



Electricity Usage Monitor (C++)

By Piyush Patil

Project Objective



Automate Calculation

Eliminate manual computations and streamline the billing process through automated logic implementation



Reduce Errors

Minimize human calculation mistakes and ensure consistent, reliable bill generation



Fast Results

Deliver accurate billing calculations instantly based on consumption units

Core Technologies & Implementation

Programming Foundation

The system leverages fundamental C++ programming concepts including object-oriented principles, data structures, and algorithmic logic to create an efficient billing solution.

Key Components

- Loop structures for iteration
- Functions for modularity
- Conditional statements for logic

```
electricity.cpp:1-X
electricity.cpp:1
1 #include <iostream>
2 #include <svector>
3 #include <string>
4 using namespace std;
5
6 class Appliance {
7 public:
8     string name;
9     double power; // watts
10    double hours; // hours per day
11
12    double dailyEnergy() {
13        return (power * hours) / 1000.0; // kWh
14    }
15 };
16
17 int main() {
18     int n;
19     double costPerUnit;
20     vector<Appliance> appliances;
21
22     cout << "Enter number of appliances: ";
23     cin >> n;
24
25     for (int i = 0; i < n; i++) {
26         Appliance a;
27
28         cout << "\nEnter appliance " << i + 1 << " name: ";
29         cin >> ws;
30         getline(cin, a.name);
31
32         cout << "Power (in watts): ";
33         cin >> a.power;
34
35         cout << "Usage per day (in hours): ";
36         cin >> a.hours;
37
38         appliances.push_back(a);
39     }
40
41     cout << "\nEnter electricity cost per unit (per kWh): ";
42     cin >> costPerUnit;
43
44     double totalEnergy = 0;
45
46     cout << "\n--- Daily Usage Report ---\n";
47     for (int i = 0; i < appliances.size(); i++) {
48         double energy = appliances[i].dailyEnergy();
49         totalEnergy += energy;
50         cout << appliances[i].name << ": " << energy << " kWh\n";
51     }
52
53     double totalCost = totalEnergy * costPerUnit;
54
55     cout << "\nTotal Daily Consumption: " << totalEnergy << " kWh";
56     cout << "\nEstimated Daily Cost: Rs. " << totalCost;
57
58     return 0;
59 }
```



System Features & Capabilities

1

User Input Module

Accepts units consumed from users through console interface with input validation

2

Automatic Calculation

Applies tiered pricing logic to compute total bill amount based on consumption slabs

3

Bill Display

Presents complete billing details including units, rate slabs, and final amount

4

Console Interface

Simple, intuitive text-based interface for user interaction and system navigation



Key Learning Outcomes

01

Enhanced C++ Proficiency

Strengthened understanding of core programming concepts including functions, loops, conditionals, and file handling operations

03

Problem-Solving Skills

Developed systematic approach to break down complex problems into manageable, implementable components

02

Practical Application

Gained hands-on experience applying theoretical knowledge to solve real-world billing system requirements

04

Foundation for Expansion

Created scalable base that can be extended with database integration, GUI interface, or advanced billing features



Conclusion & Future Scope

Project Summary

This C++ implementation successfully demonstrates automated electricity bill calculation with accurate results and reduced manual intervention. The system provides a solid foundation for basic billing operations.

Future Enhancements

- Database integration using SQLite or MySQL
- Graphical user interface development
- Customer management system
- Bill history tracking and reporting





Thank You