./

Report – School Billing System

Course Code: <CODE>



**Submitted By: Preethu k**

**Candidate ID: 105061**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **Approved By** | **Remarks/Revision Details** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Document History**

# 

**Table of Content**

**1. Problem Statement.**

**2. Project Description.**

**3. Existing system.**

**4. Proposed system.**

**5. Requirements.**

**6. Test Plan.**

**7. Test cases.**

**8. Expected Results.**

**SCHOOL BILLING SYSTEM.**

**1.PROBLEM STATEMENT:**

To develop a School Management System to assist in the Management of Fees and salary which ease the process of doing this job than earlier pen and paper-based management.

**2.PROJECT DESCRIPTION:**

The school Billing System project keeps records of all the students, teachers and staff working in the institution. The Program is run by the administrator who can add, search, modify and delete the records according to the need. The basic feature of this project is that it shows fees that the students need to pay or dues and advance of the students. It also records the information related to the salary that is to be provided to the teachers and the staff working in the organisation.

The data within the program can be recorded by the input of current month and date. This system utilizes data structure to store the records and provides access to the user whenever required. Users shall have different access privileges for the system. It is a user friendly web based system which efficiently takes care of the billing activities of the school.

**3.EXISTING SYSTEM:**

The existing system did not have the option to track bills or fees payments from students. It was done manually. It was a laborious task. The system did not have the option to print the term reports or fees payment reports. All the accounts were maintained manually. It was a difficult task to manage as the school accommodated large number of students.

**4.PROPOSED SYSTEM:**

This school billing system will help the user to print the fee paid reports. There are different types of categories like teacher, student and staff. This system makes billing automated, accurate and easier. This kind of application is very essential for any small or medium sized organisation. An accountant, regardless of the number of staff and students, will maintain all records digitally.

**5.REQUIREMENTS:**

**Hardware Requirement:**

The hardware minimum and maximum recommended requirements are listed below.

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Recommended of Requirements.** | **Maximum Recommended of Requirements.** |
| Internal Memory (RAM) | 2.00GB | 3.00GB or Higher |
| Hard Disk Capacity (CPU) | 60.00GB | 80.00GB or Higher |
| Processor | Intel Pentium 1.60GHZ | Intel(R) core 2.40GHZ or Higher |
| Monitor | 17” Cobrod 32bit | 18” Cobrod or higher 64bit |
| Video Card | 128MB AGP | 256 MB AGP or Higher |

**Software Requirement:**

The software minimum recommended requirements and maximum recommended requirements are listed below.

|  |  |  |
| --- | --- | --- |
| **Software** | **Minimum Recommended Requirements** | **Maximum recommended Requirements** |
| System type | Microsoft WIN7 or XP 32bit operating System | Microsoft Win10 64bit operating system |
| Storage | FAT File system | NTFS File System. |
| Programming language | Dev c++ for Windows 7 | Dev c++ for Windows 8 or Higher |

**6.TEST PLAN:**

In the software development project, errors can be injected at any stage during development. There are different techniques for detecting and eliminating errors that originate in that phase. However, no technique is perfect, and it is expected that some of the errors of the earlier phases will finally manifest themselves in the code. This is particularly true because in the earlier phases and most of the verification techniques are manual because no executable code exists. ultimately those remaining errors will be reflected on the code. Hence, the code developed during the coding activity is likely to have some requirement errors and design errors, in addition to errors introduced during the coding activity. Behaviour can be observed, testing is the phase where the errors remaining from all the previous phases must be detected. Hence, testing performs a very critical role for quality assurance and for ensuring the reliability of the software.

During testing the program to be tested is executed with a set of test cases, and the output of the program for the test case is evaluated to determine. If the programme is performing as expected. Due to its approach, dynamic testing can only ascertain the presence of errors in the programme, the exact nature of the errors is not usually decided by testing. Testing forms the first step in determining the errors in a program. Clearly the success of testing in revealing errors in programs depends critically on the test cases.

Testing a large system is a very complex activity, and like any complex activity it has to be broken into smaller activities. Due to this, for a project, incremental testing is generally performed, in which components and subsystems of the system are tested separately before integrating them to form the system for system testing. This form of testing, though necessary to ensure quality for a large system, introduces new issues of how to select components for testing and how to combine them to form a subsystem and system.

**7.TEST CASE:**

**>** Data structures have been used effectively to handle co-related functions and      store the record. Data structures required are:

**>** struct dat - to store the date (month and day) of entry of records.

**>** struct student - to store and organise the record of individual students.

**>** struct teacher - to store and organise the record of individual teachers/staff.

**>** Different functions are used for performing different billing operations in the   school billing system.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO OF TEST CASES** | **FUNCTIONS** | **DESCRIPTION** | **STATUS** |
| **TEST CASE 1** | start() | Shows the account selection screen | Pass |
| **TEST CASE  2** | chkdat() | For checking date | Pass |
| **TEST CASE  3** | addrec() | For adding records | Pass |
| **TEST CASE  4** | modrec() | For modifying records | Pass |
| **TEST CASE  5** | searchrec() | For searching records | Pass |
| **TEST CASE 6** | delrec() | For deleting records | Pass |
| **TEST CASE  7** | fee() | For recording the fee paid and displaying fine, due, total and advance. | Pass |
| **TEST CASE  8** | salary() | For calculating the salary of teachers and staff | Pass |

**8.EXPECTED RESULT:**

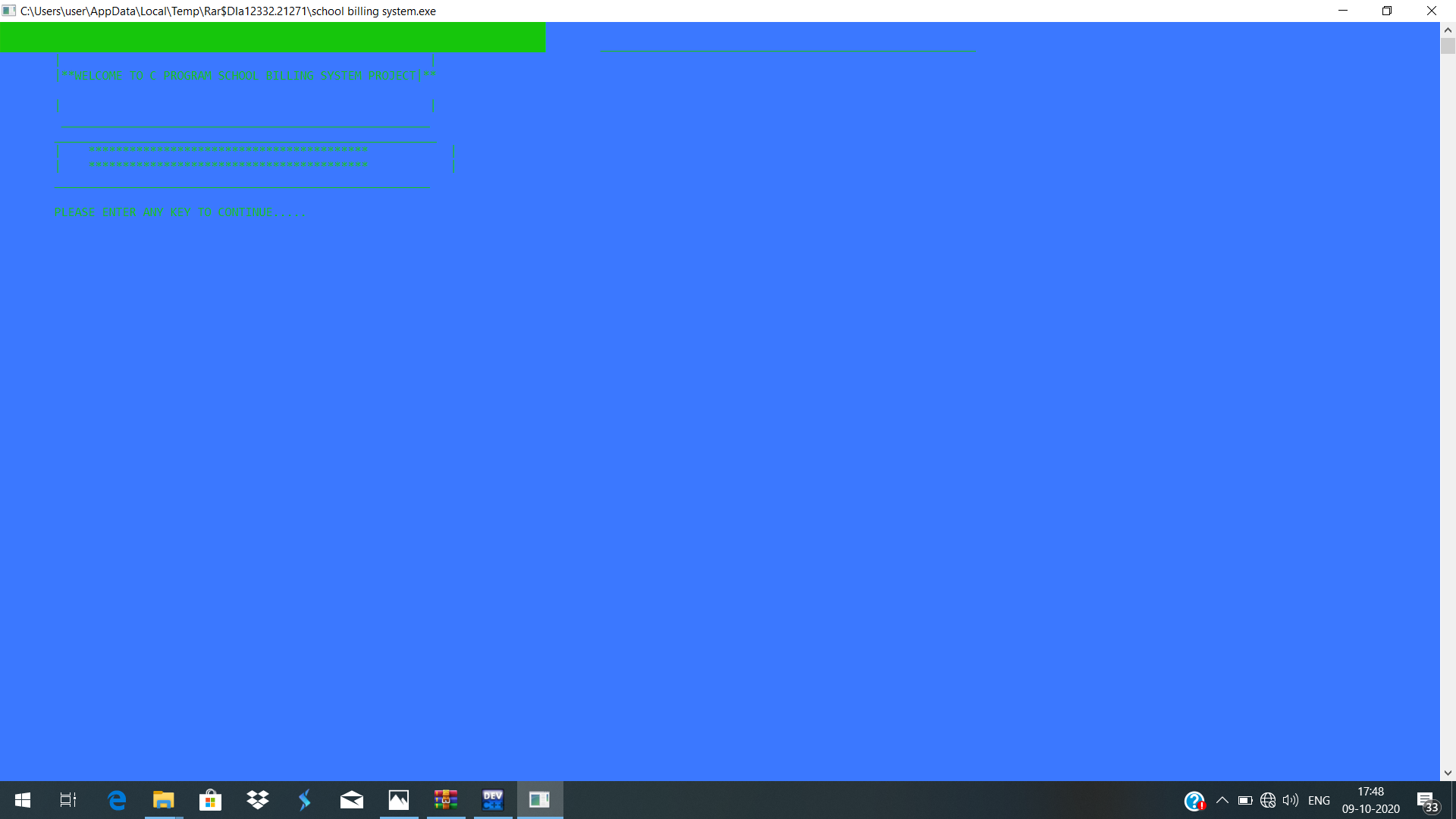
**>** In this project, we can add, modify, search and delete the records of both the account types i.e. Students, Teachers and Staff.

**>** In addition to that, this mini project in C allows us to display fees, dues, total and advance of students, and salary-related information of teachers and staff.

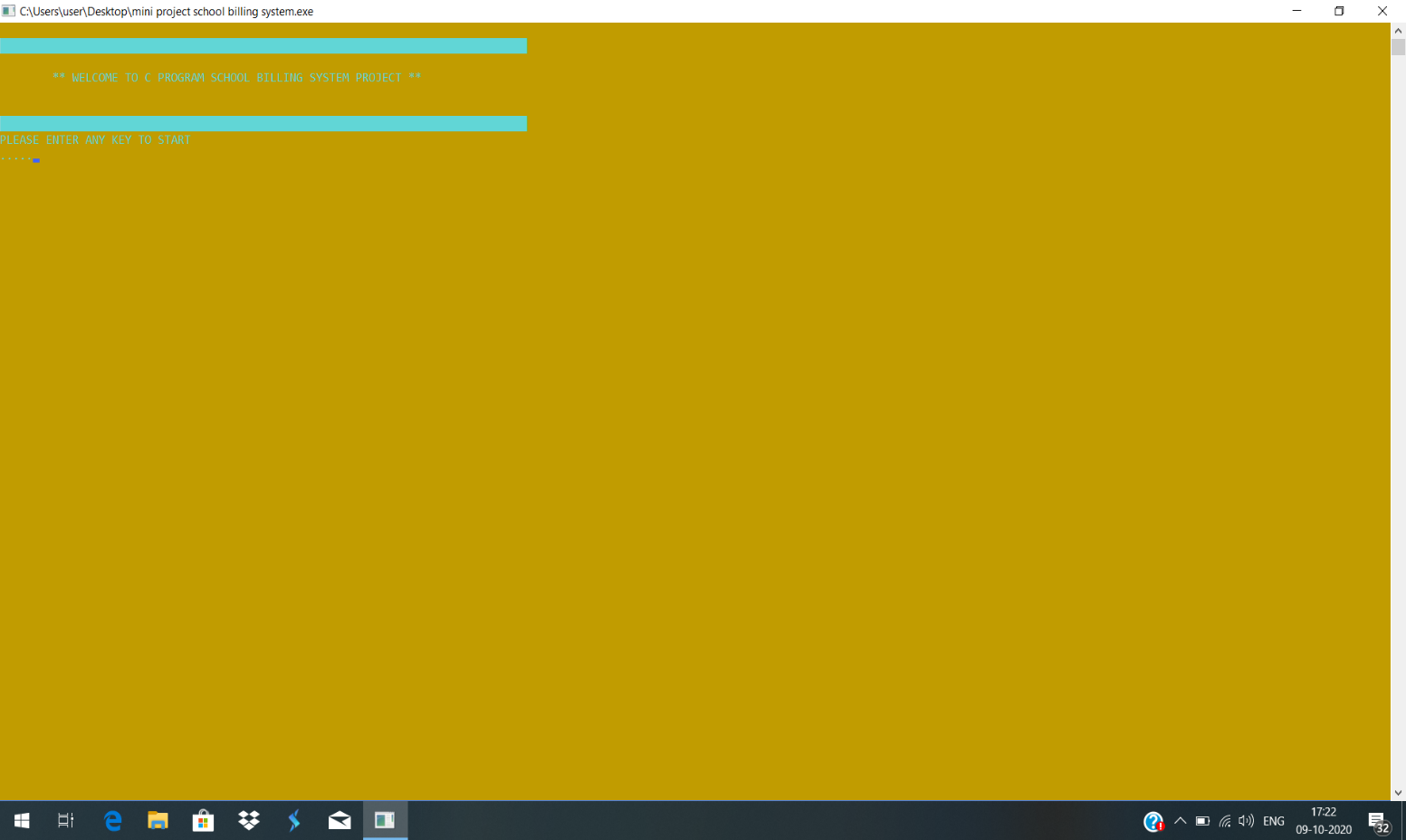
**>** The data within the programme can be recorded by the input of current month and date. The administrator will need to select the type of account for either student or teacher and can perform the billing operation as required.

**>** The project keeps the record of student’s name, class and roll number while it also keeps the records of the teachers in the similar way.

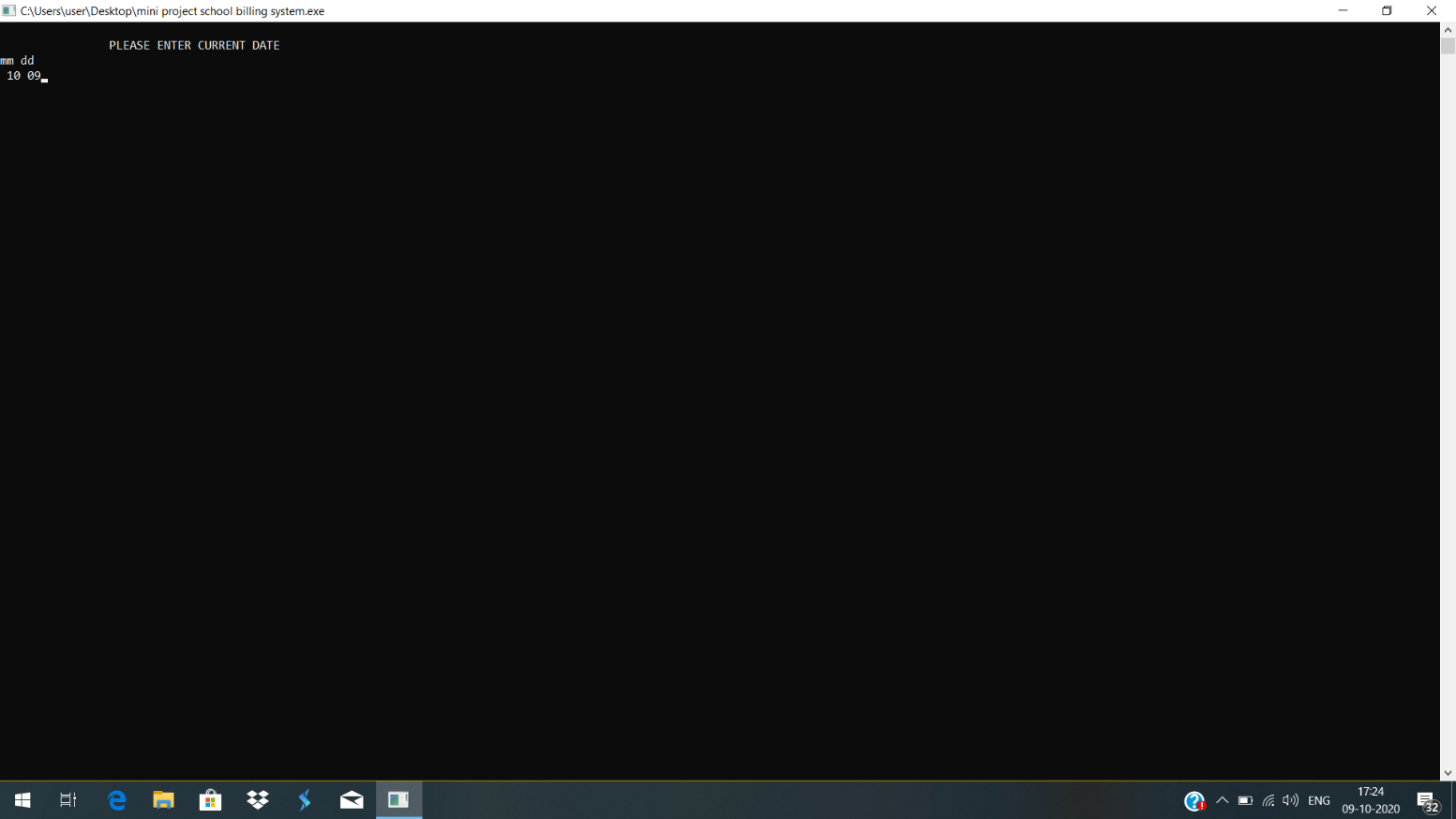
WELCOME PAGE:



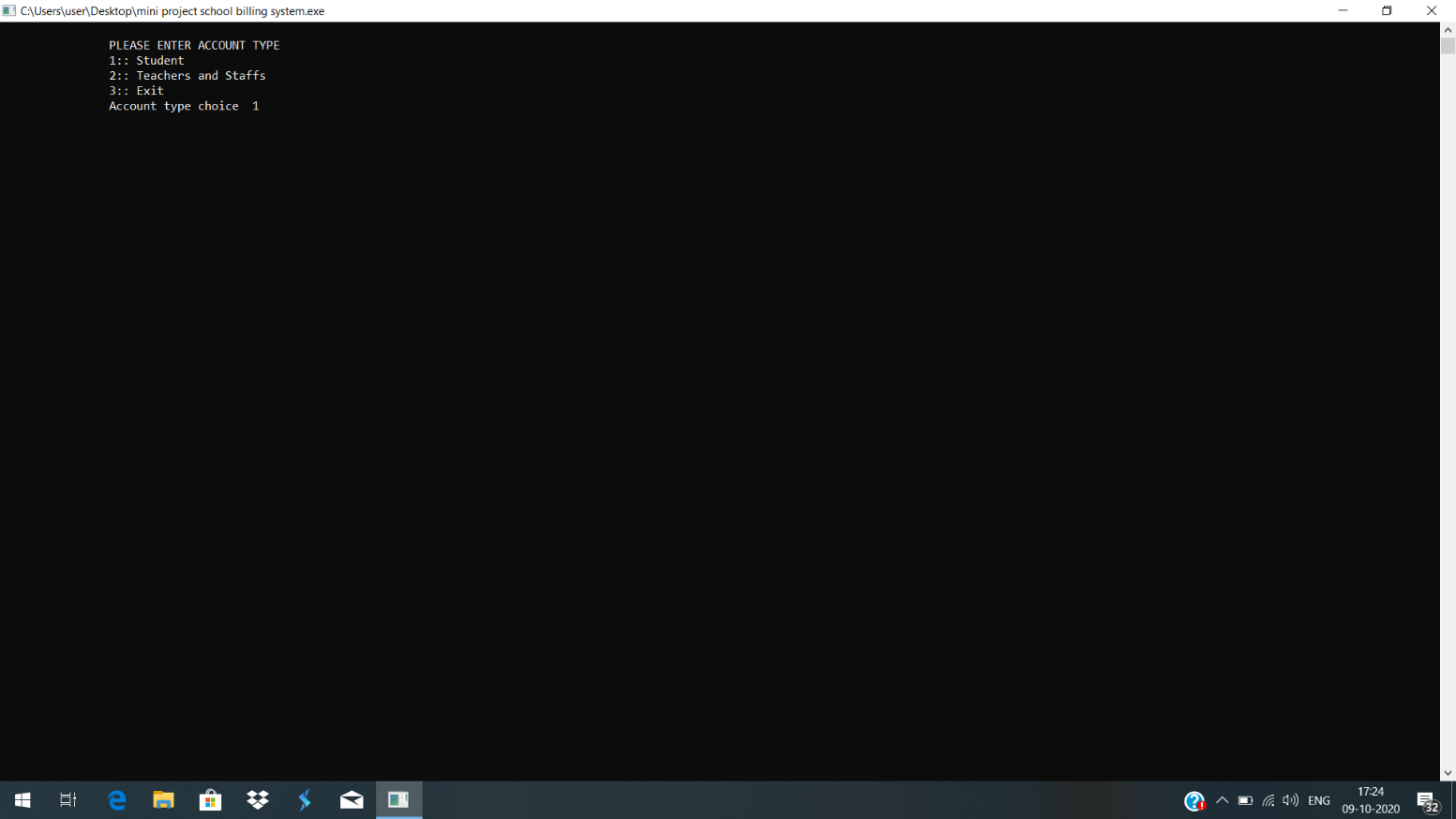
LOGIN PAGE:



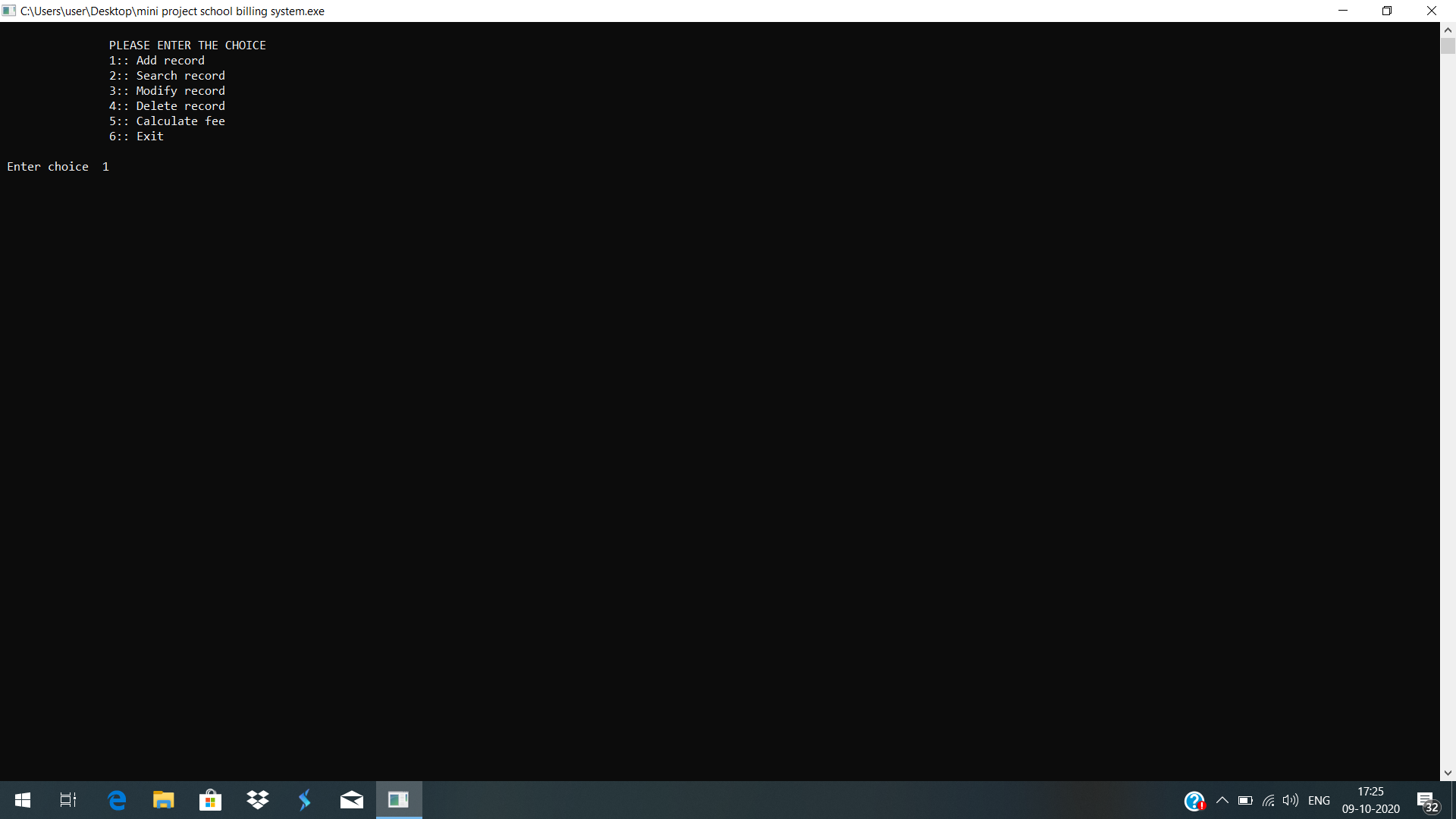
DATE ENTER:



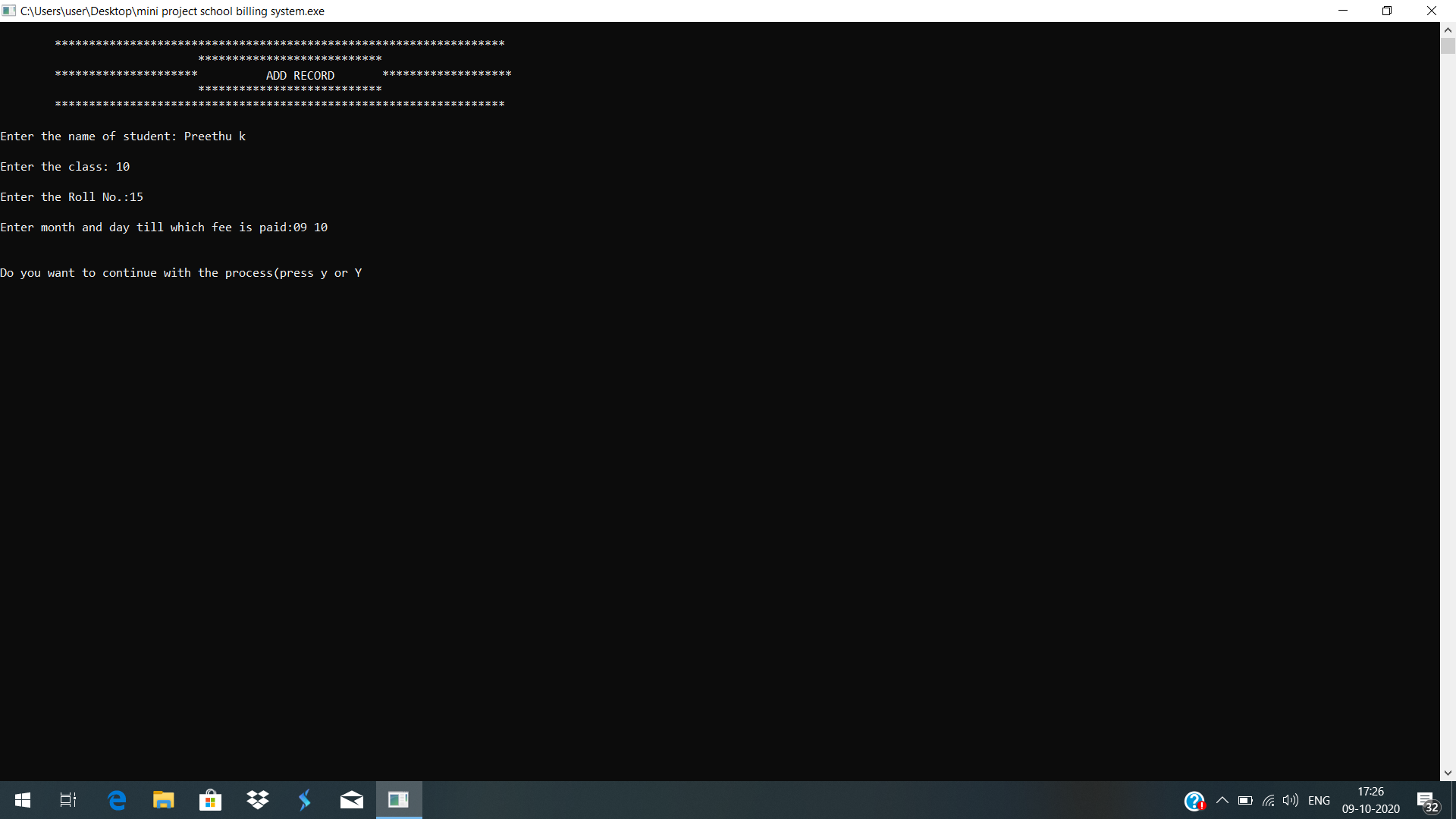
ACCOUNT TYPE:



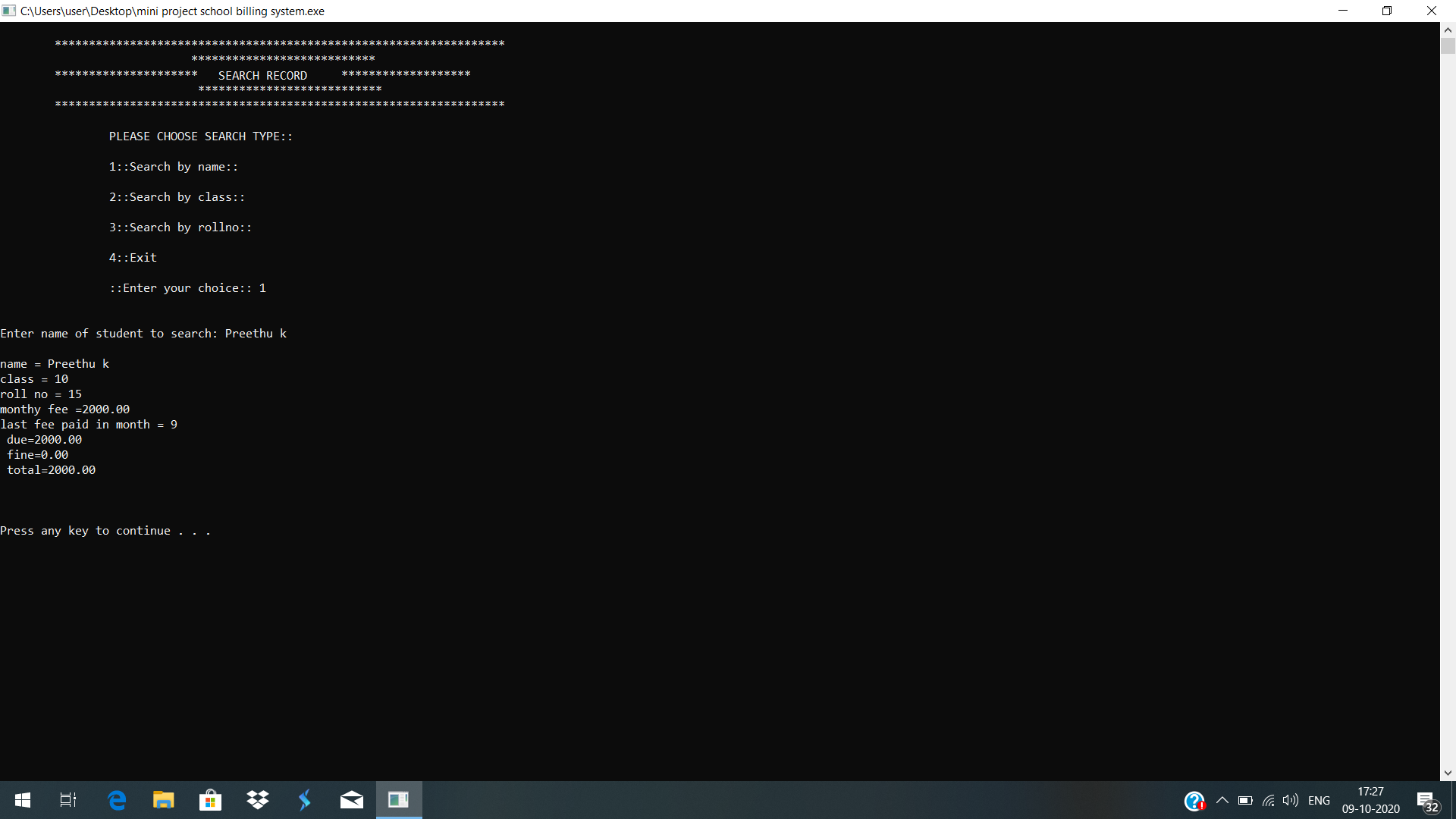
CHOICE ENTER:



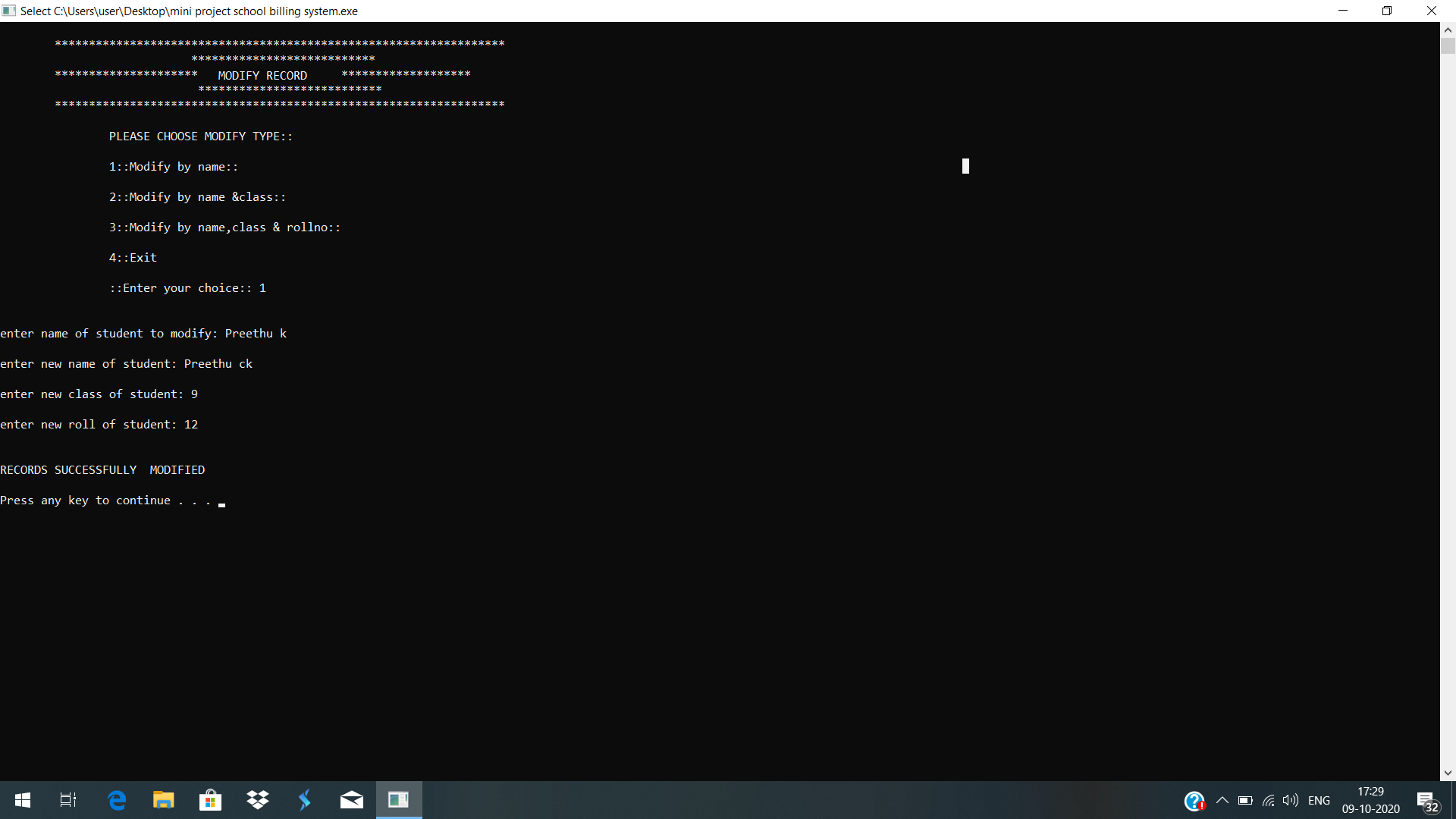
RECORD ADDITION:



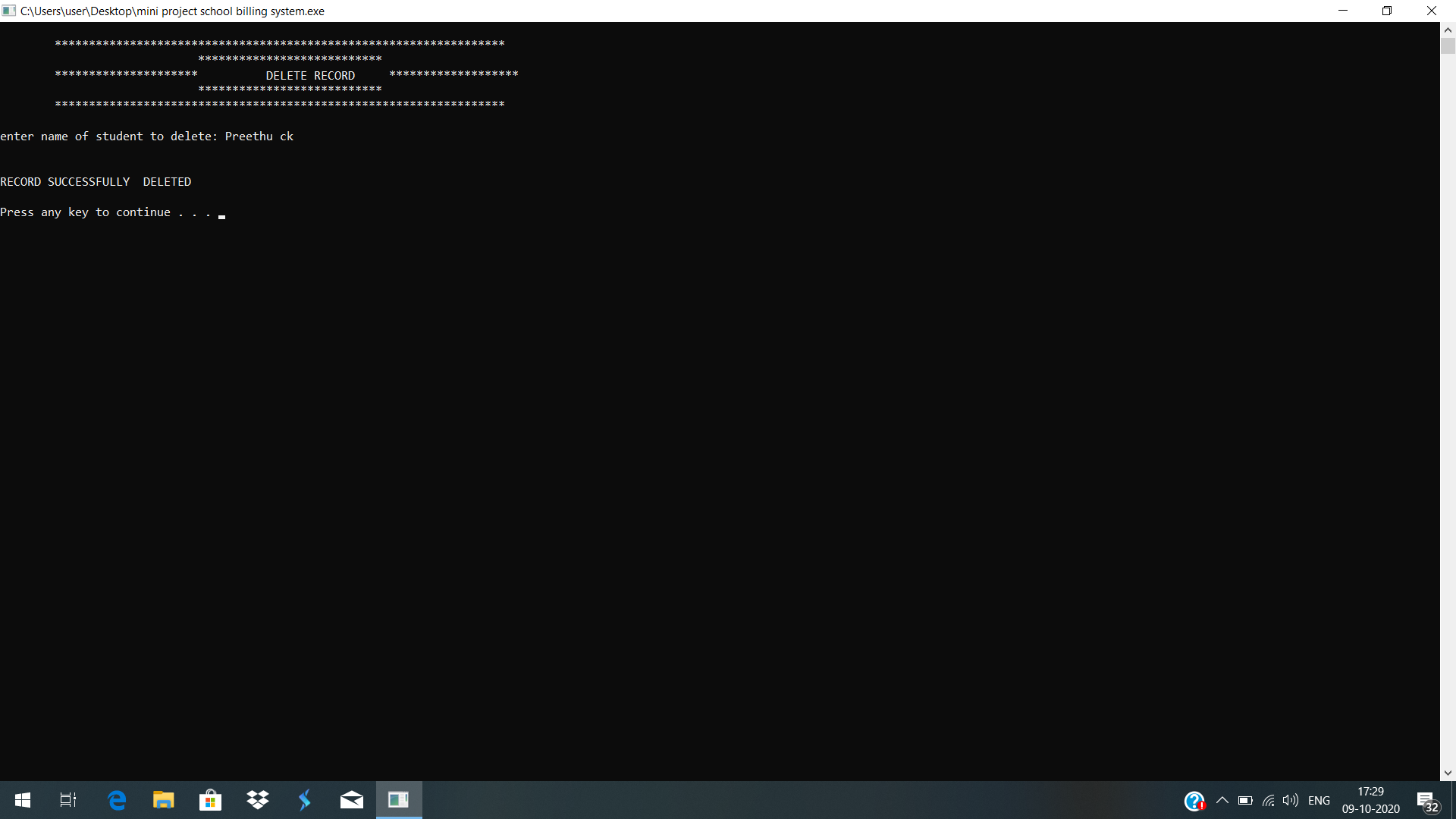
RECORD SEARCH:



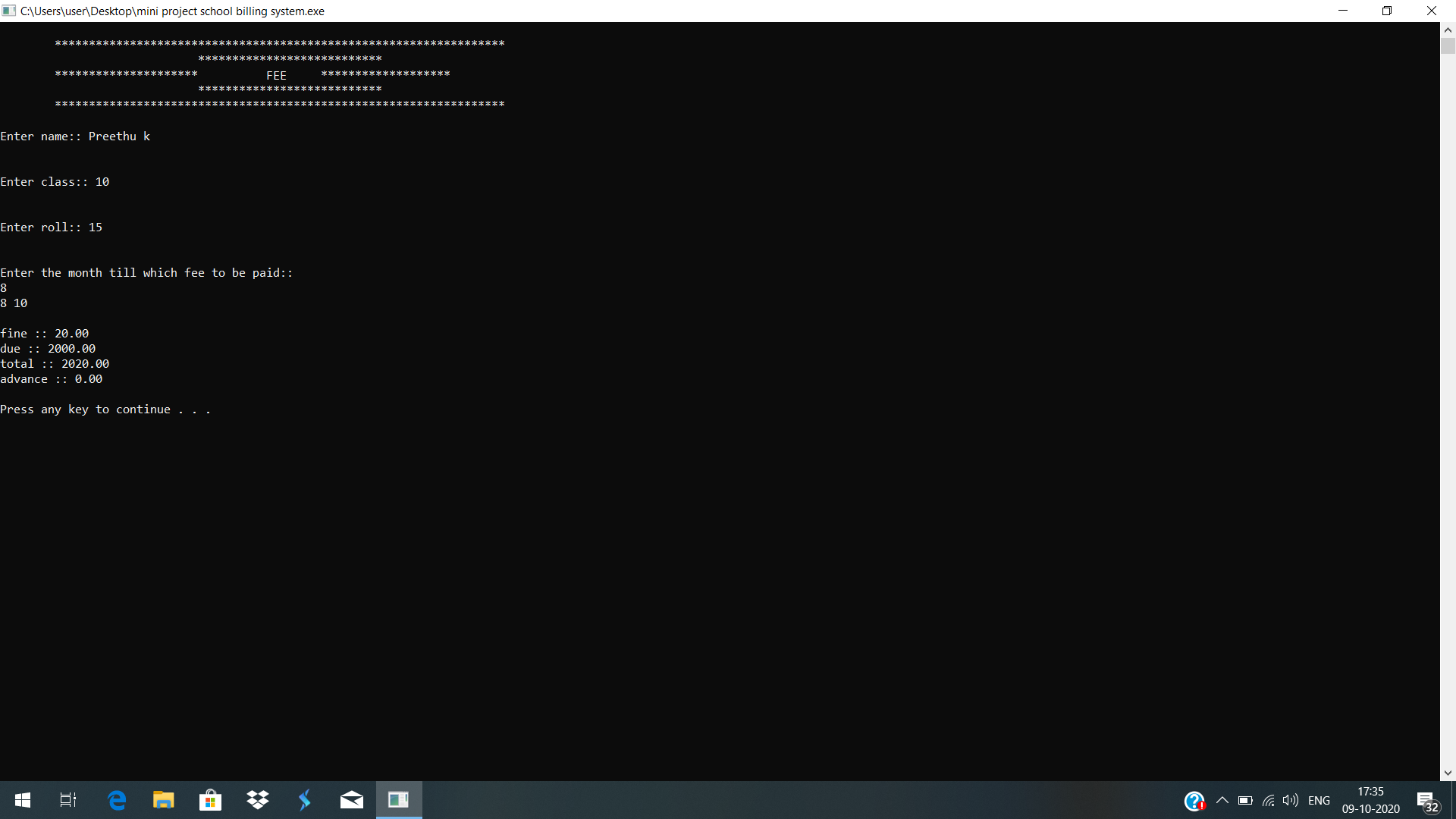
RECORD MODIFICATION:



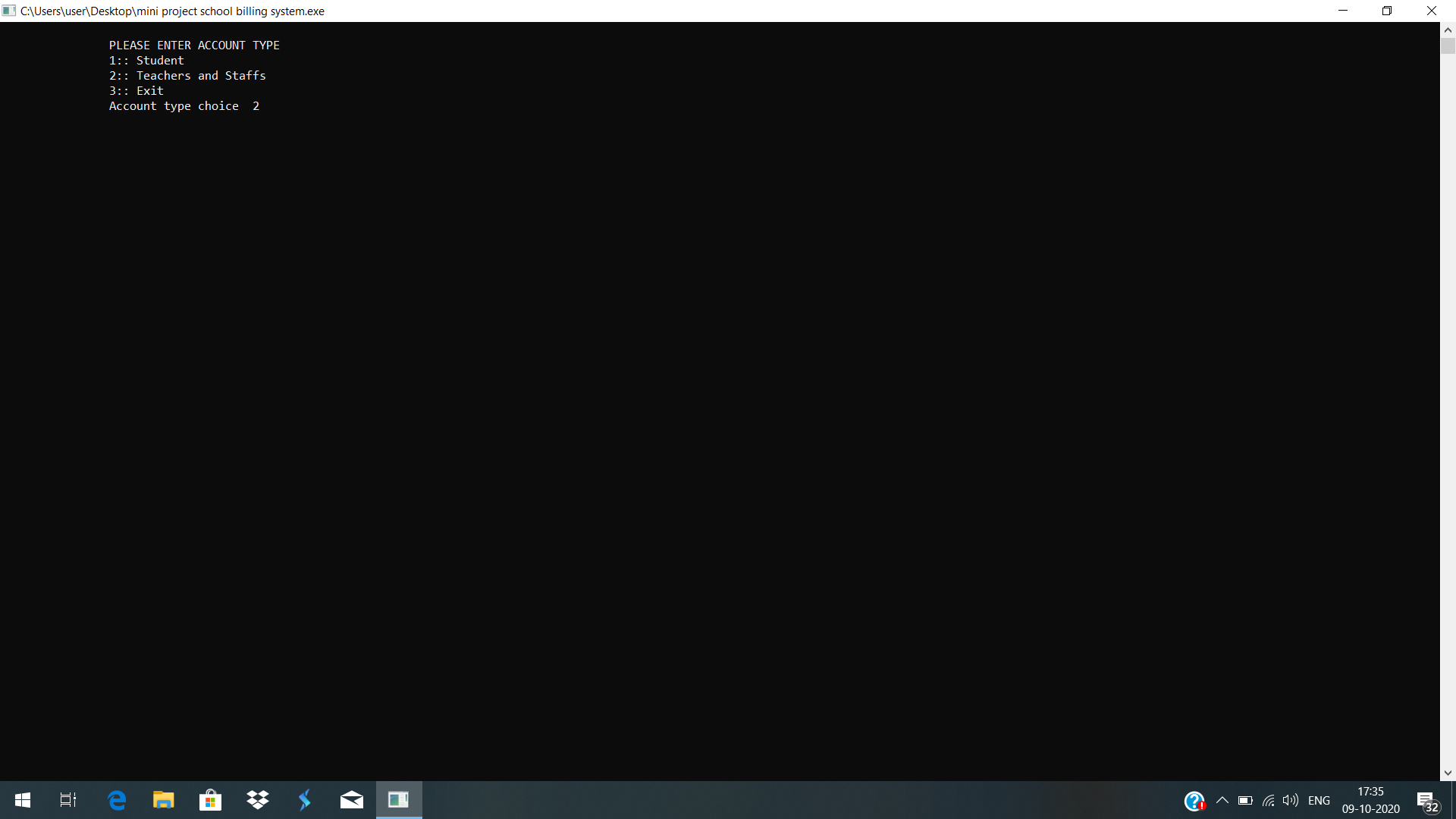
RECORD DELETION:



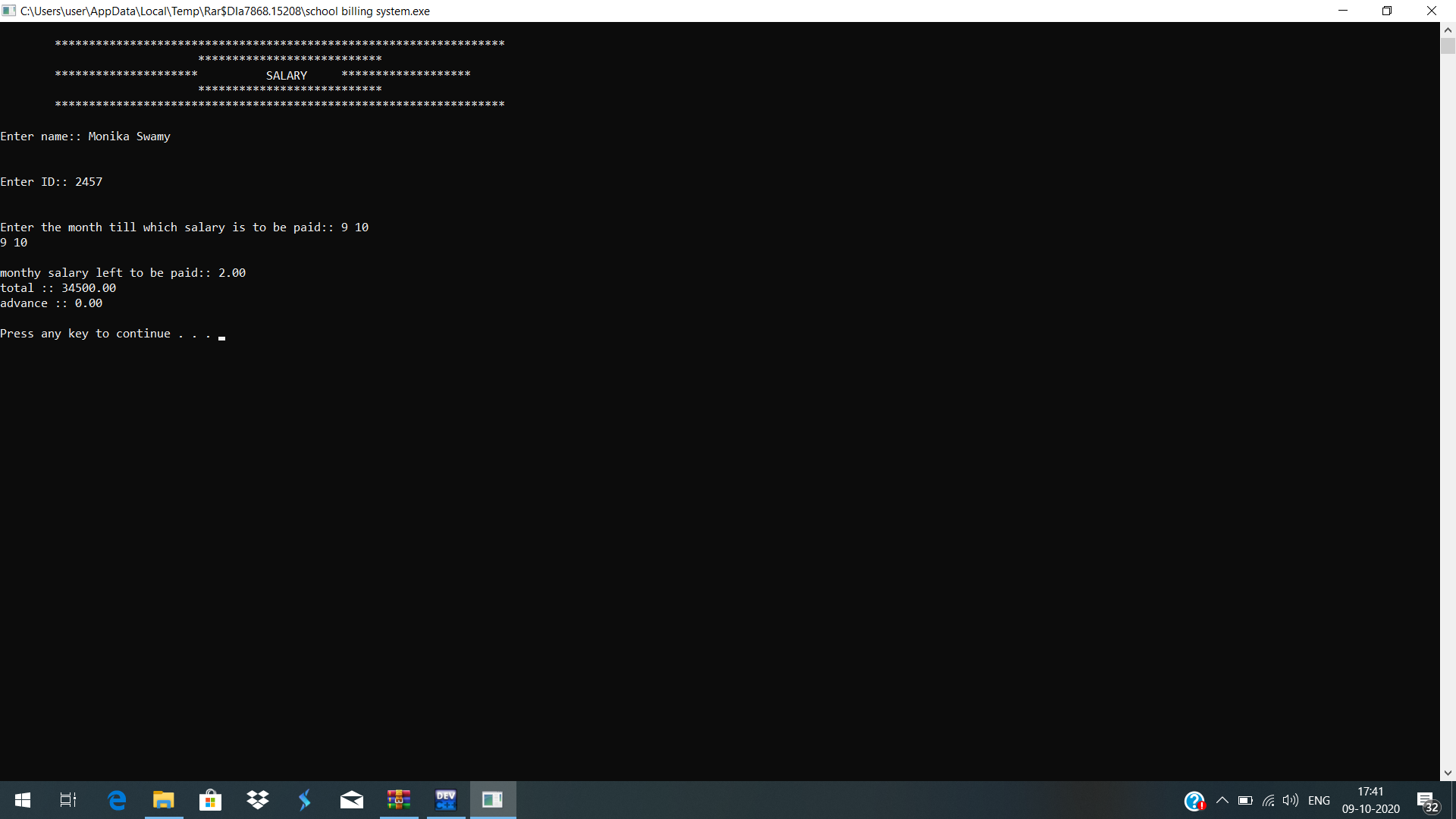
FEE CALCULATION:



ACCOUNT TYPE:



SALARY CALCULATION:



EXIT PAGE:

