

**PSP [20ES104] COURSE PROJECT REPORT**

**On**

**“Electricity Bill Calculator”**

Developed By:

**H.T.NO STUDENT NAME**

2203A51350 G. Vikram Yadav

2203A51334 A. Nishank Reddy

2203A51332 A. Akhilesh Bhargav

2203A51344 E. Sai Suhas Chandra

Under the Guidance of

Mr. Riyaz Mohammed

Assistant Professor

Submitted to

Department Computer Science and Artificial Intelligence SR University

Ananthasagar(V), Hasanparthy(M), Hanamkonda(Dist.) – 506371

[www.sru.edu.in](http://www.sru.edu.in/)

**June 2023**

**Department of Computer Science and Engineering**

**CERTIFICATE**

This is to certify that the PSP course project report entitled **“Electricity Bill Calculator”** is a record of bonafide work carried out by the student(s) G. Vikram, A. Akhilesh, A. Nishank, E. Sai Suhas bearing roll number(s)2203A51350 , 2203A51332 ,2203A51334,2203A51344 of Computer Science and Engineering department during the academic year 2022-23.

**Supervisor**

(Riyaz Mohammed)

**INDEX**

**Sl. No Title Page No.**

1. Problem statement 1
2. Module-wise description 2
3. Knowledge required to develop the project 4
4. Source code (.c file code followed by .h file code) 5
5. Results 35

# PROBLEM STATEMENT:

The Electricity Cost Calculator project is an application based micro project that predicts the following month’s electricity bill based on the appliances or loads used. This project employs a multi-file and multi-platform strategy (Linux and Windows). People who do not have a technical understanding of calculating power bills can use this program to forecast their electricity bills for the coming months; however, an electricity bill calculator must have the following features:

• All loads’ power rating

• Unit consumed per day

• Units consumed per month

• Total load calculation

Provide the functionality for below mentioned:

* 1. Reads year month and date dynamically:
     + Enter year:
     + Enter month (1-12):
     + Enter date (1-31):
  2. Reads total units consumed by the user in the particular month:
     + Enter total units consumed:
  3. Calculates the total bill and displays the amount of taxes are applied:
     + Units consumed per day:
     + Bill:
     + Tax:
     + Surcharge:
     + Total bill:
  4. This code also shows the previous bills information, and included an option to clear the previous data.

# MODULE-WISE DESCRIPTION:

In this application all variables and structure are declared globally so that these variables and structure members can be accessed throughout the program at any function call. The memory allocation will be done in this program dynamically. The application asks the person who runs the program to enter total units consumed so that it generates a final price to be paid.

In this application four modules are used.

1. Read/Input Module:
   * The user is prompted to enter the year, month, date, and units consumed.
   * Based on the units consumed, the corresponding bill amount is calculated.
   * The tax, surcharge, and total bill amount are calculated.
   * The bill details are stored in a file named "bills.txt" using the **fprintf** function.
2. Print Module:
   * The user is prompted to print the bill details.
   * If the user chooses to print (by entering 'y' or 'Y'), the contents of the "bills.txt" file are read and printed to the console using the **fgetc** function.
3. Clear Module:
   * The user is prompted to clear the bill details.
   * If the user chooses to clear (by entering 'y' or 'Y'), the "bills.txt" file is cleared using the **clearAllBills** function.
   * The **clearAllBills** function opens the "bills.txt" file in write mode ("w") and then immediately closes it, effectively clearing its contents.

**KNOWLEDGE REQUIRED TO DEVELOP THIS APPLICATION**

* + Control Statements (if, if-else)
  + Loop Statements (while, for)
  + Files
  + Functions (Any type of user defined functions)
  + Structure (structures and nested structures)
  + Pointers (pointer to strings and pointers to structures)

**SOURCE CODE [.C FILE]:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

typedef struct {

int date;

int month;

int year;

int units;

float bill;

float tax;

float surcharge;

float totalBill;

} Bill;

int main() {

int year, month, date;

int units;

float totalBill, surcharge, tax, bill, unitPerDay;

printf(" WELCOME TO ELECTRICITY BILL CALCULATOR\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n");

printf("ELECTRICITY BOARD RATE CHART (Rates/Unit):\n\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 0 to 50 Units = Rs 1/Unit \*\n\n");

printf("\* 51 to 100 Units = Rs 1.5/Unit \*\n\n");

printf("\* 101 to 150 Units = Rs 2.5/Unit \*\n\n");

printf("\* 151 to 200 Units = Rs 3.5/Unit \*\n\n");

printf("\* 200 to 250 Units = Rs 5.0/Unit \*\n\n");

printf("\* Above 250 units = Rs 7.0/Unit \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n");

printf("Enter the year: ");

scanf("%d", &year);

printf("Enter the month: ");

scanf("%d", &month);

printf("Enter the date: ");

scanf("%d", &date);

printf("Enter total units consumed: ");

scanf("%d", &units);

if (units <= 50) {

bill = units \* 1.0;

} else if (units <= 100) {

bill = 50 \* 1.0 + (units - 50) \* 1.5;

} else if (units <= 150) {

bill = 50 \* 1.0 + 50 \* 1.5 + (units - 100) \* 2.5;

} else if (units <= 200) {

bill = 50 \* 1.0 + 50 \* 1.5 + 50 \* 2.5 + (units - 150) \* 3.5;

} else if (units <= 250) {

bill = 50 \* 1.0 + 50 \* 1.5 + 50 \* 2.5 + 50 \* 3.5 + (units - 200) \* 5.0;

} else {

bill = units \* 7.0;

}

unitPerDay = units / 28;

tax = bill \* 0.2;

surcharge = bill + 50;

totalBill = bill + tax + surcharge;

printf("\nUnits per day : Rs %.2f\n", unitPerDay);

printf("ENERGY CHARGES : Rs %.2f\n", bill);

printf("CUST CHARGES : Rs %.2f\n", tax);

printf("FIXED CHARGES : Rs %.2f\n", surcharge);

printf("..................................................\n");

printf("\tNET AMOUNT : Rs %.2f\t\t\n", totalBill);

printf("\tARREARS : Rs 0.00 ");

printf("\n--------------------------------------------------\n");

printf("|\tNET AMOUNT : Rs %.2f\t\t|\n", totalBill);

printf("--------------------------------------------------\n");

void printAllBills(Bill \*bills, int numBills){

printf("\nBILL DETAILS\n");

printf("--------------------------------------------------\n");

for (int i = 0; i < numBills; i++) {

printf("Date: %02d/%02d/%d\n", bills[i].date, bills[i].month, bills[i].year);

printf("Units: %d\n", bills[i].units);

printf("Bill: Rs %.2f\n", bills[i].bill);

printf("Tax: Rs %.2f\n", bills[i].tax);

printf("Surcharge: Rs %.2f\n", bills[i].surcharge);

printf("-----------------------\n");

printf("Total Bill: Rs %.2f\n", bills[i].totalBill);

printf("-----------------------\n");

}

printf("--------------------------------------------------\n");

}

void clearAllBills() {

FILE \*file = fopen("bills.txt", "w");

if (file == NULL) {

printf("Error opening the file.\n");

return;

}

fclose(file);

}

int compareBills(const void \*a, const void \*b) {

const Bill \*bill1 = (const Bill \*)a;

const Bill \*bill2 = (const Bill \*)b;

// Compare year

if (bill1->year != bill2->year) {

return bill1->year - bill2->year;

}

// Compare month

if (bill1->month != bill2->month) {

return bill1->month - bill2->month;

}

// Compare date

return bill1->date - bill2->date;

}

FILE \*file = fopen("bills.txt", "a");

if (file == NULL) {

printf("Error opening the file.\n");

return 1;

}

fprintf(file, "Date: %02d/%02d/%d\n", date, month, year);

fprintf(file, "```````````````````````\n");

fprintf(file, "Units : %d\n", units);

fprintf(file, "Bill : Rs %.2f\n", bill);

fprintf(file, "Tax : Rs %.2f\n", tax);

fprintf(file, "Surcharge : Rs %.2f\n", surcharge);

fprintf(file, "-----------------------\n");

fprintf(file, "Total Bill: Rs %.2f\n", totalBill);

fprintf(file, "-----------------------\n\n");

fclose(file);

// Prompt to print the bill details

char printOption;

printf("\nDo you want to print the bill details? (y/n): ");

scanf(" %c", &printOption);

if (printOption == 'y' || printOption == 'Y') {

file = fopen("bills.txt", "r");

if (file == NULL) {

printf("Error opening the file.\n");

return 1;

}

char ch;

printf("\nBILL DETAILS\n");

printf("--------------------------------------------------\n");

while ((ch = fgetc(file)) != EOF) {

printf("%c", ch);

}

printf("--------------------------------------------------\n");

fclose(file);

}

else {

exit(0);

}

// Prompt to clear the bill details

char clearOption;

printf("\nDo you want to clear the bill details? (y/n): ");

scanf(" %c", &clearOption);

if (clearOption == 'y' || clearOption == 'Y') {

clearAllBills();

printf("Bill details cleared successfully.\n");

}

return 0;

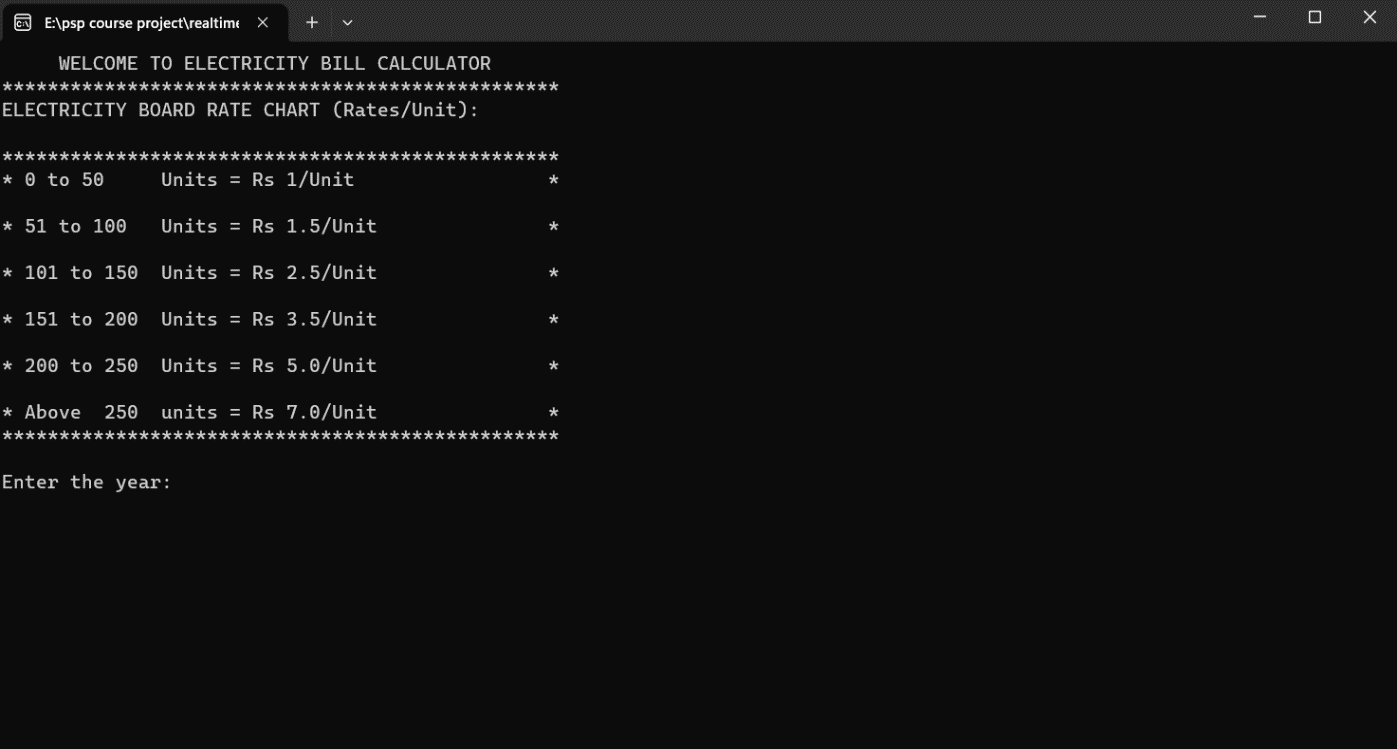
}

**RESULTS**

**1.1-**

Run the program.

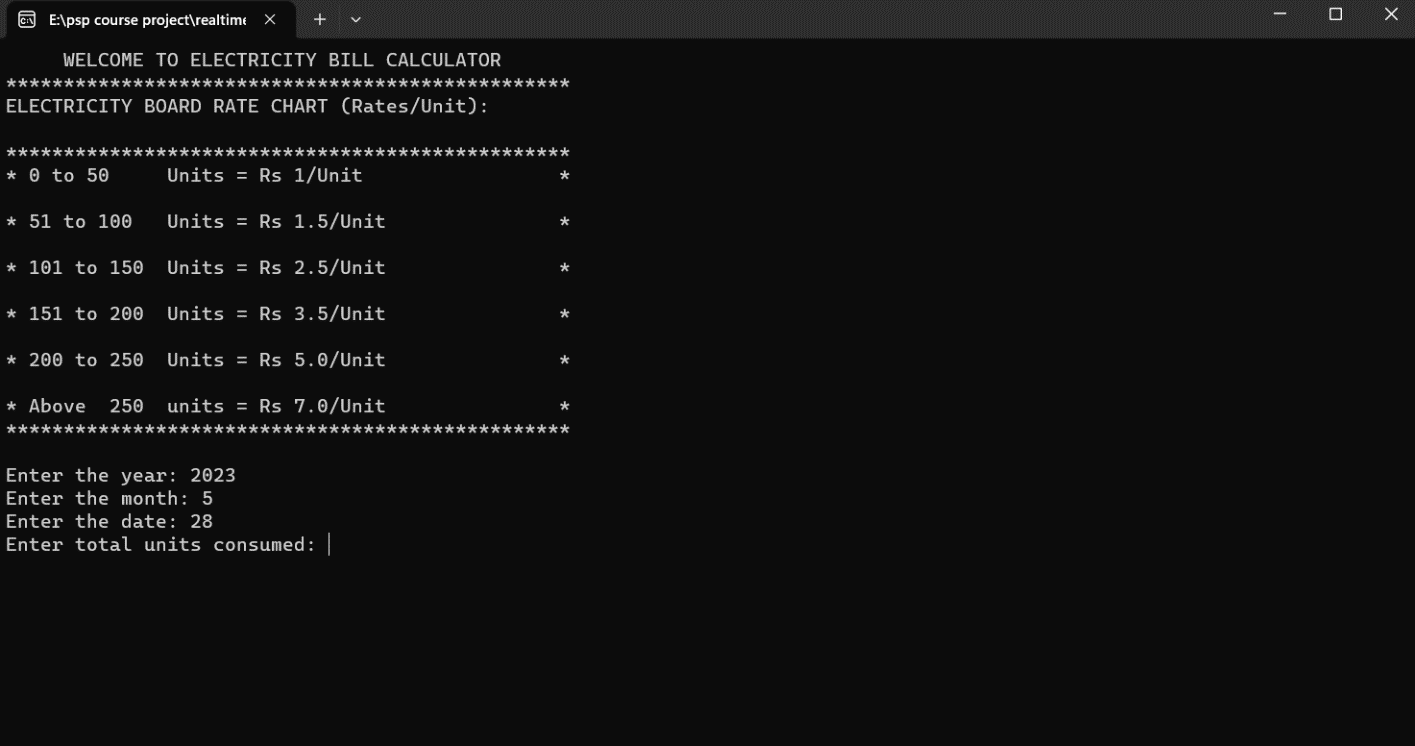
Enter year, month and date .

****

**………………………………………………………………………………………**

**1.2-**

Enter total units consumed by the user.

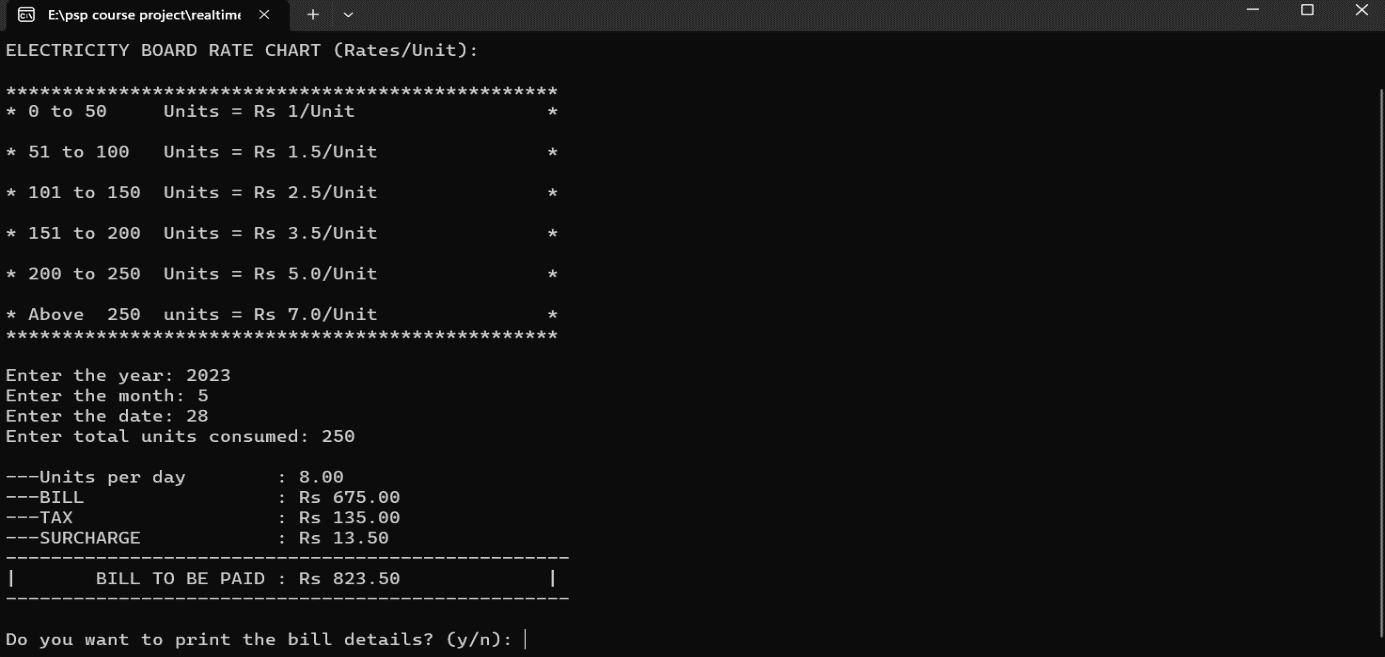


**………………………………………………………………………………………**

**1.3-**

After entering the units ,program generates and displays the total amount

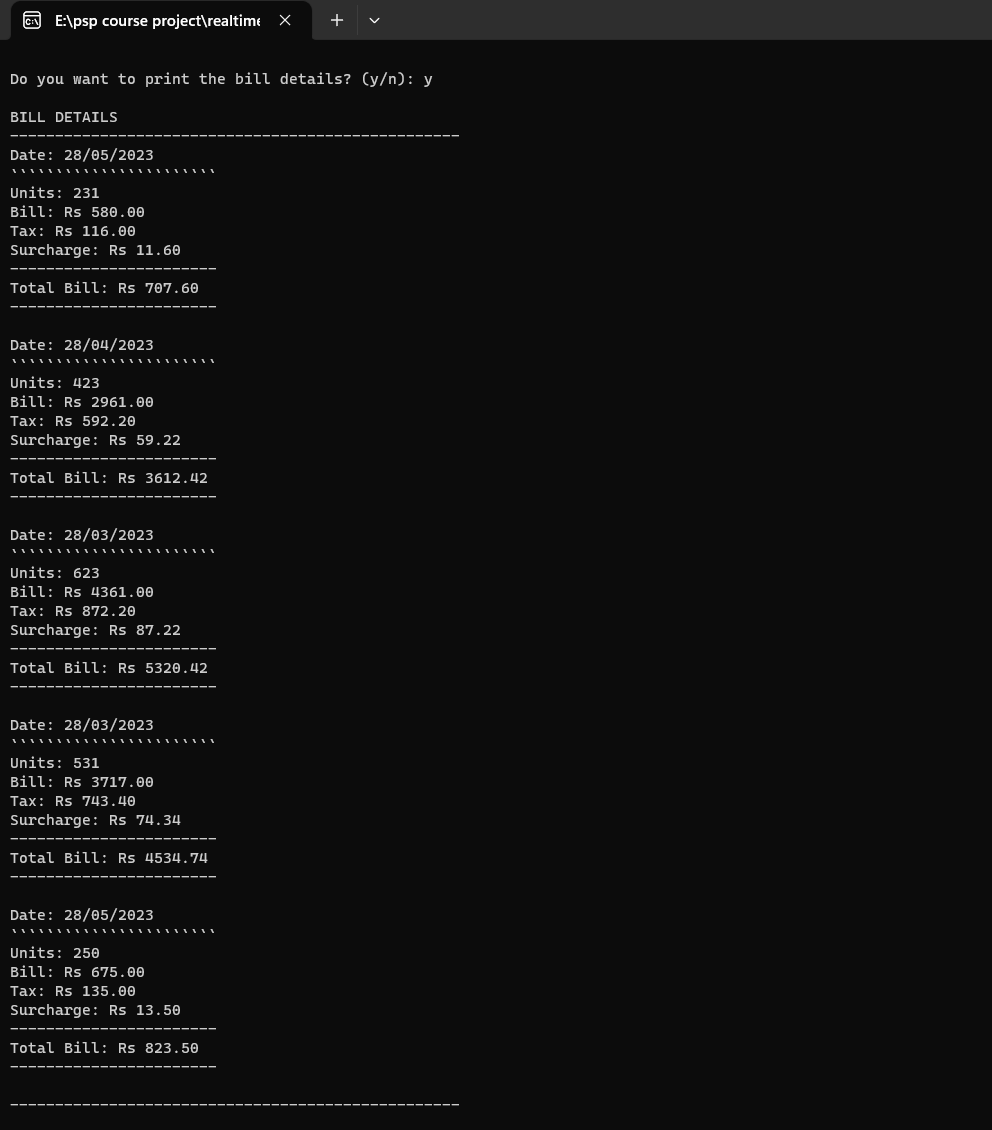
to be paid,tax,surcharge and also units consumed by the user .



**………………………………………………………………………………………**

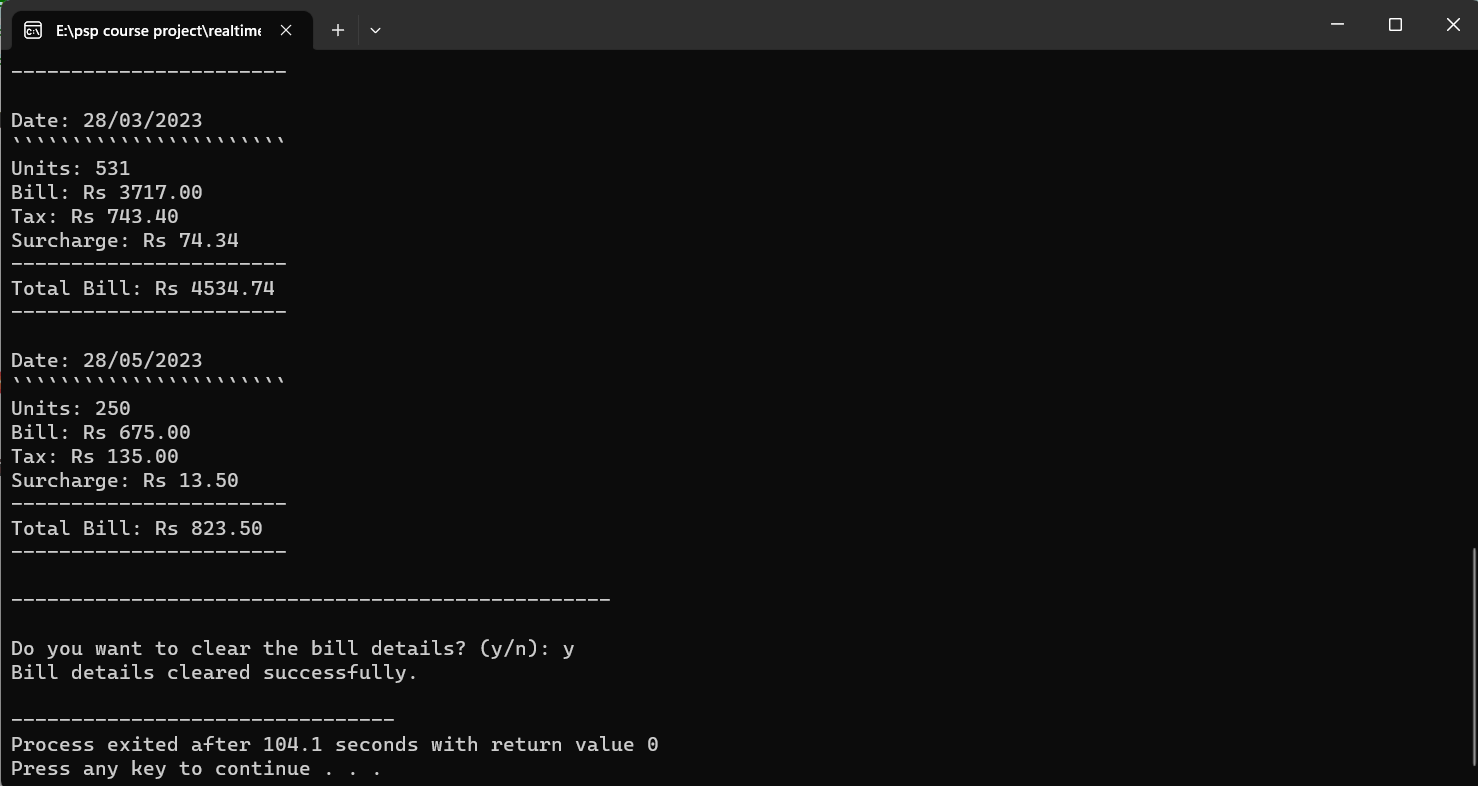
**1.4-**

If we enter ‘y’ or ‘Y’ it displays the previous bills in a specific order.



**1.5-**

We also had an option to clear the recent bill details (by entring ‘y’ / ‘Y’) at the last so that we can clear the file if the shown details are not necessary.



**………………………………………………………………………………………**

**<<<<END>>>>**