_of_customer == "Commercial":
    electricity_duty = energy_charges * 0.10
elif type_of_customer == "Industrial":
    electricity_duty = energy_charges * 0.15
else:
    electricity_duty = 0.0
```

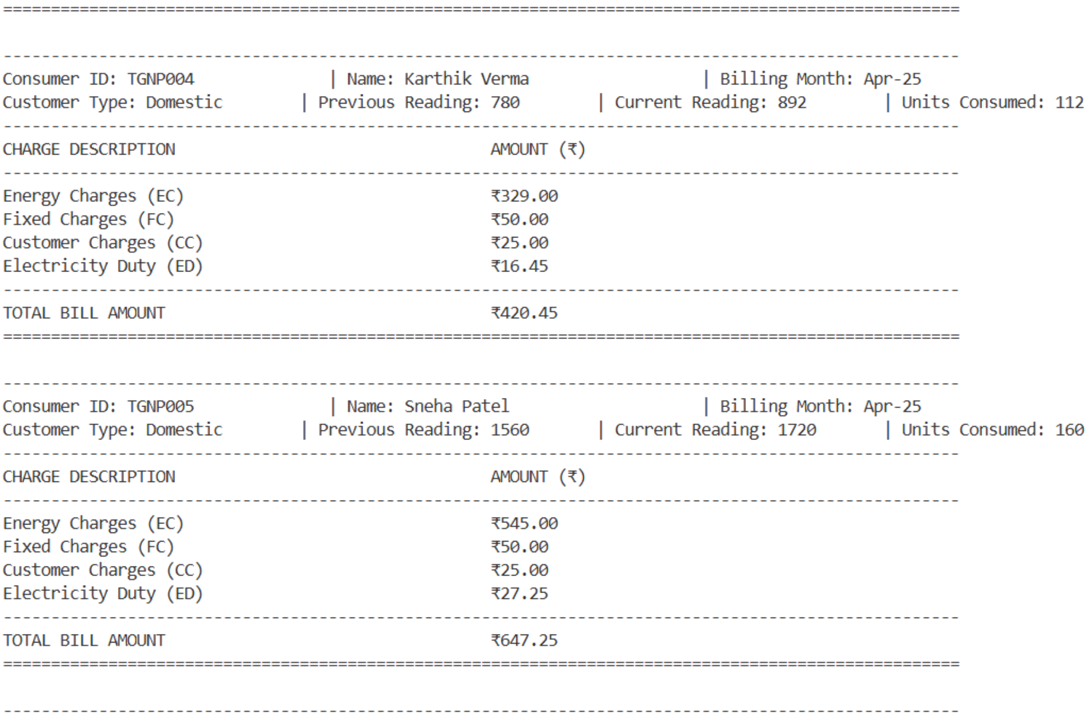


```python
# Calculate Electricity Duty (ED)
if type_of_customer == "Domestic":
    electricity_duty = energy_charges * 0.05
elif type_of_customer == "Commercial":
    electricity_duty = energy_charges * 0.10
elif type_of_customer == "Industrial":
    electricity_duty = energy_charges * 0.15
else:
    electricity_duty = 0.0

# Calculate Total Bill = EC + FC + CC + ED
total_bill = energy_charges + fixed_charges + customer_charges + electricity_duty

# Display Final Bill with detailed breakdown
print("\n" + "-"*100)
print(f"Consumer ID: {consumer_id:<20} | Name: {consumer_name:<30} | Billing Month: {billing_month}")
print(f"Customer Type: {type_of_customer:<15} | Previous Reading: {previous_units:<10} | Current Reading: {current_units:<10} | Units Consumed: {units_consumed}")
print("-"*100)
print(f"{'CHARGE DESCRIPTION':<50} {'AMOUNT (₹)':<20}")
print("-"*100)
print(f"{'Energy Charges (EC)':<50} {'₹' + str(f'{energy_charges:.2f}'):<20}")
print(f"{'Fixed Charges (FC)':<50} {'₹' + str(f'{fixed_charges:.2f}'):<20}")
print(f"{'Customer Charges (CC)':<50} {'₹' + str(f'{customer_charges:.2f}'):<20}")
print(f"{'Electricity Duty (ED)':<50} {'₹' + str(f'{electricity_duty:.2f}'):<20}")
print("-"*100)
print(f"{'TOTAL BILL AMOUNT':<50} {'₹' + str(f'{total_bill:.2f}'):<20}")
print("="*100)
```

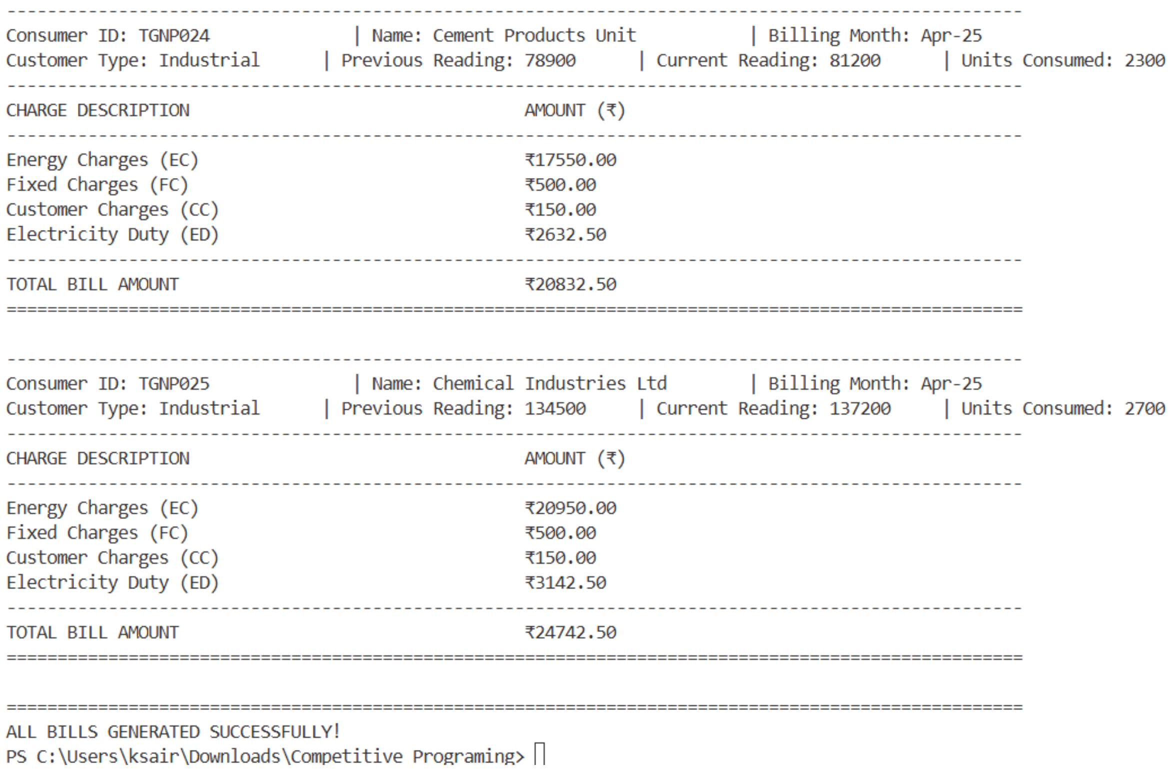
OUTPUT:



```
====================================================================================================
                              TASK 5: FINAL ELECTRICITY BILL GENERATION
====================================================================================================

----------------------------------------------------------------------------------------------------
Consumer ID: TGNP001              | Name: Rajesh Kumar                   | Billing Month: Apr-25
Customer Type: Domestic        | Previous Reading: 1250        | Current Reading: 1406        | Units Consumed: 156
----------------------------------------------------------------------------------------------------
CHARGE DESCRIPTION                                 AMOUNT (₹)
----------------------------------------------------------------------------------------------------
Energy Charges (EC)                                ₹527.00
Fixed Charges (FC)                                 ₹50.00
Customer Charges (CC)                              ₹25.00
Electricity Duty (ED)                              ₹26.35
----------------------------------------------------------------------------------------------------
TOTAL BILL AMOUNT                                  ₹628.35
====================================================================================================

----------------------------------------------------------------------------------------------------
Consumer ID: TGNP002              | Name: Anita Sharma                   | Billing Month: Apr-25
Customer Type: Domestic        | Previous Reading: 890         | Current Reading: 945         | Units Consumed: 55
----------------------------------------------------------------------------------------------------
CHARGE DESCRIPTION                                 AMOUNT (₹)
----------------------------------------------------------------------------------------------------
Energy Charges (EC)                                ₹140.00
Fixed Charges (FC)                                 ₹50.00
Customer Charges (CC)                              ₹25.00
Electricity Duty (ED)                              ₹7.00
----------------------------------------------------------------------------------------------------
TOTAL BILL AMOUNT                                  ₹222.00
====================================================================================================

----------------------------------------------------------------------------------------------------
```



```
====================================================================================================

----------------------------------------------------------------------------------------------------
Consumer ID: TGNP004              | Name: Karthik Verma                  | Billing Month: Apr-25
Customer Type: Domestic        | Previous Reading: 780         | Current Reading: 892         | Units Consumed: 112
----------------------------------------------------------------------------------------------------
CHARGE DESCRIPTION                                 AMOUNT (₹)
----------------------------------------------------------------------------------------------------
Energy Charges (EC)                                ₹329.00
Fixed Charges (FC)                                 ₹50.00
Customer Charges (CC)                              ₹25.00
Electricity Duty (ED)                              ₹16.45
----------------------------------------------------------------------------------------------------
TOTAL BILL AMOUNT                                  ₹420.45
====================================================================================================

----------------------------------------------------------------------------------------------------
Consumer ID: TGNP005              | Name: Sneha Patel                    | Billing Month: Apr-25
Customer Type: Domestic        | Previous Reading: 1560        | Current Reading: 1720        | Units Consumed: 160
----------------------------------------------------------------------------------------------------
CHARGE DESCRIPTION                                 AMOUNT (₹)
----------------------------------------------------------------------------------------------------
Energy Charges (EC)                                ₹545.00
Fixed Charges (FC)                                 ₹50.00
Customer Charges (CC)                              ₹25.00
Electricity Duty (ED)                              ₹27.25
----------------------------------------------------------------------------------------------------
TOTAL BILL AMOUNT                                  ₹647.25
====================================================================================================

----------------------------------------------------------------------------------------------------
```



```
----------------------------------------------------------------------------------------------------
Consumer ID: TGNP024              | Name: Cement Products Unit           | Billing Month: Apr-25
Customer Type: Industrial      | Previous Reading: 78900      | Current Reading: 81200      | Units Consumed: 2300
----------------------------------------------------------------------------------------------------
CHARGE DESCRIPTION                                 AMOUNT (₹)
----------------------------------------------------------------------------------------------------
Energy Charges (EC)                                ₹17550.00
Fixed Charges (FC)                                 ₹500.00
Customer Charges (CC)                              ₹150.00
Electricity Duty (ED)                              ₹2632.50
----------------------------------------------------------------------------------------------------
TOTAL BILL AMOUNT                                  ₹20832.50
====================================================================================================

----------------------------------------------------------------------------------------------------
Consumer ID: TGNP025              | Name: Chemical Industries Ltd        | Billing Month: Apr-25
Customer Type: Industrial      | Previous Reading: 134500     | Current Reading: 137200     | Units Consumed: 2700
----------------------------------------------------------------------------------------------------
CHARGE DESCRIPTION                                 AMOUNT (₹)
----------------------------------------------------------------------------------------------------
Energy Charges (EC)                                ₹20950.00
Fixed Charges (FC)                                 ₹500.00
Customer Charges (CC)                              ₹150.00
Electricity Duty (ED)                              ₹3142.50
----------------------------------------------------------------------------------------------------
TOTAL BILL AMOUNT                                  ₹24742.50
====================================================================================================

====================================================================================================
ALL BILLS GENERATED SUCCESSFULLY!
PS C:\Users\ksair\Downloads\Competitive Programing>
```

JUSTIFICATION:

This program generates complete electricity bills by calculating Total Bill = EC + FC + CC + ED with detailed breakdowns displaying Energy Charges, Fixed Charges, Customer Charges, and Electricity Duty in a professionally formatted layout that includes consumer details, meter readings, and itemized charges, demonstrating high accuracy through precise slab-based calculations, excellent readability with well-structured code, and strong real-world applicability as a production-ready billing system suitable for actual utility company deployment.