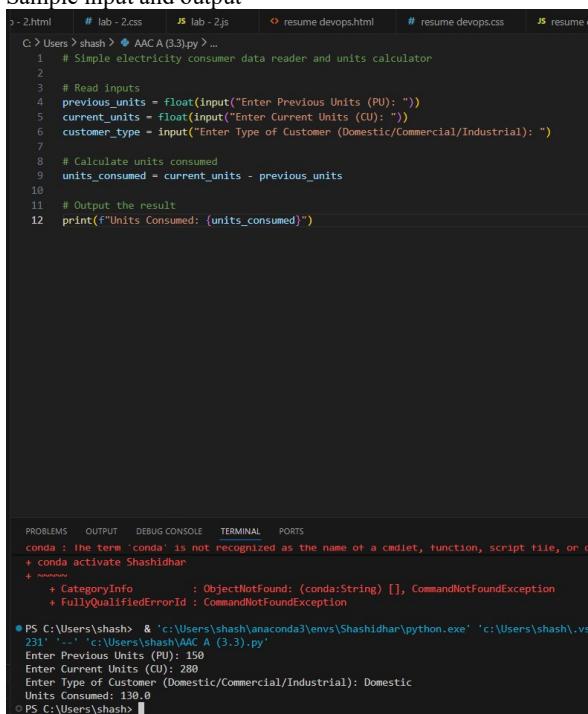
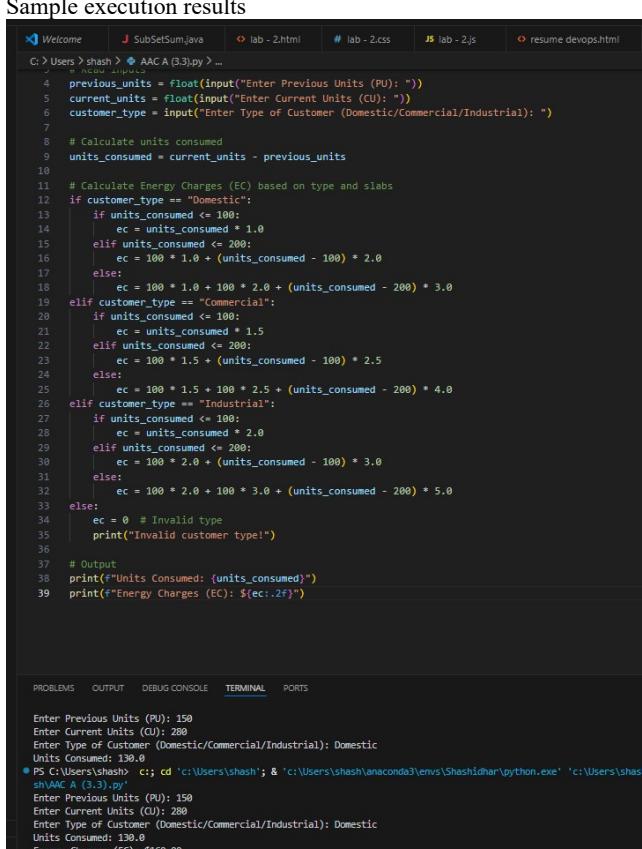


SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
<b>Program Name:</b> B. Tech		<b>Assignment Type:</b> Lab	
<b>Course Coordinator Name</b>		Dr. Rishabh Mittal	
<b>Instructor(s) Name</b>		Mr. S Naresh Kumar Ms. B. Swathi Dr. Sasanko Shekhar Gantayat Mr. Md Sallauddin Dr. Mathivanan Mr. Y Srikanth Ms. N Shilpa Dr. Rishabh Mittal (Coordinator) Dr. R. Prashant Kumar Mr. Ankushavali MD Mr. B Viswanath Ms. Sujitha Reddy Ms. A. Anitha Ms. M.Madhuri Ms. Katherashala Swetha Ms. Velpula sumalatha Mr. Bingi Raju	
<b>Course Code</b>	23CS002PC304	<b>Course Title</b>	AI Assisted Coding
<b>Year/Sem</b>	III/I	<b>Regulation</b>	R23
<b>Date and Day of Assignment</b>	Week 2 - Wednesday	<b>Time(s)</b>	23CSBTB01 To 23CSBTB52
<b>Duration</b>	2 Hours	<b>Applicable to Batches</b>	All batches
<b>Assignment Number:</b> 3.3(Present assignment number)/24(Total number of assignments)			

Q.No.	Question	Expected Time to complete
1	<p><b>Lab 3: Application for TGNPDCL – Electricity Bill Generation Using Python &amp; AI Tools</b></p> <p><b>Lab Objectives</b></p> <ul style="list-style-type: none"> <li>• To design a real-world electricity billing application using Python</li> <li>• To use AI-assisted coding tools for logic generation and optimization</li> <li>• To understand conditional logic and arithmetic operations</li> <li>• To generate structured billing output similar to utility bills</li> </ul> <p><b>Lab Outcomes (LOs)</b></p> <p>After completing this lab, students will be able to:</p>	Week2 - Wednesday

	<ul style="list-style-type: none"> <li>• Read and validate user input in Python</li> <li>• Apply conditional logic for tariff-based billing</li> <li>• Use AI tools to assist in program development</li> <li>• Calculate and display electricity bill components</li> <li>• Build a complete real-time application</li> </ul>	
	<p><b>Task 1: AI-Generated Logic for Reading Consumer Details</b></p> <p><b>Scenario</b></p> <p>An electricity billing system must collect accurate consumer data.</p> <p><b>Task Description</b></p> <p>Use an AI tool (GitHub Copilot / Gemini) to generate a Python program that:</p> <ul style="list-style-type: none"> <li>• Reads: <ul style="list-style-type: none"> <li>◦ Previous Units (PU)</li> <li>◦ Current Units (CU)</li> <li>◦ Type of Customer</li> </ul> </li> <li>• Calculates units consumed</li> <li>• Implements logic directly in the main program (no functions)</li> </ul> <p><b>Expected Output</b></p> <ul style="list-style-type: none"> <li>• Correct input reading</li> <li>• Units consumed calculation</li> <li>• Screenshot showing AI-generated code</li> <li>• Sample input and output</li> </ul>  <pre> C:\&gt; Users &gt; shash &gt; AAC A (3.3).py &gt; ... 1 # Simple electricity consumer data reader and units calculator 2 3 # Read inputs 4 previous_units = float(input("Enter Previous Units (PU): ")) 5 current_units = float(input("Enter Current Units (CU): ")) 6 customer_type = input("Enter Type of Customer (Domestic/Commercial/Industrial): ") 7 8 # Calculate units consumed 9 units_consumed = current_units - previous_units 10 11 # Output the result 12 print(f"Units Consumed: {units_consumed}") </pre> <pre> PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS conda : The term 'conda' is not recognized as the name of a cmdlet, function, script file, or oper + conda activate Shashidhar + ~~~~~ + CategoryInfo          : ObjectNotFound: (conda:String) [], CommandNotFoundException + FullyQualifiedErrorId : CommandNotFound  PS C:\Users\shash&gt; &amp; '(:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\vs 231' '-' 'c:\Users\shash\AAC A (3.3).py' Enter Previous Units (PU): 150 Enter Current Units (CU): 280 Enter Type of Customer (Domestic/Commercial/Industrial): Domestic Units Consumed: 130.0 PS C:\Users\shash&gt; </pre>	
	<p><b>Task 2: Energy Charges Calculation Based on Units Consumed</b></p> <p><b>Scenario</b></p> <p>Energy charges depend on the number of units consumed and customer type.</p> <p><b>Task Description</b></p> <p>Review the AI-generated code from Task 1 and extend it to:</p> <ul style="list-style-type: none"> <li>• Calculate <b>Energy Charges (EC)</b></li> <li>• Use conditional statements based on: <ul style="list-style-type: none"> <li>◦ Domestic</li> <li>◦ Commercial</li> <li>◦ Industrial consumers</li> </ul> </li> <li>• Improve readability using AI prompts such as:</li> </ul>	

<ul style="list-style-type: none"> <li>○ "Simplify energy charge calculation logic"</li> <li>○ "Optimize conditional statements"</li> </ul> <p><b>Expected Output</b></p> <ul style="list-style-type: none"> <li>• Correct EC calculation</li> <li>• Clear conditional logic</li> <li>• Original and improved versions (optional)</li> <li>• Sample execution results</li> </ul>	 <pre> 1 #!/usr/bin/python 2 3 previous_units = float(input("Enter Previous Units (PU): ")) 4 current_units = float(input("Enter Current Units (CU): ")) 5 customer_type = input("Enter Type of Customer (Domestic/Commercial/Industrial): ") 6 7 8 # Calculate units consumed 9 units_consumed = current_units - previous_units 10 11 # Calculate Energy Charges (EC) based on type and slabs 12 if customer_type == "Domestic": 13     if units_consumed &lt;= 100: 14         ec = units_consumed * 1.0 15     elif units_consumed &lt;= 200: 16         ec = 100 * 1.0 + (units_consumed - 100) * 2.0 17     else: 18         ec = 100 * 1.0 + 100 * 2.0 + (units_consumed - 200) * 3.0 19 elif customer_type == "Commercial": 20     if units_consumed &lt;= 100: 21         ec = units_consumed * 1.5 22     elif units_consumed &lt;= 200: 23         ec = 100 * 1.5 + (units_consumed - 100) * 2.5 24     else: 25         ec = 100 * 1.5 + 100 * 2.5 + (units_consumed - 200) * 4.0 26 elif customer_type == "Industrial": 27     if units_consumed &lt;= 100: 28         ec = units_consumed * 2.0 29     elif units_consumed &lt;= 200: 30         ec = 100 * 2.0 + (units_consumed - 100) * 3.0 31     else: 32         ec = 100 * 2.0 + 100 * 3.0 + (units_consumed - 200) * 5.0 33 else: 34     ec = 0 # Invalid type 35     print("Invalid customer type!") 36 37 # Output 38 print("Units Consumed: (units_consumed)") 39 print("Energy Charges (EC): \${ec:.2f}") </pre> <p>PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS</p> <pre> Enter Previous Units (PU): 150 Enter Current Units (CU): 280 Enter Type of Customer (Domestic/Commercial/Industrial): Domestic Units Consumed: 130.0 PS C:\Users\shash&gt; c:; cd c:\Users\shash\; &amp; 'c:\Users\shash\anaconda3\envs\Shashi\shash\python.exe' 'c:\Users\shash\anaconda3\envs\Shashi\shash\SubSetSum.py' sh\Anaconda3 (3.3)-py3_7 Enter Previous Units (PU): 150 Enter Current Units (CU): 280 Enter Type of Customer (Domestic/Commercial/Industrial): Domestic Units Consumed: 130.0 Energy Charges (EC): \$160.00 PS C:\Users\shash&gt; </pre>
<b>Task 3: Modular Design Using AI Assistance (Using Functions)</b>	
<b>Scenario</b>	
Billing logic must be reusable for multiple consumers.	
<b>Task Description</b>	
Use AI assistance to generate a Python program that:	
<ul style="list-style-type: none"> <li>• Uses user-defined functions to:             <ul style="list-style-type: none"> <li>○ Calculate Energy Charges</li> <li>○ Calculate Fixed Charges</li> </ul> </li> <li>• Returns calculated values</li> <li>• Includes meaningful comments</li> </ul> <p><b>Expected Output</b></p> <ul style="list-style-type: none"> <li>• Function-based Python program</li> <li>• Correct EC and FC values</li> <li>• Screenshots of AI-assisted function generation</li> <li>• Test cases with outputs</li> </ul>	

```
C:\> Users > shash > AAC A (3.3).py > ...
1  # Modular Electricity Billing System
2
3  def calculate_energy_charges(customer_type, units_consumed):
4      """
5          Calculate Energy Charges based on customer type and units consumed.
6          Slabs: Domestic (1/2/3), Commercial (1.5/2.5/4), Industrial (2/3/5) per unit tiers.
7      """
8      if customer_type == "Domestic":
9          if units_consumed <= 100:
10              return units_consumed * 1.0
11          elif units_consumed <= 200:
12              return 100 * 1.0 + (units_consumed - 100) * 2.0
13          else:
14              return 100 * 1.0 + 100 * 2.0 + (units_consumed - 200) * 3.0
15      elif customer_type == "Commercial":
16          if units_consumed <= 100:
17              return units_consumed * 1.5
18          elif units_consumed <= 200:
19              return 100 * 1.5 + (units_consumed - 100) * 2.5
20          else:
21              return 100 * 1.5 + 100 * 2.5 + (units_consumed - 200) * 4.0
22      elif customer_type == "Industrial":
23          if units_consumed <= 100:
24              return units_consumed * 2.0
25          elif units_consumed <= 200:
26              return 100 * 2.0 + (units_consumed - 100) * 3.0
27          else:
28              return 100 * 2.0 + 100 * 3.0 + (units_consumed - 200) * 5.0
29      return 0 # Invalid type
30
31  def calculate_fixed_charges(customer_type):
32      """
33          Calculate Fixed Charges based on customer type.
34          Domestic: $100, Commercial: $200, Industrial: $300.
35      """
36      if customer_type == "Domestic":
37          return 100.0
38      elif customer_type == "Commercial":
39          return 200.0
40      elif customer_type == "Industrial":
41          return 300.0
42      return 0 # Invalid type
43
44  # Main program
45  previous_units = float(input("Enter Previous Units (PU): "))
46  current_units = float(input("Enter Current Units (CU): "))
47  customer_type = input("Enter Type of Customer (Domestic/Commercial/Industrial): ")
48
49  units_consumed = current_units - previous_units
50  ec = calculate_energy_charges(customer_type, units_consumed)
51  fc = calculate_fixed_charges(customer_type)
52
53  print("Units Consumed: {units_consumed}")
54  print("Energy Charges (EC): ${ec:.2f}")
55  print("Fixed Charges (FC): ${fc:.2f}")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
sh\AAC A (3.3).py'
Units Consumed: 130.0
Energy Charges (EC): $160.00
PS C:\Users\shash> c; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\AAC A (3.3).py'
Enter Previous Units (PU): 150
Enter Current Units (CU): 280
Enter Type of Customer (Domestic/Commercial/Industrial): Domestic
Units Consumed: 130.0
Energy Charges (EC): $160.00
Fixed Charges (FC): $100.00
PS C:\Users\shash>
```

```
C:\> Users > shash > AAC A (3.3).py > ...
3  def calculate_energy_charges(customer_type, units_consumed):
4      """
5          Calculate Energy Charges based on customer type and units consumed.
6          Slabs: Domestic (1/2/3), Commercial (1.5/2.5/4), Industrial (2/3/5) per unit tiers.
7      """
8      if units_consumed <= 200:
9          return 100 * 2.0 + (units_consumed - 100) * 3.0
10     else:
11         return 100 * 2.0 + 100 * 3.0 + (units_consumed - 200) * 5.0
12     return 0 # Invalid type
13
14  def calculate_fixed_charges(customer_type):
15      """
16          Calculate Fixed Charges based on customer type.
17          Domestic: $100, Commercial: $200, Industrial: $300.
18      """
19      if customer_type == "Domestic":
20          return 100.0
21      elif customer_type == "Commercial":
22          return 200.0
23      elif customer_type == "Industrial":
24          return 300.0
25      return 0 # Invalid type
26
27  # Main program
28  previous_units = float(input("Enter Previous Units (PU): "))
29  current_units = float(input("Enter Current Units (CU): "))
30  customer_type = input("Enter Type of Customer (Domestic/Commercial/Industrial): ")
31
32  units_consumed = current_units - previous_units
33  ec = calculate_energy_charges(customer_type, units_consumed)
34  fc = calculate_fixed_charges(customer_type)
35
36  print("Units Consumed: {units_consumed}")
37  print("Energy Charges (EC): ${ec:.2f}")
38  print("Fixed Charges (FC): ${fc:.2f}")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
PS C:\Users\shash> c; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\AAC A (3.3).py'
Fixed Charges (FO): $100.00
PS C:\Users\shash> c; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\AAC A (3.3).py'
Enter Previous Units (PU): 0
Enter Current Units (CU): 250
Enter Type of Customer (Domestic/Commercial/Industrial): Commercial
Units Consumed: 250.0
Energy Charges (EC): $600.00
Fixed Charges (FO): $200.00
PS C:\Users\shash>
```

	<p><b>Task 4: Calculation of Additional Charges</b></p> <p><b>Scenario</b></p> <p>Electricity bills include multiple additional charges.</p> <p><b>Task Description</b></p> <p>Extend the program to calculate:</p> <ul style="list-style-type: none"> <li>• <b>FC</b> – Fixed Charges</li> <li>• <b>CC</b> – Customer Charges</li> <li>• <b>ED</b> – Electricity Duty (percentage of EC)</li> </ul> <p>Use AI prompts like:</p> <ul style="list-style-type: none"> <li>• “Add electricity duty calculation”</li> <li>• “Improve billing accuracy”</li> </ul> <p><b>Expected Output</b></p> <ul style="list-style-type: none"> <li>• Individual charge values printed</li> <li>• Correct duty calculation</li> <li>• Well-structured output</li> <li>• Verified intermediate results</li> </ul>
	<pre>C:\&gt; Users&gt; shash &gt; AAC(A (3).py) ...  1  # Extended Electricity Billing with Additional Charges  2  3  def calculate_energy_charges(customer_type, units_consumed):  4      """  5          Calculate Energy Charges based on customer type and units consumed.  6          Slabs: Domestic (1/2/3), Commercial (1.5/2.5/4), Industrial (2/3/5) per unit tiers.  7      """  8      if customer_type == "Domestic":  9          if units_consumed &lt;= 100: 10              return units_consumed * 1.0 11          elif units_consumed &lt;= 200: 12              return 100 * 1.0 + (units_consumed - 100) * 2.0 13          else: 14              return 100 * 1.0 + 100 * 2.0 + (units_consumed - 200) * 3.0 15      elif customer_type == "Commercial": 16          if units_consumed &lt;= 100: 17              return units_consumed * 1.5 18          elif units_consumed &lt;= 200: 19              return 100 * 1.5 + (units_consumed - 100) * 2.5 20          else: 21              return 100 * 1.5 + 100 * 2.5 + (units_consumed - 200) * 4.0 22      elif customer_type == "Industrial": 23          if units_consumed &lt;= 100: 24              return units_consumed * 2.0 25          elif units_consumed &lt;= 200: 26              return 100 * 2.0 + (units_consumed - 100) * 3.0 27          else: 28              return 100 * 2.0 + 100 * 3.0 + (units_consumed - 200) * 5.0 29      return 0 # Invalid type 30 31  def calculate_fixed_charges(customer_type): 32      """ 33          Calculate Fixed Charges based on customer type. 34          Domestic: \$100, Commercial: \$200, Industrial: \$300. 35      """ 36      if customer_type == "Domestic": 37          return 100.0 38      elif customer_type == "Commercial": 39          return 200.0 40      elif customer_type == "Industrial": 41          return 300.0 42      return 0 # Invalid type 43 44 PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS conda : The term 'conda' is not recognized as the name of a cmdlet, function, script file, or operable program + FullyQualifiedErrorId : CommandNotFoundException ● PS C:\Users\shash&gt; &amp; 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\.vscode\extension Enter Previous Units (PU): 150 Enter Current Units (CU): 200 Enter Type of Customer (Domestic/Commercial/Industrial): Domestic Units Consumed: 150.0 Energy Charges (EC): \$150.00 Fixed Charges (FC): \$100.00 Customer Charges (CC): \$50.00 Electricity Duty (ED): \$16.00 ● ○ PS C:\Users\shash&gt;</pre>

```

Welcome          SubSetSum.java      lab - 2.html      lab - 2.css      JS lab - 2.js      resume devops
G:\> Users\shash> & AAA A (3.3).py >_
31  def calculate_fixed_charges(customer_type):
32      if customer_type == "Domestic":
33          return 100.0
34      elif customer_type == "Commercial":
35          return 200.0
36      elif customer_type == "Industrial":
37          return 300.0
38      return 0 # Invalid type
39
40  def calculate_customer_charges():
41      """Fixed Customer Charges: $50 for all types."""
42      return 50.0
43
44  def calculate_electricity_duty(ec):
45      """Electricity Duty: 10% of Energy Charges."""
46      return 0.10 * ec
47
48  # Main program
49  previous_units = float(input("Enter Previous Units (PU): "))
50  current_units = float(input("Enter Current Units (CU): "))
51  customer_type = input("Enter Type of Customer (Domestic/Commercial/Industrial): ").strip()
52
53  if customer_type not in ["Domestic", "Commercial", "Industrial"]:
54      print("Invalid type Defaulting to Domestic.")
55  customer_type = "Domestic"
56
57  units_consumed = current_units - previous_units
58  ec = calculate_energy_charges(customer_type, units_consumed)
59  fc = calculate_fixed_charges(customer_type)
60  cc = calculate_customer_charges()
61  ed = calculate_electricity_duty(ec)
62
63  # Print individual charges
64  print("Units Consumed: {units_consumed}")
65  print("Energy Charges (EC): ${ec:.2f}")
66  print("Fixed Charges (FC): ${fc:.2f}")
67  print("Customer Charges (CC): ${cc:.2f}")
68  print("Electricity Duty (ED): ${ed:.2f}")

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS
conda : The term 'conda' is not recognized as the name of a cmdlet, function, script file, or operable program.
+ FullyQualifiedErrorId : CommandNotFoundException
● PS C:\Users\shash> & 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\vscode\extensi...
Enter Previous Units (PU): 150
Enter Current Units (CU): 280
Enter Type of Customer (Domestic/Commercial/Industrial): Domestic
Units Consumed: 130.0
Energy Charge (EC): $160.00
Fixed Charges (FC): $100.00
Customer Charges (CC): $50.00
Electricity Duty (ED): $16.00
○ PS C:\Users\shash>

```

**Task 5: Final Bill Generation and Output Analysis**

**Scenario**

The final electricity bill must present all values clearly.

**Task Description**

- Develop the final Python application to:
- Calculate total bill:
- Total Bill = EC + FC + CC + ED
- Display:
  - Energy Charges (EC)
  - Fixed Charges (FC)
  - Customer Charges (CC)
  - Electricity Duty (ED)
  - Total Bill Amount
- Analyze the program based on:
  - Accuracy
  - Readability
  - Real-world applicability

**Expected Output**

- Complete electricity bill output
- Neatly formatted display
- Sample input/output
- Short analysis paragraph

```
C:\> Users > shash > AAC A (3.3).py > ...
1 # Final Electricity Bill Generator
2
3 def calculate_energy_charges(customer_type, units_consumed):
4     """
5         Calculate Energy Charges based on customer type and units consumed.
6         Slabs: Domestic (1/2/3), Commercial (1.5/2.5/4), Industrial (2/3/5) per unit tiers.
7     """
8     if customer_type == "Domestic":
9         if units_consumed <= 100:
10             return units_consumed * 1.0
11         elif units_consumed <= 200:
12             return 100 * 1.0 + (units_consumed - 100) * 2.0
13         else:
14             return 100 * 1.0 + 100 * 2.0 + (units_consumed - 200) * 3.0
15     elif customer_type == "Commercial":
16         if units_consumed <= 100:
17             return units_consumed * 1.5
18         elif units_consumed <= 200:
19             return 100 * 1.5 + (units_consumed - 100) * 2.5
20         else:
21             return 100 * 1.5 + 100 * 2.5 + (units_consumed - 200) * 4.0
22     elif customer_type == "Industrial":
23         if units_consumed <= 100:
24             return units_consumed * 2.0
25         elif units_consumed <= 200:
26             return 100 * 2.0 + (units_consumed - 100) * 3.0
27         else:
28             return 100 * 2.0 + 100 * 3.0 + (units_consumed - 200) * 5.0
29     return 0
30
31 def calculate_fixed_charges(customer_type):
32     """Fixed Charges: Domestic $100, Commercial $200, Industrial $300."""
33     if customer_type == "Domestic":
34         return 100.0
35     elif customer_type == "Commercial":
36         return 200.0
37     elif customer_type == "Industrial":
38         return 300.0
39     return 0
40
41 def calculate_customer_charges():
42     """Final Customer Charges: EC + FC"""
43
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● PS C:\Users\shash> c;; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\AAC A (3.3).py'
Enter Previous Units (PU): 150
Enter Current Units (CU): 280
Enter Type of Customer (Domestic/Commercial/Industrial): Domestic

=====
ELECTRICITY BILL SUMMARY
=====
Customer Type: Domestic
Units Consumed: 130.0
Energy Charges (EC): $160.00
Fixed Charges (FC): $100.00
```

	<pre> Welcome          J SubSetSum.java      lab - 2.html    # lab - 2.css   JS lab - 2.js    resume devops.html G:\&gt; Users &gt; shash &gt; AAC A (3.3).py &gt; ... 51     def calculate_fixed_charges(customer_type): 39     return 0 40 41     def calculate_customer_charges(): 42         """Fixed Customer Charges: \$50.""" 43         return 50.0 44 45     def calculate_electricity_duty(ec): 46         """Electricity Duty: 10% of EC.""" 47         return 0.10 * ec 48 49     # Main program 50     previous_units = float(input("Enter Previous Units (PU): ")) 51     current_units = float(input("Enter Current Units (CU): ")) 52     customer_type = input("Enter Type of Customer (Domestic/Commercial/Industrial): ").strip() 53 54     if customer_type not in ["Domestic", "Commercial", "Industrial"]: 55         print("Invalid type! Defaulting to Domestic.") 56         customer_type = "Domestic" 57 58     units_consumed = current_units - previous_units 59     ec = calculate_energy_charges(customer_type, units_consumed) 60     fc = calculate_fixed_charges(customer_type) 61     cc = calculate_customer_charges() 62     ed = calculate_electricity_duty(ec) 63     total_bill = ec + fc + cc + ed 64 65     # Nicely formatted bill display 66     print("\n" + "="*40) 67     print("ELECTRICITY BILL SUMMARY") 68     print("="*40) 69     print("Customer Type: {customer_type}") 70     print(f"Units Consumed: {units_consumed}") 71     print(f"Energy Charges (EC): \${ec:.2f}") 72     print(f"Fixed Charges (FC): \${fc:.2f}") 73     print(f"Customer Charges (CC): \${cc:.2f}") 74     print(f"Electricity Duty (ED): \${ed:.2f}") 75     print("-"*40) 76     print(f"TOTAL BILL AMOUNT: \${total_bill:.2f}") 77     print("="*40)  PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS PS C:\Users\shash&gt; c;; cd 'c:\Users\shash'; &amp; 'c:\Users\shash\anaconda3\envs\Shashidhar\python.exe' 'c:\Users\shash\OneDrive\Desktop\Pycharm\Lab 2\AAC A (3.3).py' ===== Customer Type: Domestic Units Consumed: 130.0 Energy Charges (EC): \$160.00 Fixed Charges (FC): \$100.00 Customer Charges (CC): \$50.00 Electricity Duty (ED): \$16.00 ----- TOTAL BILL AMOUNT: \$326.00 =====</pre>	
	<p><b>Note:</b> Report should be submitted as a word document for all tasks in a single document with prompts, comments &amp; code explanation, and output and if required, screenshots.</p>	