K.RAMAKRISHNAN
COLLEGE OF ENGINEERING
An Autonomous Institution
Permanently Affiliated to Anna University Chennai, Approved by AICTE New Delhi,
ISO 9001:2015, 14001:2015 certified institution, Accredited by NBA and with A grade by NAAC
Samayapuram, Tiruchirappalli – 621 112, Tamilnadu, India.

ONLINE EB BILL CALCULATION

A PROJECT REPORT

Submitted by

KAVIYARASAN C (8115U23ME023)

in partial fulfillment of requirements for the award of the course

MGB1201 - PYTHON PROGRAMMING

in

DEPARTMENT OF MECHANICAL ENGINEERING

K. RAMAKRISHNAN COLLEGE OF ENGINEERING

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

DECEMBER - 2024

K. RAMAKRISHNAN COLLEGE OF ENGINEERING

(Autonomous Institution affiliated to Anna University, Chennai)

TRICHY-621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"ONLINE EB BILL CALCULATION"** is the bonafide work of **KAVIYARASAN C (8115U23ME023)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

SIGNATURE
Dr. T. M. NITHYA, M.E.,Ph.D.,
HEAD OF THE DEPARTMENT
ASSOCIATE PROFESSOR
Department of CSE
K.Ramakrishnan College of
Engineering (Autonomous)
Samayapuram–621112.

SIGNATURE
Mrs.S.RAJESWARI M.E.
SUPERVISOR
ASSISTANT PROFESSOR
Department of CSE
K.Ramakrishnan College of Engineering
(Autonomous)
Samayapuram–621112.

Submitted for the End Semester Examination held on…….

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

I declare that the project report on **"ONLINE EB BILL CALCULATION"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **MGB1201 – PYTHON PROGRAMMING**

.

Signature

KAVIYARASAN C

Place: Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Engineering (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

I would like to thank **Dr. D. SRINIVASAN, B.E, M.E., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. T. M. NITHYA, M.E.,Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express our deep expression and sincere gratitude to our project supervisor **Mrs.S.RAJESWARI M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To achieve a prominent position among the top technical institutions

MISSION OF THE INSTITUTION

M1: To bestow standard technical education par excellence through state of the art infrastructure, competent faculty and high ethical standards.

M2: To nurture research and entrepreneurial skills among students in cutting edge technologies.

M3: To provide education for developing high-quality professionals to transform the society.

VISION OF THE DEPARTMENT

To create eminent professionals of Computer Science and Engineering by imparting quality education.

MISSION OF THE DEPARTMENT

M1: To provide technical exposure in the field of Computer Science and Engineering through state of the art infrastructure and ethical standards.

M2: To engage the students in research and development activities in the field of Computer Science and Engineering.

M3: To empower the learners to involve in industrial and multi-disciplinary projects for addressing the societal needs.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

Our graduates shall

PEO1: Analyse, design and create innovative products for addressing social needs.

PEO2: Equip themselves for employability, higher studies and research.

PEO3: Nurture the leadership qualities and entrepreneurial skills for their successful career.

PROGRAM OUTCOMES

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write

effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Apply the basic and advanced knowledge in developing software, hardware and firmware solutions addressing real life problems.

- **PSO2:** Design, develop, test and implement product-based solutions for their career enhancement.

ABSTRACT

This Project presents a Python-based solution for calculating online electricity bills (EB). The solution aims to provide a clear, efficient, and user-friendly method for determining the cost of electricity usage over a specified billing period. The proposed system incorporates user input for parameters such as consumption units, tariff rates, and fixed charges, which are typical components of electricity billing calculations. The system can handle various tariff structures, including tiered rates, where the cost per unit changes based on consumption thresholds. The calculation accounts for both fixed charges (e.g., meter charges) and variable charges (based on actual consumption). The approach is designed to be easily adaptable to different billing structures and regulatory requirements, making it applicable in various geographic regions. To ensure accurate calculations, the system includes robust validation mechanisms for user inputs.

ABSTRACT WITH POs AND PSOs MAPPING

ABSTRACT	POs MAPPED	PSOs MAPPED
This Project presents a Python-based solution for calculating online electricity bills (EB). The solution aims to provide a clear, efficient, and user-friendly method for determining the cost of electricity usage over a specified billing period. The proposed system incorporates user input for parameters such as consumption units, tariff rates, and fixed charges, which are typical components of electricity billing calculations. The system can handle various tariff structures, including tiered rates, where the cost per unit changes based on consumption thresholds. The calculation accounts for both fixed charges (e.g., meter charges) and variable charges (based on actual consumption).The approach is designed to be easily adaptable to different billing structures and regulatory requirements, making it applicable in various geographic regions. To ensure accurate calculations, the system includes robust validation mechanisms for user inputs.	**PO1,PO2, PO3,PO12**	**PSO1**

Note: 1- Low, 2-Medium, 3- High

SUPERVISORHEAD OF THE DEPARTMENT

TABLE OF CONTENTS

1

CHAPTER 1

INTRODUCTION

1.1 Objective

1. Develop a Robust Calculation Algorithm: Create an algorithm that can handle various tariff structures, including tiered and flat rates.

2. Incorporate Fixed and Variable Charges: Ensure that both fixed charges (such as meter fees) and variable charges (based on consumption) are accurately calculated.

3. Validate User Input: Implement input validation mechanisms to ensure data accuracy and integrity.

4. Provide Detailed Billing Information: Generate detailed billing statements that include all components of the charges.

5. Ensure Adaptability: Design the system to be flexible and adaptable to different geographic regions and regulatory requirement

1.2 Overview

The Online Electricity Bill Calculation System is a Python-based project aimed at automating the computation of electricity bills for domestic and commercial users. It simplifies the billing process by taking user inputs (like category and units consumed), applying relevant tariff rates, and generating an invoice.

1.3 Python Programming Concepts

1. Variables and Data Types

- Variables: Store user inputs, such as electricity consumption data, rates, and billing details.
- Data Types: Use integers for numerical data (e.g., unit consumption), floats for decimal numbers (e.g., rates), and strings for textual data (e.g., user names).

2. Control Structures

- Conditional Statements: Use if-else statements to make decisions based on user inputs (e.g., calculating different rates for different consumption slabs).
- Loops: Implement for and while loops for repetitive tasks, such as iterating through user records.

3. Functions

- Functions: Define reusable blocks of code to perform specific tasks like calculating total bills, categorizing expenses, or generating reports.
- Parameters: Pass necessary data to functions to perform calculations and return results.

4. Data Structures

- Lists: Store multiple items, such as a list of user inputs or monthly bills.
- Dictionaries: Use key-value pairs to organize related data, like user information and categorized expenses.

5. Modules and Libraries

- Built-in Modules: Use standard Python libraries like math for mathematical operations and datetime for handling dates.

6. File Handling

- Reading and Writing Files: Use Python's file handling capabilities to save and retrieve user data (open(), read(), write(), and close() functions).

7. Exception Handling

- Try-Except Blocks: Manage errors gracefully to ensure the program doesn't crash unexpectedly, handling exceptions such as incorrect user inputs or file access errors.

8. Object-Oriented Programming (OOP)

- Classes and Objects: Model real-world entities, like users and bills, and their interactions.
- Encapsulation: Group related properties and methods within classes for better organization and reusability.

CHAPTER 2

PROJECT METHODOLOGY

2.1Proposed Work

1.User Input Module

- Develop a user-friendly interface for entering electricity consumption data.
- Ensure efficient data storage for future reference and calculations.

2.Data Storage

- Implement secure storage solutions to maintain user data integrity.
- Design a database schema to organize and manage the collected data.

3.Expense Categorization

- Create algorithms to automatically classify electricity expenses into categories.
- Utilize Python functions to implement the categorization logic.

4.Total and Balance Calculation

- Formulate equations to compute the total electricity bill and any remaining balance.
- Integrate these calculations within the system to provide real-time updates.

5.Budget Alerts

- Develop a notification system to alert users if their electricity usage exceeds predefined budget limits.
- Use scheduling libraries to send timely alerts and reminders.

6.Data Persistence

- Ensure that all user data is saved and retrievable across sessions.

- Implement robust file handling or database transactions to maintain data persistence.

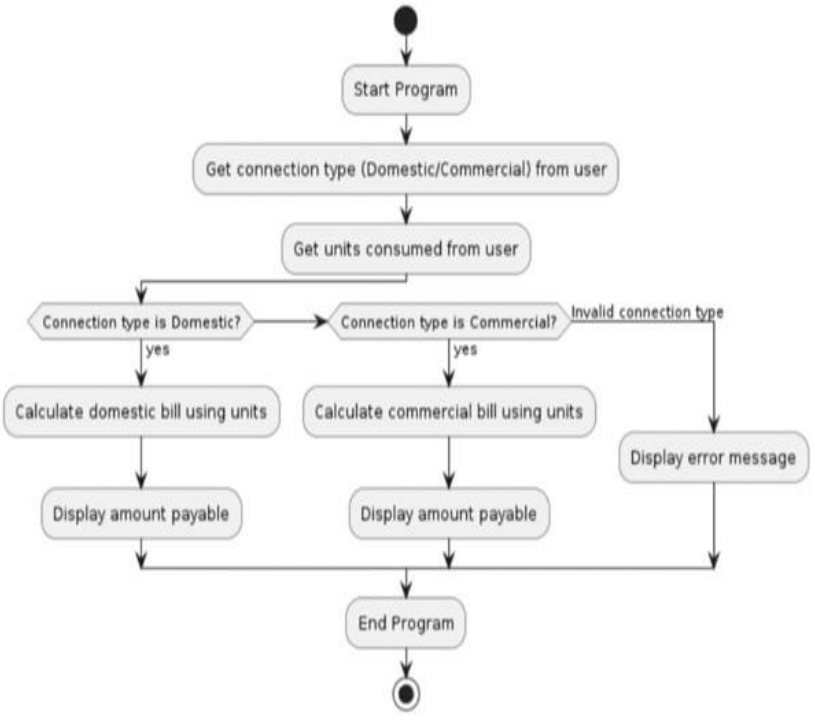
7.Simple Analytics

- Summarize and visualize usage patterns using analytical models.
- Employ libraries like Pandas and Matplotlib to create informative graphs and charts.

8.Report Generation

- Develop templates for generating detailed monthly, quarterly, and yearly reports.
- Use libraries such as Report Lab or FPDF to produce professional-quality PDF reports.

2.2 Block Diagram



CHAPTER 3

MODULE DESCRIPTION

1. Input/Output

- input() function: Used to take user input for the connection type and units consumed.
- print() function: Used to display the calculated amount payable to the user.

2. Conditional Statements

- if-else statements: Used to determine the electricity rate based on the number of units consumed and to calculate the electricity duty based on the total bill amount.

3. Functions

- Function Definition: Functions like calculate_domestic_bill() and calculate_commercial_bill() are defined to encapsulate the logic for calculating bills based on different criteria.
- Function Calls: These functions are called within the main() function to perform specific tasks based on user input.

4. Main Guard

- if __name__ == "__main__": This is used to ensure that the main() function runs only when the script is executed directly, not when imported as a module.

5. Arithmetic Operations

- Used to calculate the bill amount based on units consumed and to apply electricity duty based on the total bill amount.

CHAPTER 4

RESULTS AND DISCUSSION

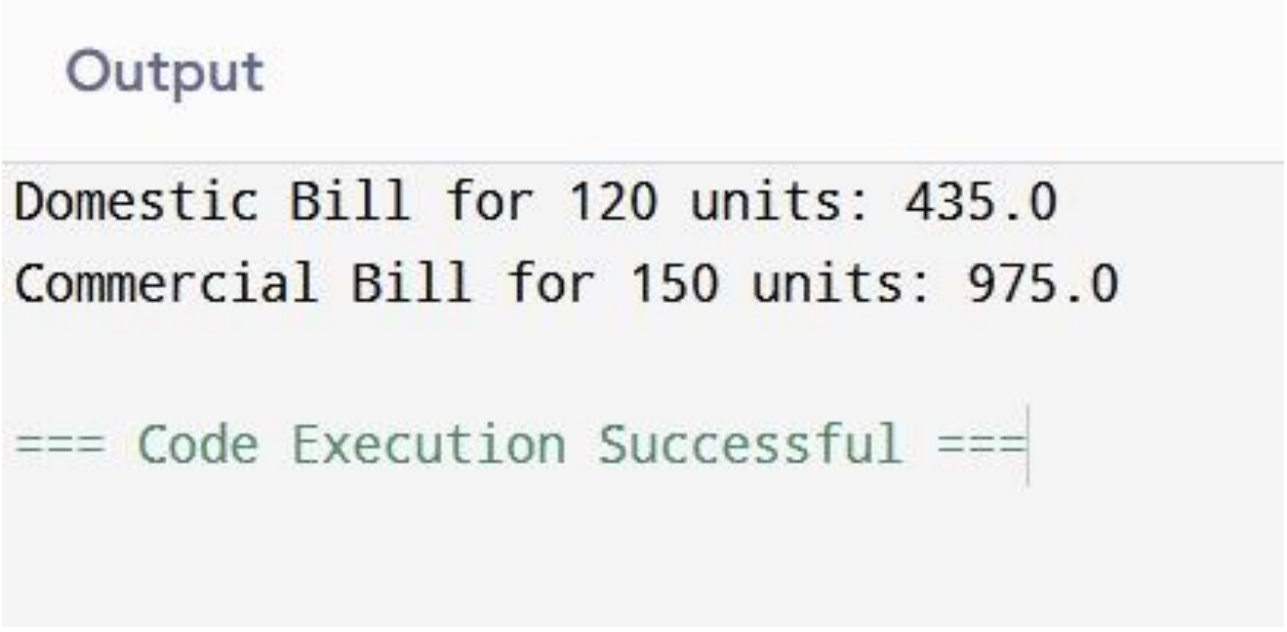
PROGRAM

```
# Function to calculate domestic bill
def calculate_domestic_bill(units):
    if units <= 50:
        return units * 2.3
    elif units <= 100:
        return 50 * 2.3 + (units - 50) * 4.2
    else:
        return 50 * 2.3 + 50 * 4.2 + (units - 100) * 5.5
def calculate_commercial_bill(units):
    if units <= 50:
        return units * 5.2
    elif units <= 100:
        return 50 * 5.2 + (units - 50) * 6.8
    else:
        return 50 * 5.2 + 50 * 6.8 + (units - 100) * 7.5
domestic_units = 120
commercial_units = 150

domestic_bill = calculate_domestic_bill(domestic_units)
commercial_bill = calculate_commercial_bill(commercial_units)

print("Domestic Bill for", domestic_units, "units:", domestic_bill)
print("Commercial Bill for", commercial_units, "units:", commercial_bill)
```

OUTPUT


Output
Domestic Bill for 120 units: 435.0
Commercial Bill for 150 units: 975.0
=== Code Execution Successful ===

CHAPTER 5

CONCLUSION

This Python program for online electricity bill calculation illustrates the importance and benefits of modular design. By breaking down the problem into smaller, manageable components, the program is easier to develop, test, and maintain. Robust input validation ensures data integrity, while the flexible calculation logic and application of discounts and penalties cater to a wide range of billing scenarios. Overall, the program serves as an effective and scalable solution for online electricity bill calculation.

REFERENCES:

1. Anurag Gupta, IPS Jharkhand, GP Biswass, " Python Programming,
2. Reema Theraja , "Python Programming: Using Problem Solving Approach" Oxford Higher Education,2014.
3. Gowrishankar S, Veena A, "Introduction to Python Programming".
4. https//:w3schools.com 16

APPENDIX

(Coding)

```
# Function to calculate domestic bill
def calculate_domestic_bill(units):
    if units <= 50:
        return units * 2.3
    elif units <= 100:
        return 50 * 2.3 + (units - 50) * 4.2
    else:
        return 50 * 2.3 + 50 * 4.2 + (units - 100) * 5.5

# Function to calculate commercial bill
def calculate_commercial_bill(units):
    if units <= 50:
        return units * 5.2
    elif units <= 100:
        return 50 * 5.2 + (units - 50) * 6.8
    else:
        return 50 * 5.2 + 50 * 6.8 + (units - 100) * 7.5

# Example usage:
domestic_units = 120
commercial_units = 150

domestic_bill = calculate_domestic_bill(domestic_units)
commercial_bill = calculate_commercial_bill(commercial_units)
```

```
print("Domestic Bill for", domestic_units, "units:", domestic_bill)
print("Commercial Bill for", commercial_units, "units:", commercial_bill)
```

APPENDIX

(Output)

Domestic Bill for 120 units: 435.0

Commercial Bill for 150 units: 975.0

=== Code Execution Successful ===

=== Session Ended. Please Run the code again ===