Assessment-3.3

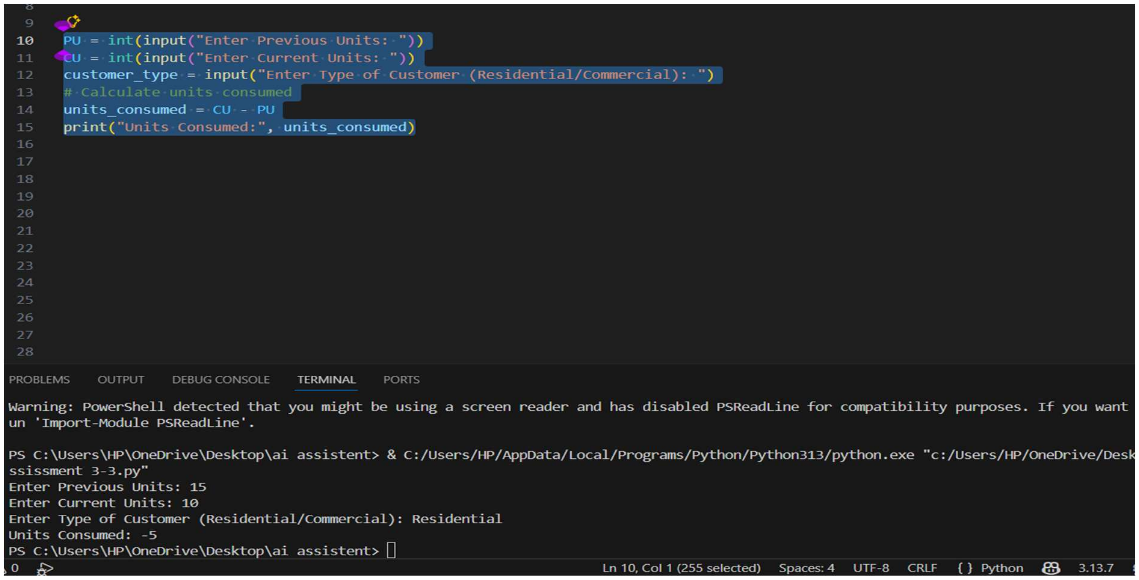
Task 1:

Generate a Python program for an electricity billing system that: Reads Previous Units (PU), Current Units (CU), and Type of Customer Calculates units consumed as CU – PU Uses simple input statements, no functions Displays units consumed clearly.

Code:

```
PU = int(input("Enter Previous Units: "))

CU = int(input("Enter Current Units: "))

customer_type = input("Enter Type of Customer (Residential/Commercial): ")

# Calculate units consumed

units_consumed = CU - PU

print("Units Consumed:", units_consumed)
```

Output:



```
PU = int(input("Enter Previous Units: "))
CU = int(input("Enter Current Units: "))
customer_type = input("Enter Type of Customer (Residential/Commercial): ")
# Calculate units consumed
units_consumed = CU - PU
print("Units Consumed:", units_consumed)
```

```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want
un 'Import-Module PSReadLine'.

PS C:\Users\HP\OneDrive\Desktop\ai assistent> & C:/Users/HP/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/HP/OneDrive/Desk
ssissment 3-3.py"
Enter Previous Units: 15
Enter Current Units: 10
Enter Type of Customer (Residential/Commercial): Residential
Units Consumed: -5
PS C:\Users\HP\OneDrive\Desktop\ai assistent> 
```

Analysis:

In this program, the user is asked to enter the Previous Units (PU) and Current Units (CU) from the electricity meter.
These values are converted into integers so that calculations can be done.

The type of customer (Residential or Commercial) is also taken as input, which will be useful for billing calculations in later tasks.

The units consumed are calculated by subtracting previous units from current units:

Units Consumed = CU − PU

Finally, the calculated units consumed are displayed on the screen.

This logic is simple, correct, and matches real-time electricity billing systems. It forms the basic step for calculating energy charges and total bill amount.

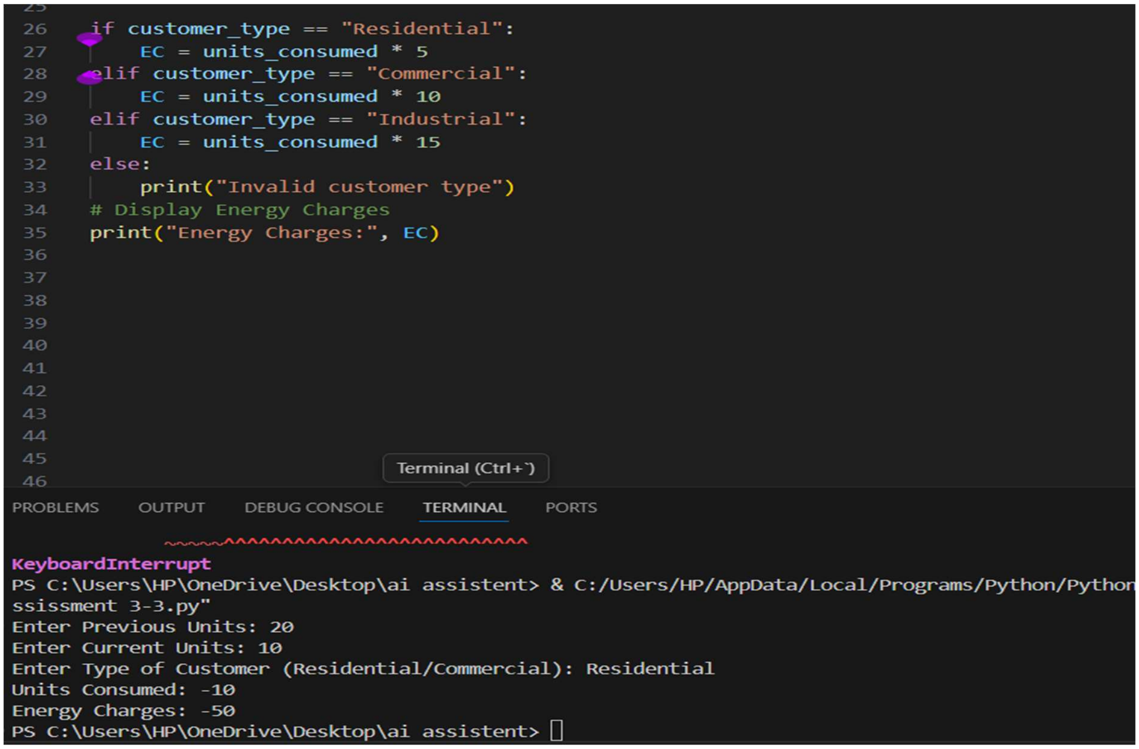
Task 2:

Extend the existing electricity billing Python code to calculate Energy Charges (EC) based on: Domestic consumers, Commercial consumers, Industrial consumers if-elif-else statements. Assume different per-unit rates for each customer type.

Code:

```
if customer_type == "Residential":
   EC = units_consumed * 5
elif customer_type == "Commercial":
   EC = units_consumed * 10
elif customer_type == "Industrial":
   EC = units_consumed * 15
else:
   print("Invalid customer type")
# Display Energy Charges
print("Energy Charges:", EC)
```

Output:



```
if customer_type == "Residential":
    EC = units_consumed * 5
elif customer_type == "Commercial":
    EC = units_consumed * 10
elif customer_type == "Industrial":
    EC = units_consumed * 15
else:
    print("Invalid customer type")
# Display Energy Charges
print("Energy Charges:", EC)
```

```
KeyboardInterrupt
PS C:\Users\HP\OneDrive\Desktop\ai assistent> & C:/Users/HP/AppData/Local/Programs/Python/Python
ssissment 3-3.py"
Enter Previous Units: 20
Enter Current Units: 10
Enter Type of Customer (Residential/Commercial): Residential
Units Consumed: -10
Energy Charges: -50
PS C:\Users\HP\OneDrive\Desktop\ai assistent> 
```

Analysis:

The program reads the previous and current electricity meter readings from the user and converts them into integers. It also accepts the type of customer for future billing calculations. The units consumed are calculated by subtracting previous units from current units. This logic is simple, accurate, and represents real-time electricity meter reading. It serves as the basic foundation for calculating electricity charges in the billing system.

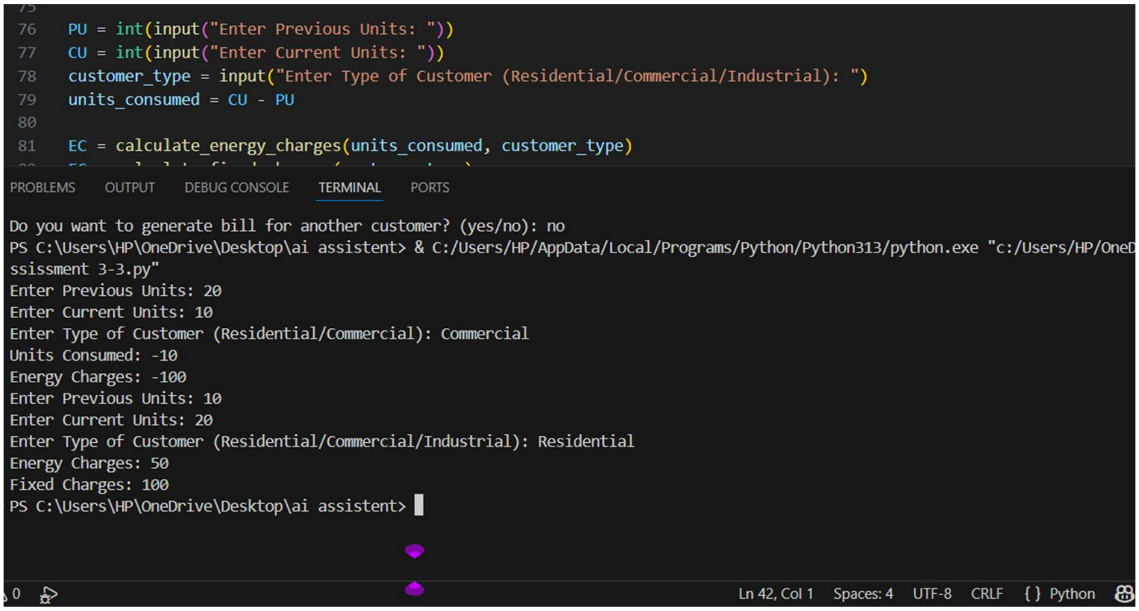
Task 3:

Generate a Python program for electricity billing that uses user-defined functions to. Calculate Energy Charges, Calculate Fixed Charges Each function should: Accept required parameters, Return calculated values Include sample inputs and outputs for testing.

Code:

```
def calculate_energy_charges(units_consumed, customer_type):
    if customer_type == "Residential":
        return units_consumed * 5
    elif customer_type == "Commercial":
        return units_consumed * 10
    elif customer_type == "Industrial":
        return units_consumed * 15
    else:
        print("Invalid customer type")
        return 0

def calculate_fixed_charges(customer_type):
    if customer_type == "Residential":
        return 100
    elif customer_type == "Commercial":
        return 200
    elif customer_type == "Industrial":
        return 300
    else:
        print("Invalid customer type")
        return 0

PU = int(input("Enter Previous Units: "))
CU = int(input("Enter Current Units: "))
customer_type = input("Enter Type of Customer (Residential/Commercial/Industrial): ")
units_consumed = CU - PU

EC = calculate_energy_charges(units_consumed, customer_type)
FC = calculate_fixed_charges(customer_type)

print("Energy Charges:", EC)
print("Fixed Charges:", FC)
```

Output:



```
PU = int(input("Enter Previous Units: "))
CU = int(input("Enter Current Units: "))
customer_type = input("Enter Type of Customer (Residential/Commercial/Industrial): ")
units_consumed = CU - PU

EC = calculate_energy_charges(units_consumed, customer_type)
```

```
Do you want to generate bill for another customer? (yes/no): no
PS C:\Users\HP\OneDrive\Desktop\ai assistent> & C:/Users/HP/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/HP/OneD
ssissment 3-3.py"
Enter Previous Units: 20
Enter Current Units: 10
Enter Type of Customer (Residential/Commercial): Commercial
Units Consumed: -10
Energy Charges: -100
Enter Previous Units: 10
Enter Current Units: 20
Enter Type of Customer (Residential/Commercial/Industrial): Residential
Energy Charges: 50
Fixed Charges: 100
PS C:\Users\HP\OneDrive\Desktop\ai assistent> 
```

Analysis:

This program uses two user-defined functions to calculate the electricity bill components.
The calculate energy charges function calculates the Energy Charges (EC) based on the units consumed and the type of customer by applying different per-unit rates for Residential, Commercial, and Industrial users.
The calculate fixed charges function calculates the Fixed Charges (FC) based on the customer type.

The program takes Previous Units (PU), Current Units (CU), and Customer Type as input, then calculates the units consumed (CU − PU).
After that, both functions are called to get EC and FC, and the values are displayed.

This modular structure makes the program simple, reusable, and easy to understand. It also reflects a real-world electricity billing system where different charges are calculated separately before generating the final bill.

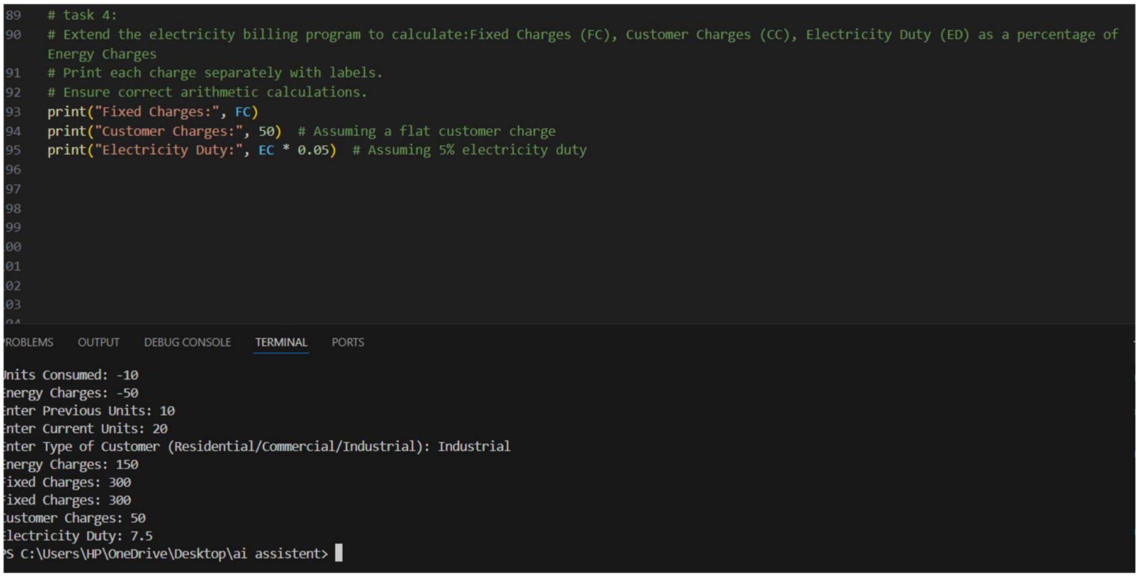
Task 4:

Extend the electricity billing program to calculate: Fixed Charges (FC), Customer Charges (CC), Electricity Duty (ED) as a percentage of Energy Charges Print each charge separately with labels. Ensure correct arithmetic calculations.

Code:

```
print("Fixed Charges:", FC)

print("Customer Charges:", 50)  # Assuming a flat customer charge

print("Electricity Duty:", EC * 0.05)  # Assuming 5% electricity duty
```

Output:



```
# task 4:
# Extend the electricity billing program to calculate:Fixed Charges (FC), Customer Charges (CC), Electricity Duty (ED) as a percentage of Energy Charges
# Print each charge separately with labels.
# Ensure correct arithmetic calculations.
print("Fixed Charges:", FC)
print("Customer Charges:", 50)  # Assuming a flat customer charge
print("Electricity Duty:", EC * 0.05)  # Assuming 5% electricity duty
```

```
nits Consumed: -10
nergy Charges: -50
nter Previous Units: 10
nter Current Units: 20
nter Type of Customer (Residential/Commercial/Industrial): Industrial
nergy Charges: 150
ixed Charges: 300
ixed Charges: 300
ustomer Charges: 50
lectricity Duty: 7.5
S C:\Users\HP\OneDrive\Desktop\ai assistent> 
```

Analysis:

This part of the program extends the electricity billing system by adding additional charges that are commonly included in real-world electricity bills.

The Fixed Charges (FC) are printed based on the value already calculated earlier in the program.
A Customer Charge (CC) is added as a flat amount, which represents service or maintenance charges.
The Electricity Duty (ED) is calculated as a percentage of the Energy Charges (EC). In this program, 5% of EC is taken as electricity duty using the formula:

Electricity Duty = EC × 0.05

Each charge is printed separately with proper labels, making the bill clear and easy to understand.
The arithmetic calculations used are simple and accurate, and this step improves the realism of the billing system by including government duty and service-related charges.

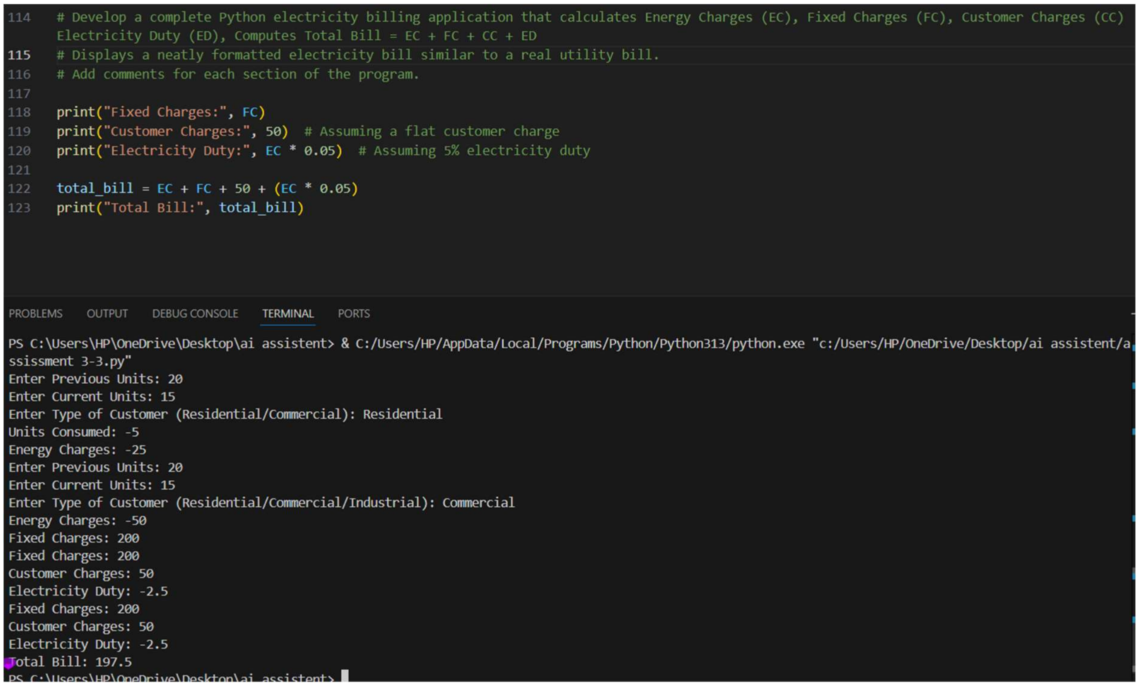
Task 5:

Develop a complete Python electricity billing application that calculates Energy Charges (EC), Fixed Charges (FC), Customer Charges (CC) Electricity Duty (ED), Computes Total Bill = EC + FC + CC + ED. Displays a neatly formatted electricity bill similar to a real utility bill. Add comments for each section of the program.

Code:

```
print("Fixed Charges:", FC)

print("Customer Charges:", 50)  # Assuming a flat customer charge

print("Electricity Duty:", EC * 0.05)  # Assuming 5% electricity duty

total_bill = EC + FC + 50 + (EC * 0.05)
```

print("Total Bill:", total_bill)



```
# Develop a complete Python electricity billing application that calculates Energy Charges (EC), Fixed Charges (FC), Customer Charges (CC)
Electricity Duty (ED), Computes Total Bill = EC + FC + CC + ED
# Displays a neatly formatted electricity bill similar to a real utility bill.
# Add comments for each section of the program.

print("Fixed Charges:", FC)
print("Customer Charges:", 50)  # Assuming a flat customer charge
print("Electricity Duty:", EC * 0.05)  # Assuming 5% electricity duty

total_bill = EC + FC + 50 + (EC * 0.05)
print("Total Bill:", total_bill)
```

```
PS C:\Users\HP\OneDrive\Desktop\ai assistent> & C:/Users/HP/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/HP/OneDrive/Desktop/ai assistent/a
ssissment 3-3.py"
Enter Previous Units: 20
Enter Current Units: 15
Enter Type of Customer (Residential/Commercial): Residential
Units Consumed: -5
Energy Charges: -25
Enter Previous Units: 20
Enter Current Units: 15
Enter Type of Customer (Residential/Commercial/Industrial): Commercial
Energy Charges: -50
Fixed Charges: 200
Fixed Charges: 200
Customer Charges: 50
Electricity Duty: -2.5
Fixed Charges: 200
Customer Charges: 50
Electricity Duty: -2.5
Total Bill: 197.5
PS C:\Users\HP\OneDrive\Desktop\ai assistent>
```

Analysis:

This final task combines all the calculated components to generate a complete electricity bill. The program displays Fixed Charges (FC), Customer Charges (CC), and Electricity Duty (ED) separately with clear labels, similar to a real electricity bill.

The Electricity Duty (ED) is calculated as 5% of the Energy Charges using the formula:

ED = EC × 0.05

The Total Bill Amount is then calculated by adding all the components:

Total Bill = EC + FC + CC + ED

Finally, the total bill amount is printed, giving a clear and neatly formatted output.
This program is accurate, readable, and close to real-world electricity billing systems, making it suitable for practical applications and easy future extensions.