Project 2

School Management System

Eng - Belal Hani Abu Sabha

Profile:

https://www.learn-in-depth-store.com/account/applystar

Github:

https://github.com/belalsabha/Master_Embedded_Systems

Linkedin:

https://www.linkedin.com/in/belal-sabha-06ba33349/

Introduction:

This program implements a Student Management System in C programming language using a FIFO (First-In-First-Out) buffer data structure to store and manage student records It gives a simple interface that allow to user to make   operations on student data.

Each student's information is represented using a structured data type (`S_Std_Data`) which stores their first name, last name, unique roll number, GPA, and registered course IDs . The program put addresses of these records in a fixed-size FIFO buffer (`FIFO_BUF`) that supports sequential addition .

Implementation:

1- Define a structure to represent the details of each student.

2-Declare an array of structures to serve as a buffer to store each structure of student with size 50 .

3-Define a control structure to manage the buffer and using pointers and counter variable to store number of students .

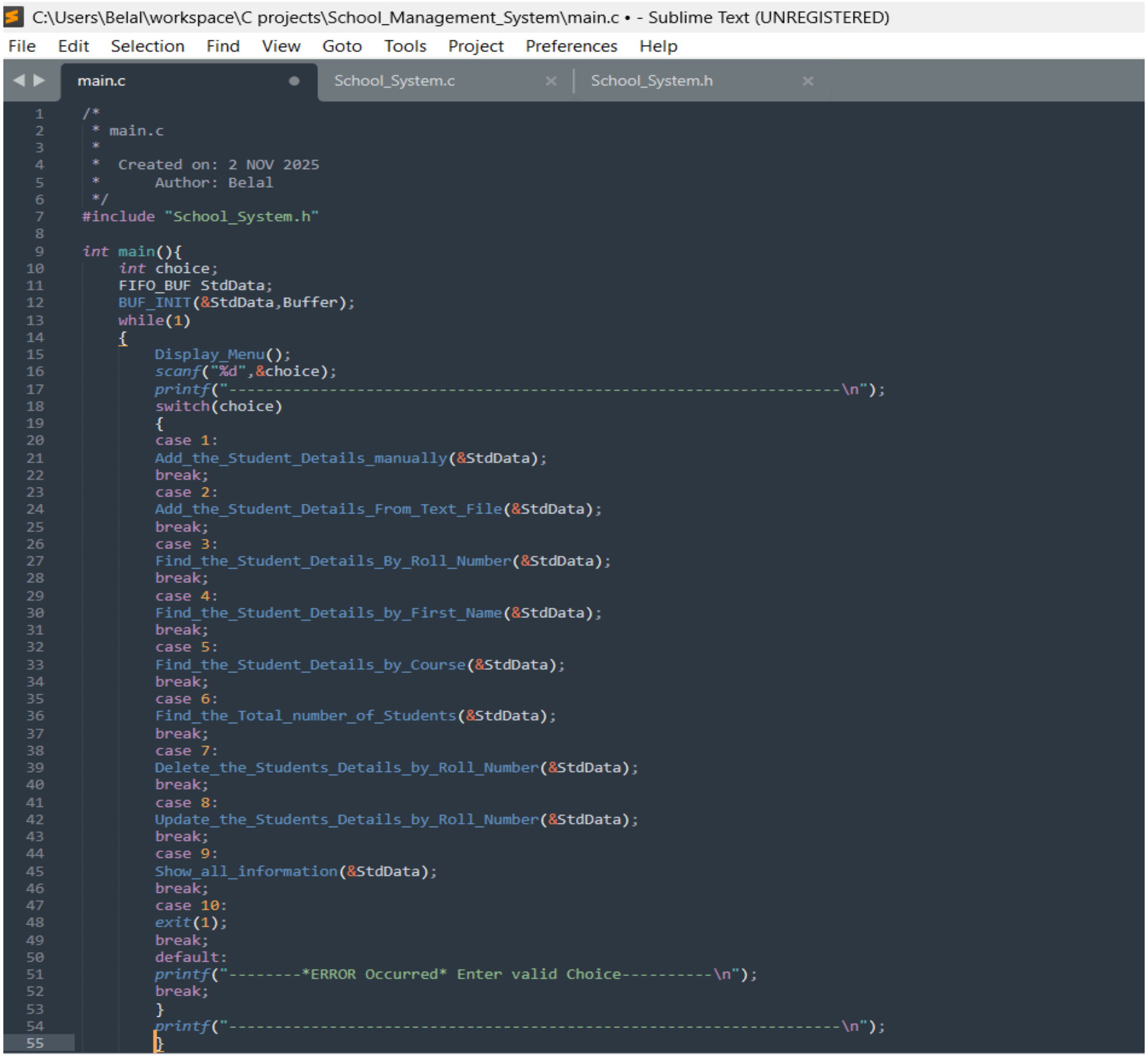
4- Implement functions that perform all required operations .

5- Define Enum to represent status of each Function .

Code:

Main.c:

In main.c we initialize student database then displays a menu that allows user to perform different operations.

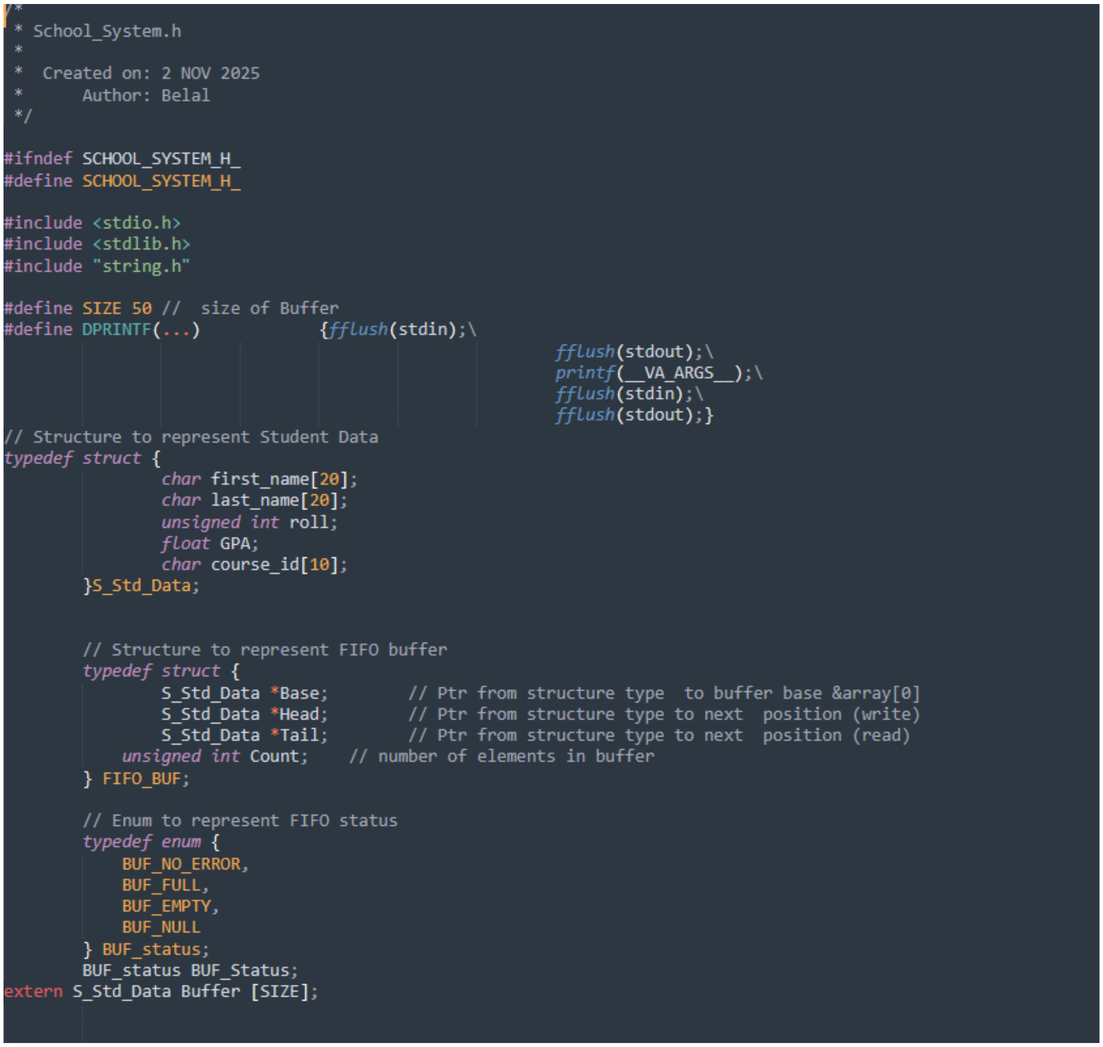


C:\Users\Belal\workspace\C projects\School_Management_System\main.c • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
main.c
School_System.c
School_System.h

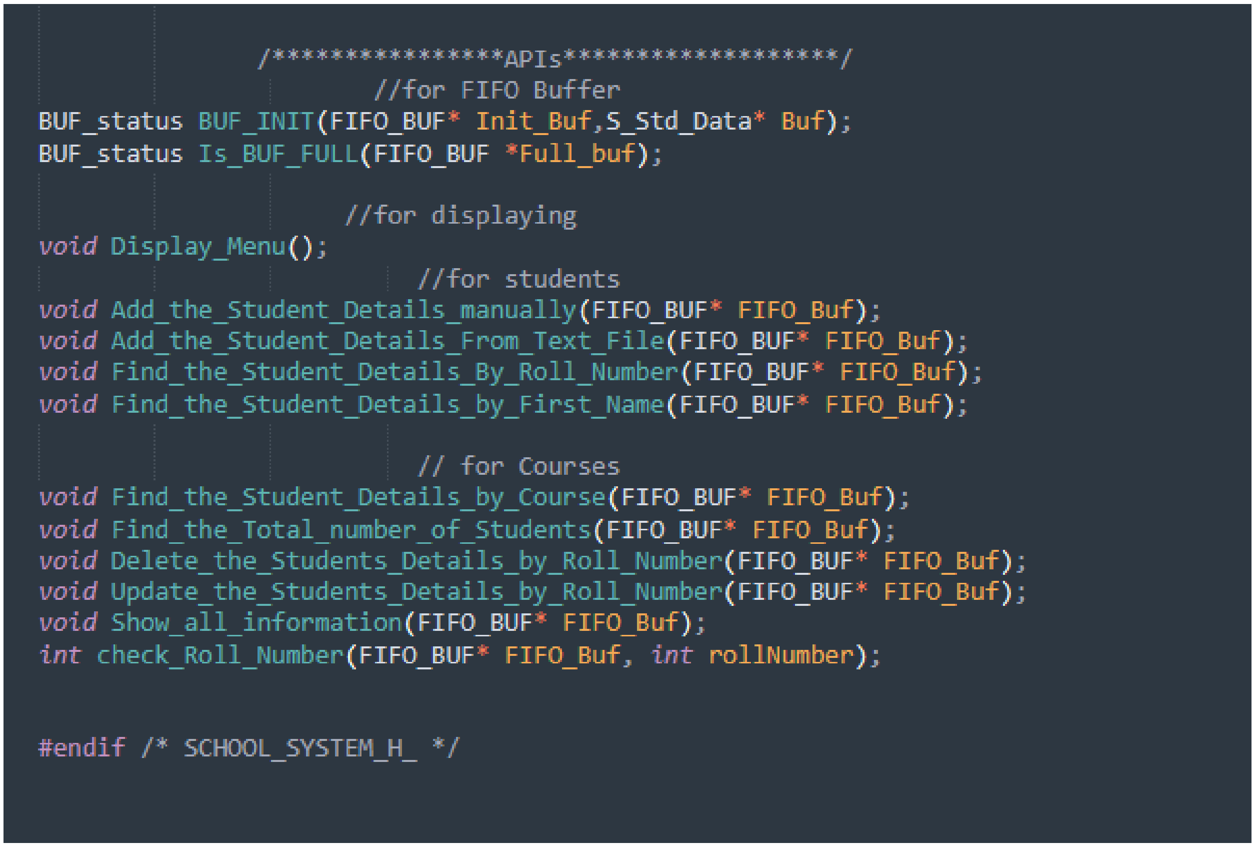
```c
/*
 * main.c
 *
 *  Created on: 2 NOV 2025
 *      Author: Belal
 */
#include "School_System.h"

int main(){
    int choice;
    FIFO_BUF StdData;
    BUF_INIT(&StdData,Buffer);
    while(1)
    {
        Display_Menu();
        scanf("%d",&choice);
        printf("-------------------------------------------------------------------\n");
        switch(choice)
        {
        case 1:
        Add_the_Student_Details_manually(&StdData);
        break;
        case 2:
        Add_the_Student_Details_From_Text_File(&StdData);
        break;
        case 3:
        Find_the_Student_Details_By_Roll_Number(&StdData);
        break;
        case 4:
        Find_the_Student_Details_by_First_Name(&StdData);
        break;
        case 5:
        Find_the_Student_Details_by_Course(&StdData);
        break;
        case 6:
        Find_the_Total_number_of_Students(&StdData);
        break;
        case 7:
        Delete_the_Students_Details_by_Roll_Number(&StdData);
        break;
        case 8:
        Update_the_Students_Details_by_Roll_Number(&StdData);
        break;
        case 9:
        Show_all_information(&StdData);
        break;
        case 10:
        exit(1);
        break;
        default:
        printf("--------*ERROR Occurred* Enter valid Choice----------\n");
        break;
        }
        printf("-------------------------------------------------------------------\n");
        }
```

School_System.h:

This header file defines a structure to represent student data, a control structure to manage the buffer, an enumeration to represent the status of each function, and finally, prototypes of all system functions.



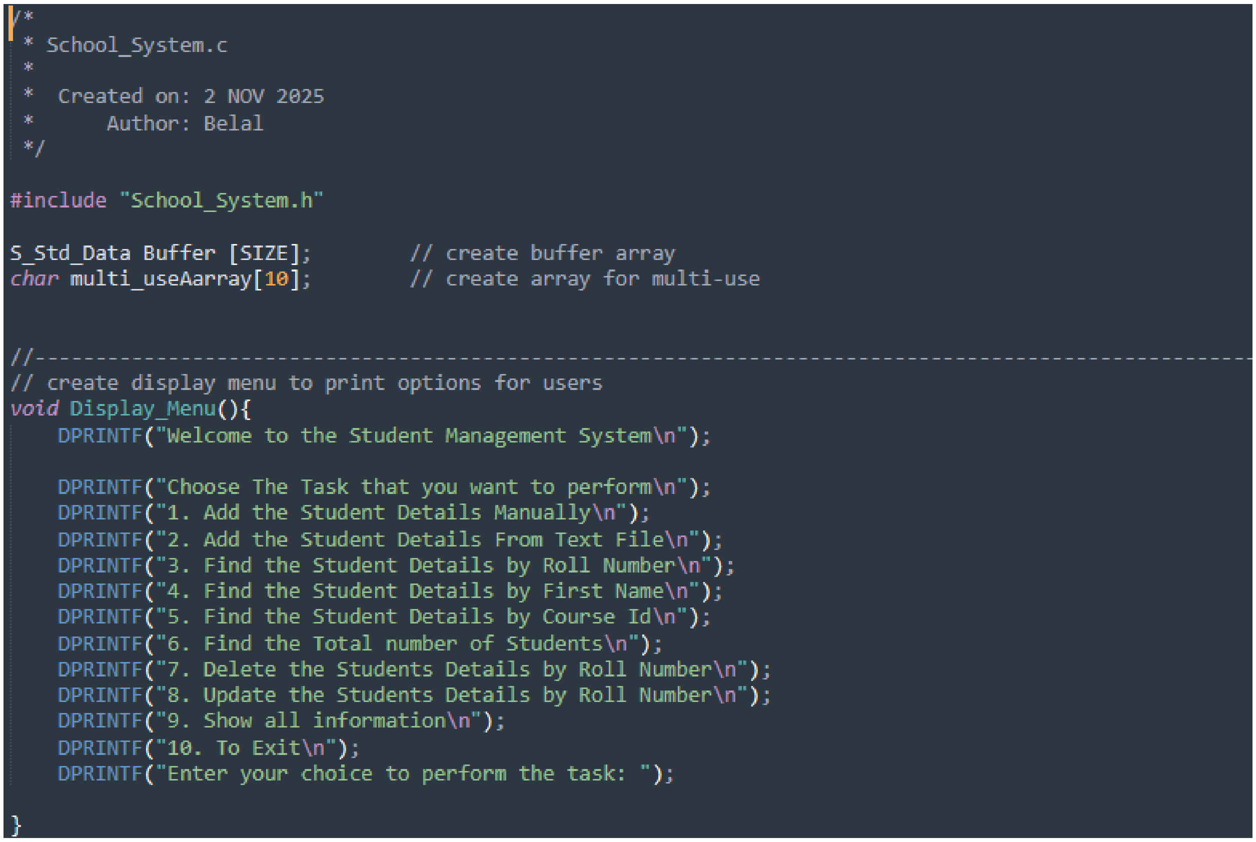
```c
/*
 * School_System.h
 *
 *  Created on: 2 NOV 2025
 *      Author: Belal
 */

#ifndef SCHOOL_SYSTEM_H_
#define SCHOOL_SYSTEM_H_

#include <stdio.h>
#include <stdlib.h>
#include "string.h"

#define SIZE 50 //  size of Buffer
#define DPRINTF(...)            {fflush(stdin);\
                                                        fflush(stdout);\
                                                        printf(__VA_ARGS__);\
                                                        fflush(stdin);\
                                                        fflush(stdout);}
// Structure to represent Student Data
typedef struct {
                char first_name[20];
                char last_name[20];
                unsigned int roll;
                float GPA;
                char course_id[10];
        }S_Std_Data;


        // Structure to represent FIFO buffer
        typedef struct {
                S_Std_Data *Base;        // Ptr from structure type  to buffer base &array[0]
                S_Std_Data *Head;        // Ptr from structure type to next  position (write)
                S_Std_Data *Tail;        // Ptr from structure type to next  position (read)
            unsigned int Count;    // number of elements in buffer
        } FIFO_BUF;

        // Enum to represent FIFO status
        typedef enum {
            BUF_NO_ERROR,
            BUF_FULL,
            BUF_EMPTY,
            BUF_NULL
        } BUF_status;
        BUF_status BUF_Status;
extern S_Std_Data Buffer [SIZE];
```



```c
               /****************APIs*******************/
                       //for FIFO Buffer
BUF_status BUF_INIT(FIFO_BUF* Init_Buf,S_Std_Data* Buf);
BUF_status Is_BUF_FULL(FIFO_BUF *Full_buf);

                     //for displaying
void Display_Menu();
                          //for students
void Add_the_Student_Details_manually(FIFO_BUF* FIFO_Buf);
void Add_the_Student_Details_From_Text_File(FIFO_BUF* FIFO_Buf);
void Find_the_Student_Details_By_Roll_Number(FIFO_BUF* FIFO_Buf);
void Find_the_Student_Details_by_First_Name(FIFO_BUF* FIFO_Buf);

                          // for Courses
void Find_the_Student_Details_by_Course(FIFO_BUF* FIFO_Buf);
void Find_the_Total_number_of_Students(FIFO_BUF* FIFO_Buf);
void Delete_the_Students_Details_by_Roll_Number(FIFO_BUF* FIFO_Buf);
void Update_the_Students_Details_by_Roll_Number(FIFO_BUF* FIFO_Buf);
void Show_all_information(FIFO_BUF* FIFO_Buf);
int check_Roll_Number(FIFO_BUF* FIFO_Buf, int rollNumber);


#endif /* SCHOOL_SYSTEM_H_ */
```

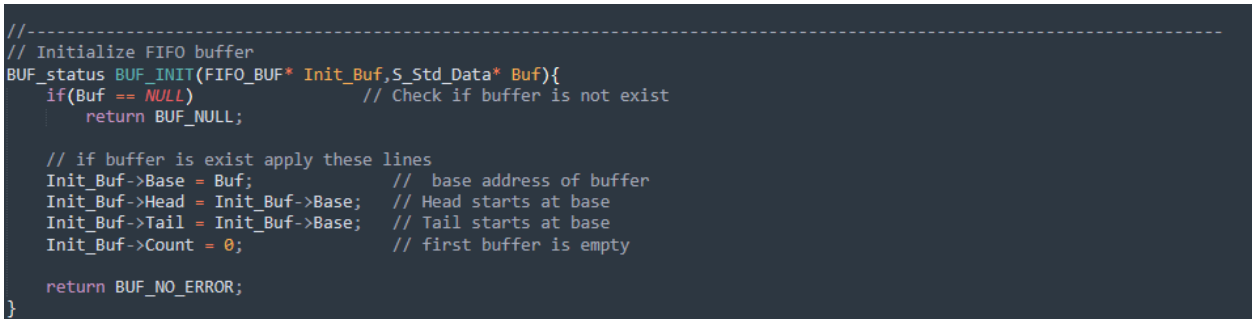
School_System.c:

This function displays a menu that contains all operations in Student Management System , and define two arrays the first act as buffer and the second for using in different situations in code .



```c
/*
 * School_System.c
 *
 *  Created on: 2 NOV 2025
 *      Author: Belal
 */

#include "School_System.h"

S_Std_Data Buffer [SIZE];        // create buffer array
char multi_useAarray[10];        // create array for multi-use


//-----------------------------------------------------------------------------------------------------------
// create display menu to print options for users
void Display_Menu(){
    DPRINTF("Welcome to the Student Management System\n");

    DPRINTF("Choose The Task that you want to perform\n");
    DPRINTF("1. Add the Student Details Manually\n");
    DPRINTF("2. Add the Student Details From Text File\n");
    DPRINTF("3. Find the Student Details by Roll Number\n");
    DPRINTF("4. Find the Student Details by First Name\n");
    DPRINTF("5. Find the Student Details by Course Id\n");
    DPRINTF("6. Find the Total number of Students\n");
    DPRINTF("7. Delete the Students Details by Roll Number\n");
    DPRINTF("8. Update the Students Details by Roll Number\n");
    DPRINTF("9. Show all information\n");
    DPRINTF("10. To Exit\n");
    DPRINTF("Enter your choice to perform the task: ");

}
```

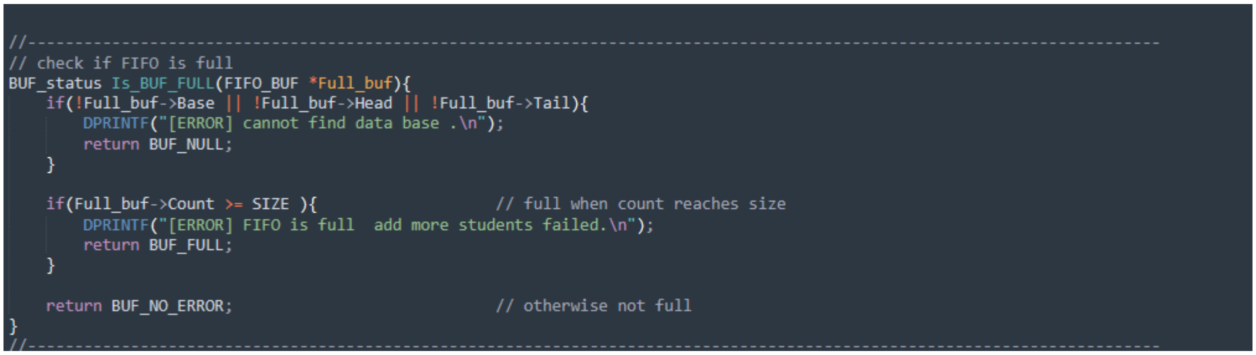
This function initializes the FIFO buffer by making its base, head, and tail pointers at the start of the buffer and the count to zero.



```c
//-------------------------------------------------------------------------------------------------------------------------
// Initialize FIFO buffer
BUF_status BUF_INIT(FIFO_BUF* Init_Buf,S_Std_Data* Buf){
    if(Buf == NULL)                     // Check if buffer is not exist
        return BUF_NULL;

    // if buffer is exist apply these lines
    Init_Buf->Base = Buf;              //  base address of buffer
    Init_Buf->Head = Init_Buf->Base;   // Head starts at base
    Init_Buf->Tail = Init_Buf->Base;   // Tail starts at base
    Init_Buf->Count = 0;               // first buffer is empty

    return BUF_NO_ERROR;
}
```

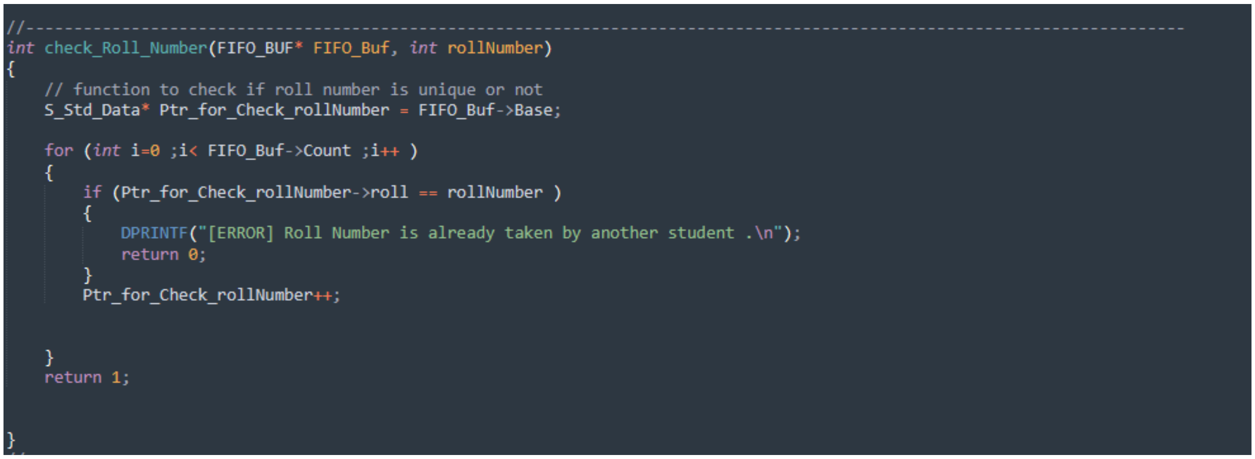
—---

This function checks if the FIFO buffer is full or invalid , returning an status message if it cannot accept more student records or cannot find data base.



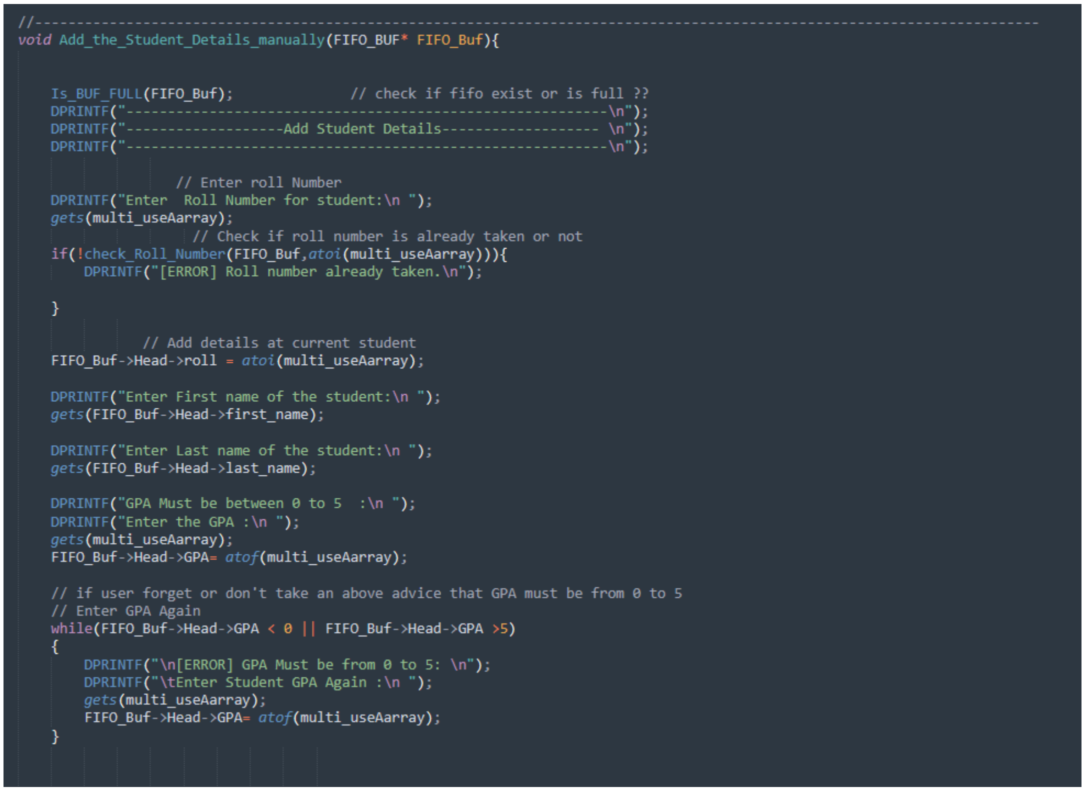
```c
//-------------------------------------------------------------------------------------------------------------------------
// check if FIFO is full
BUF_status Is_BUF_FULL(FIFO_BUF *Full_buf){
    if(!Full_buf->Base || !Full_buf->Head || !Full_buf->Tail){
        DPRINTF("[ERROR] cannot find data base .\n");
        return BUF_NULL;
    }

    if(Full_buf->Count >= SIZE ){                   // full when count reaches size
        DPRINTF("[ERROR] FIFO is full  add more students failed.\n");
        return BUF_FULL;
    }

    return BUF_NO_ERROR;                            // otherwise not full
}
//-------------------------------------------------------------------------------------------------------------------------
```

This function check if a given roll number already exists in the buffer and each student has a unique roll number.

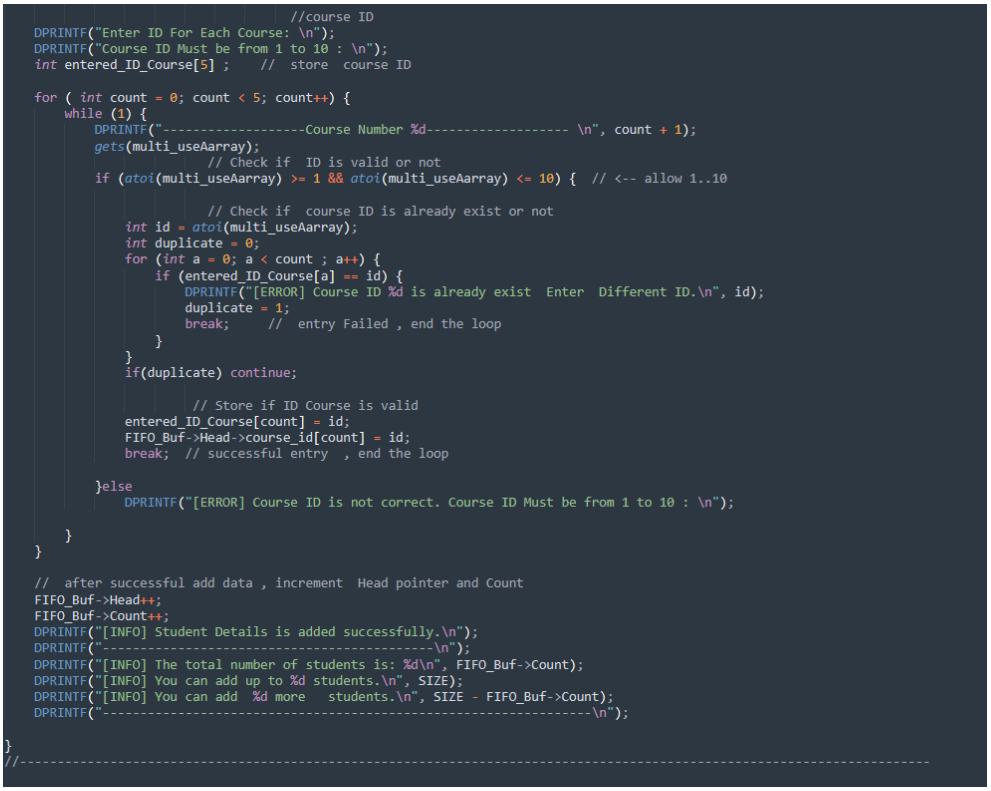


```c
//-------------------------------------------------------------------------------------------------------------
int check_Roll_Number(FIFO_BUF* FIFO_Buf, int rollNumber)
{
    // function to check if roll number is unique or not
    S_Std_Data* Ptr_for_Check_rollNumber = FIFO_Buf->Base;

    for (int i=0 ;i< FIFO_Buf->Count ;i++ )
    {
        if (Ptr_for_Check_rollNumber->roll == rollNumber )
        {
            DPRINTF("[ERROR] Roll Number is already taken by another student .\n");
            return 0;
        }
        Ptr_for_Check_rollNumber++;


    }
    return 1;


}
```

This function allows the user to add manually a new student to the FIFO buffer by entering the roll number, first and last names, GPA, and course IDs, while ensuring the GPA valid range and course IDs are not duplicated .

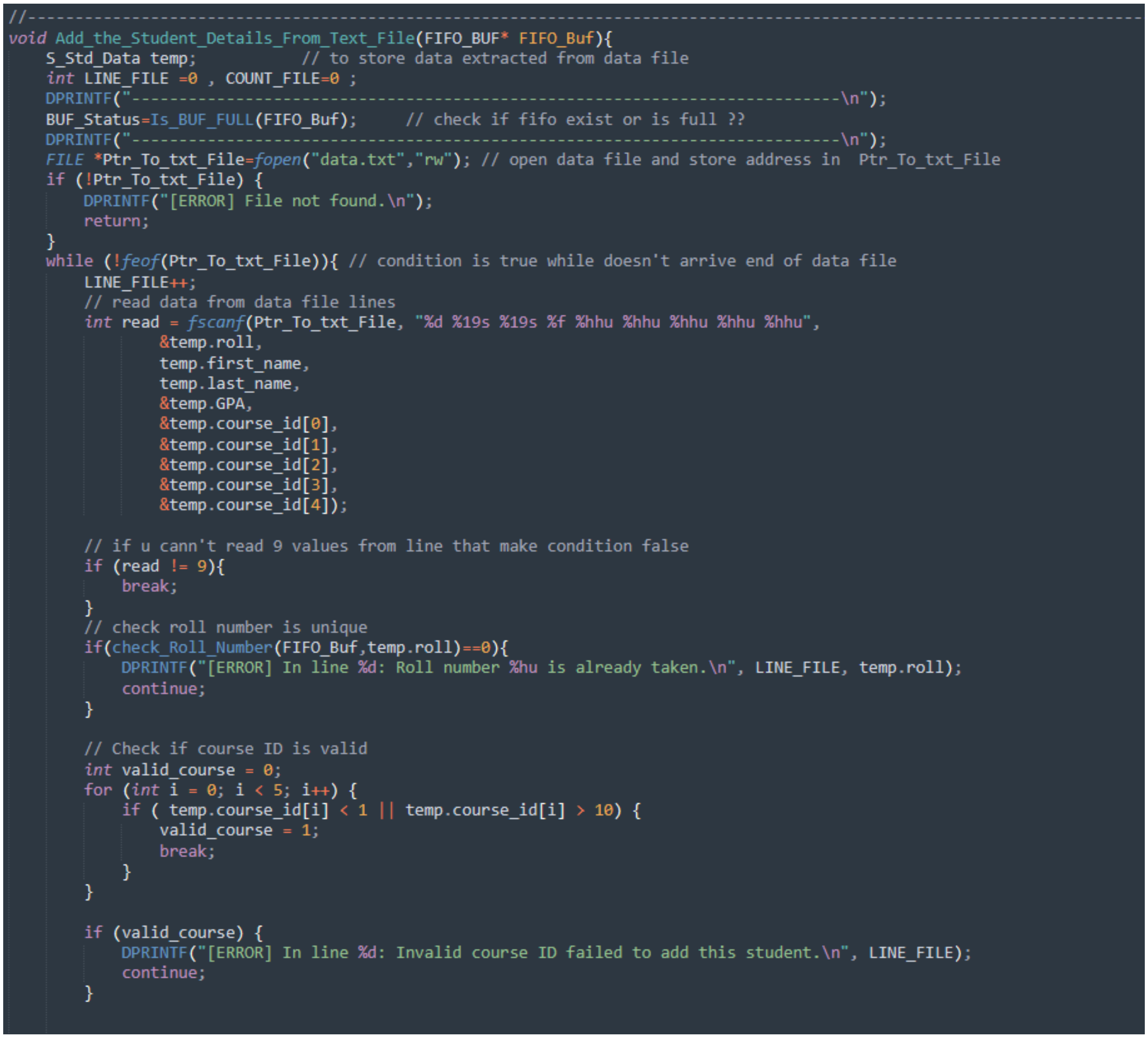


```c
//-----------------------------------------------------------------------------------------------------------------------
void Add_the_Student_Details_manually(FIFO_BUF* FIFO_Buf){


    Is_BUF_FULL(FIFO_Buf);              // check if fifo exist or is full ??
    DPRINTF("---------------------------------------------------------\n");
    DPRINTF("------------------Add Student Details------------------- \n");
    DPRINTF("---------------------------------------------------------\n");

                   // Enter roll Number
    DPRINTF("Enter  Roll Number for student:\n ");
    gets(multi_useAarray);
                     // Check if roll number is already taken or not
    if(!check_Roll_Number(FIFO_Buf,atoi(multi_useAarray))){
        DPRINTF("[ERROR] Roll number already taken.\n");

    }

               // Add details at current student
    FIFO_Buf->Head->roll = atoi(multi_useAarray);

    DPRINTF("Enter First name of the student:\n ");
    gets(FIFO_Buf->Head->first_name);

    DPRINTF("Enter Last name of the student:\n ");
    gets(FIFO_Buf->Head->last_name);

    DPRINTF("GPA Must be between 0 to 5  :\n ");
    DPRINTF("Enter the GPA :\n ");
    gets(multi_useAarray);
    FIFO_Buf->Head->GPA= atof(multi_useAarray);

    // if user forget or don't take an above advice that GPA must be from 0 to 5
    // Enter GPA Again
    while(FIFO_Buf->Head->GPA < 0 || FIFO_Buf->Head->GPA >5)
    {
        DPRINTF("\n[ERROR] GPA Must be from 0 to 5: \n");
        DPRINTF("\tEnter Student GPA Again :\n ");
        gets(multi_useAarray);
        FIFO_Buf->Head->GPA= atof(multi_useAarray);
    }
```

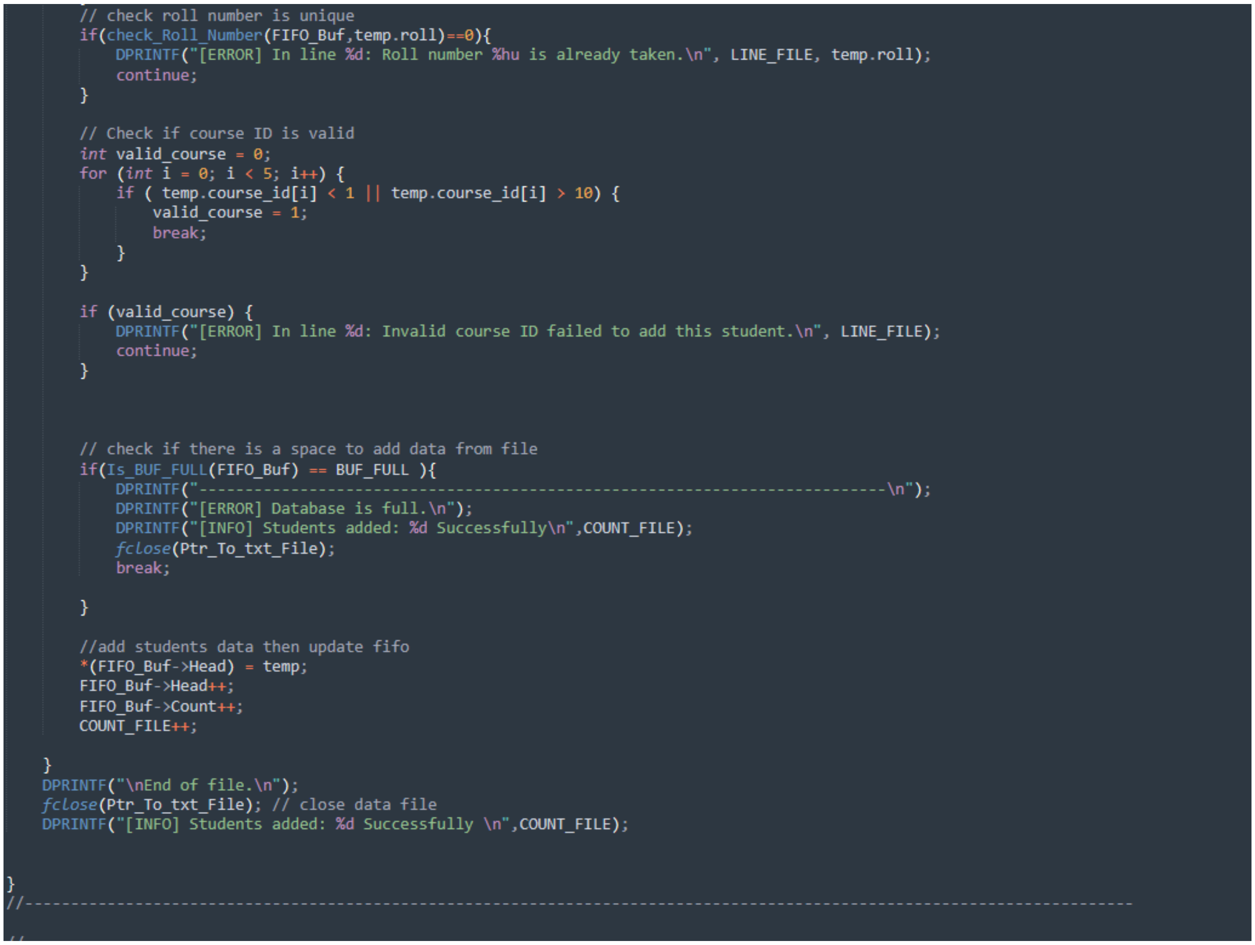


```c
                                     //course ID
    DPRINTF("Enter ID For Each Course: \n");
    DPRINTF("Course ID Must be from 1 to 10 : \n");
    int entered_ID_Course[5] ;    //  store  course ID

    for ( int count = 0; count < 5; count++) {
        while (1) {
            DPRINTF("------------------Course Number %d------------------ \n", count + 1);
            gets(multi_useAarray);
                           // Check if  ID is valid or not
            if (atoi(multi_useAarray) >= 1 && atoi(multi_useAarray) <= 10) {  // <-- allow 1..10

                           // Check if  course ID is already exist or not
                int id = atoi(multi_useAarray);
                int duplicate = 0;
                for (int a = 0; a < count ; a++) {
                    if (entered_ID_Course[a] == id) {
                        DPRINTF("[ERROR] Course ID %d is already exist  Enter  Different ID.\n", id);
                        duplicate = 1;
                        break;     //  entry Failed , end the loop
                    }
                }
                if(duplicate) continue;

                         // Store if ID Course is valid
                entered_ID_Course[count] = id;
                FIFO_Buf->Head->course_id[count] = id;
                break;  // successful entry  , end the loop

            }else
                DPRINTF("[ERROR] Course ID is not correct. Course ID Must be from 1 to 10 : \n");

        }
    }

    //  after successful add data , increment  Head pointer and Count
    FIFO_Buf->Head++;
    FIFO_Buf->Count++;
    DPRINTF("[INFO] Student Details is added successfully.\n");
    DPRINTF("--------------------------------------------\n");
    DPRINTF("[INFO] The total number of students is: %d\n", FIFO_Buf->Count);
    DPRINTF("[INFO] You can add up to %d students.\n", SIZE);
    DPRINTF("[INFO] You can add  %d more   students.\n", SIZE - FIFO_Buf->Count);
    DPRINTF("-----------------------------------------------------------------\n");

}
//-----------------------------------------------------------------------------------------------------------------------
```

This function reads student records from a text file and adds them to the FIFO buffer , check if roll numbers are unique , course IDs and buffer space while counting successfully added entries .

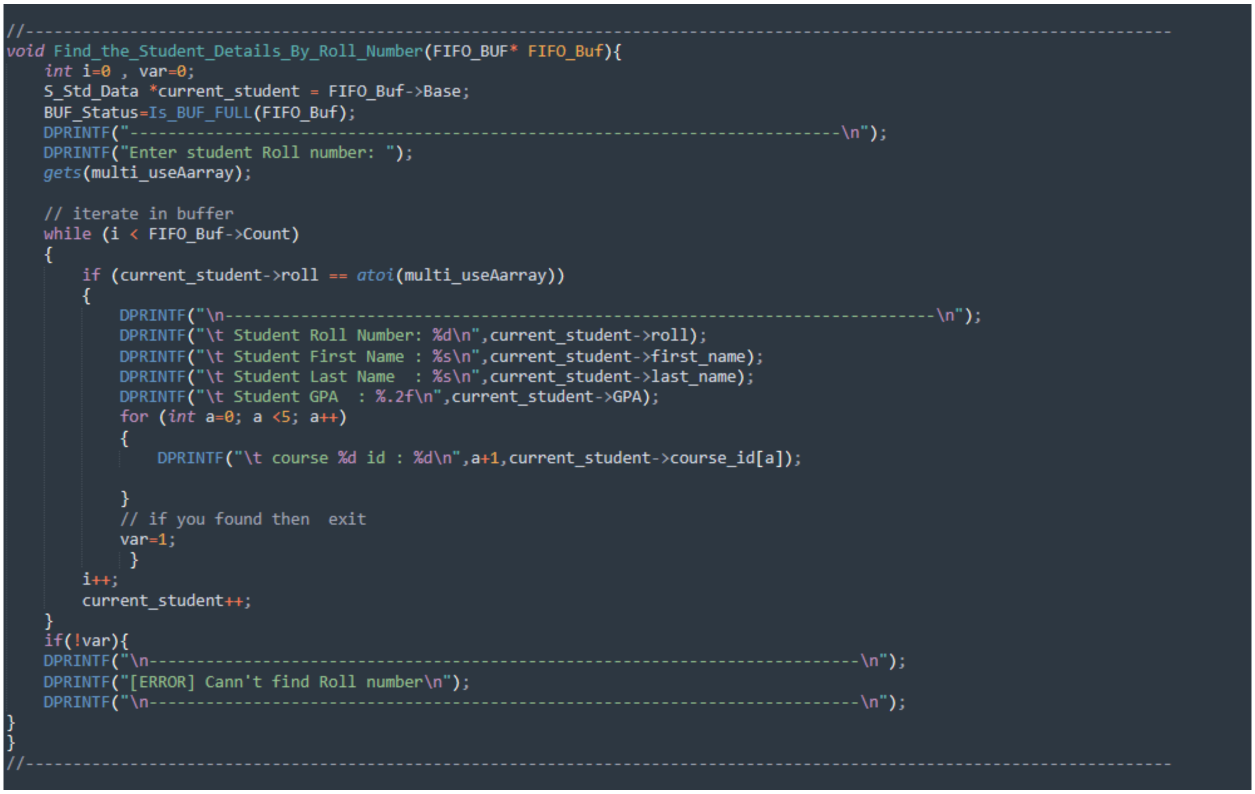


```c
//----------------------------------------------------------------------------------------------------------------
void Add_the_Student_Details_From_Text_File(FIFO_BUF* FIFO_Buf){
    S_Std_Data temp;           // to store data extracted from data file
    int LINE_FILE =0 , COUNT_FILE=0 ;
    DPRINTF("---------------------------------------------------------------------------\n");
    BUF_Status=Is_BUF_FULL(FIFO_Buf);     // check if fifo exist or is full ??
    DPRINTF("---------------------------------------------------------------------------\n");
    FILE *Ptr_To_txt_File=fopen("data.txt","rw"); // open data file and store address in  Ptr_To_txt_File
    if (!Ptr_To_txt_File) {
        DPRINTF("[ERROR] File not found.\n");
        return;
    }
    while (!feof(Ptr_To_txt_File)){ // condition is true while doesn't arrive end of data file
        LINE_FILE++;
        // read data from data file lines
        int read = fscanf(Ptr_To_txt_File, "%d %19s %19s %f %hhu %hhu %hhu %hhu %hhu",
                &temp.roll,
                temp.first_name,
                temp.last_name,
                &temp.GPA,
                &temp.course_id[0],
                &temp.course_id[1],
                &temp.course_id[2],
                &temp.course_id[3],
                &temp.course_id[4]);

        // if u cann't read 9 values from line that make condition false
        if (read != 9){
            break;
        }
        // check roll number is unique
        if(check_Roll_Number(FIFO_Buf,temp.roll)==0){
            DPRINTF("[ERROR] In line %d: Roll number %hu is already taken.\n", LINE_FILE, temp.roll);
            continue;
        }

        // Check if course ID is valid
        int valid_course = 0;
        for (int i = 0; i < 5; i++) {
            if ( temp.course_id[i] < 1 || temp.course_id[i] > 10) {
                valid_course = 1;
                break;
            }
        }

        if (valid_course) {
            DPRINTF("[ERROR] In line %d: Invalid course ID failed to add this student.\n", LINE_FILE);
            continue;
        }
```



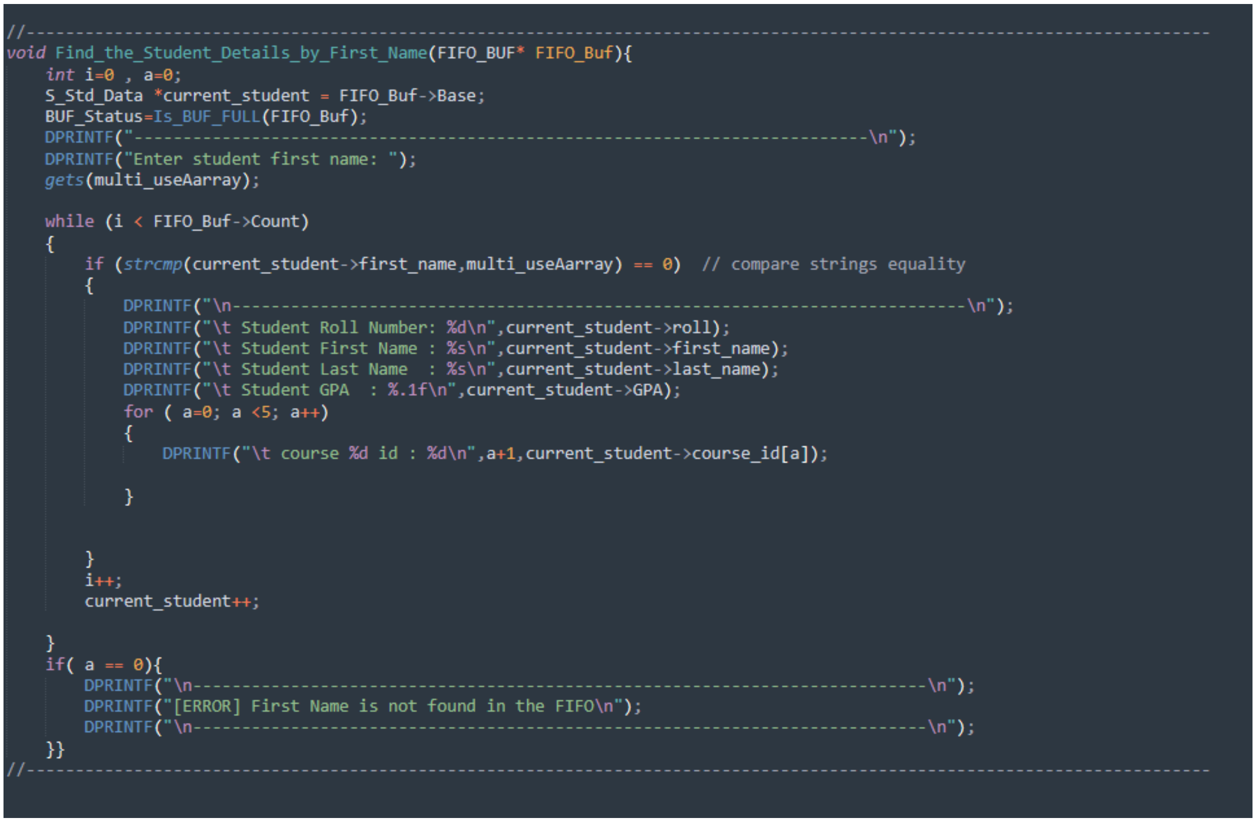
```c
        // check roll number is unique
        if(check_Roll_Number(FIFO_Buf,temp.roll)==0){
            DPRINTF("[ERROR] In line %d: Roll number %hu is already taken.\n", LINE_FILE, temp.roll);
            continue;
        }

        // Check if course ID is valid
        int valid_course = 0;
        for (int i = 0; i < 5; i++) {
            if ( temp.course_id[i] < 1 || temp.course_id[i] > 10) {
                valid_course = 1;
                break;
            }
        }

        if (valid_course) {
            DPRINTF("[ERROR] In line %d: Invalid course ID failed to add this student.\n", LINE_FILE);
            continue;
        }



        // check if there is a space to add data from file
        if(Is_BUF_FULL(FIFO_Buf) == BUF_FULL ){
            DPRINTF("---------------------------------------------------------------------------\n");
            DPRINTF("[ERROR] Database is full.\n");
            DPRINTF("[INFO] Students added: %d Successfully\n",COUNT_FILE);
            fclose(Ptr_To_txt_File);
            break;

        }

        //add students data then update fifo
        *(FIFO_Buf->Head) = temp;
        FIFO_Buf->Head++;
        FIFO_Buf->Count++;
        COUNT_FILE++;

    }
    DPRINTF("\nEnd of file.\n");
    fclose(Ptr_To_txt_File); // close data file
    DPRINTF("[INFO] Students added: %d Successfully \n",COUNT_FILE);


}
//-------------------------------------------------------------------------------------------------------------------------
```

This function searches in the FIFO buffer for a student with specified roll number and displays their details if found , otherwise displays an error message.



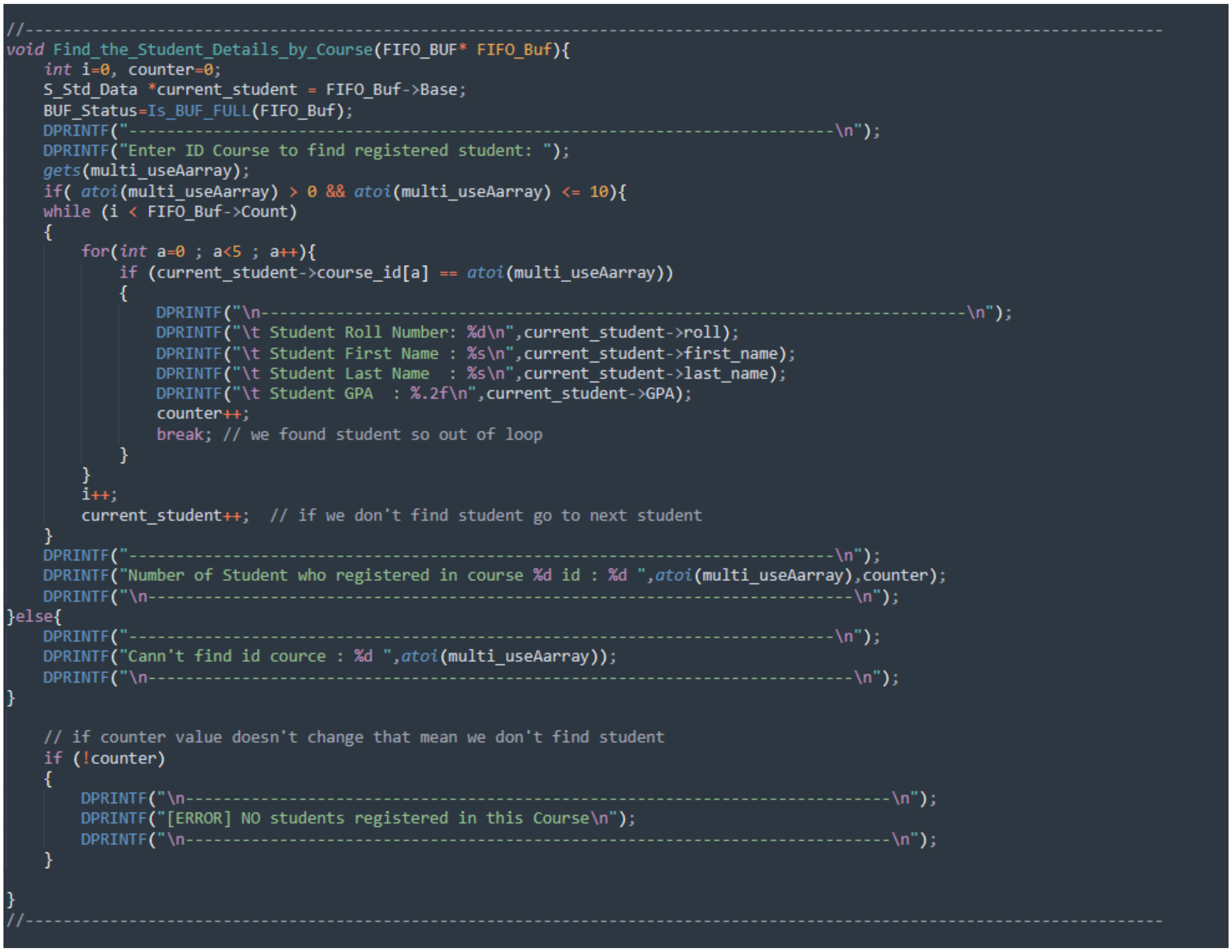
```
//------------------------------------------------------------------------------------------------------------------------
void Find_the_Student_Details_By_Roll_Number(FIFO_BUF* FIFO_Buf){
    int i=0 , var=0;
    S_Std_Data *current_student = FIFO_Buf->Base;
    BUF_Status=Is_BUF_FULL(FIFO_Buf);
    DPRINTF("---------------------------------------------------------------------------\n");
    DPRINTF("Enter student Roll number: ");
    gets(multi_useAarray);

    // iterate in buffer
    while (i < FIFO_Buf->Count)
    {
        if (current_student->roll == atoi(multi_useAarray))
        {
            DPRINTF("\n---------------------------------------------------------------------------\n");
            DPRINTF("\t Student Roll Number: %d\n",current_student->roll);
            DPRINTF("\t Student First Name : %s\n",current_student->first_name);
            DPRINTF("\t Student Last Name  : %s\n",current_student->last_name);
            DPRINTF("\t Student GPA  : %.2f\n",current_student->GPA);
            for (int a=0; a <5; a++)
            {
                DPRINTF("\t course %d id : %d\n",a+1,current_student->course_id[a]);

            }
            // if you found then  exit
            var=1;
             }
        i++;
        current_student++;
    }
    if(!var){
    DPRINTF("\n---------------------------------------------------------------------------\n");
    DPRINTF("[ERROR] Cann't find Roll number\n");
    DPRINTF("\n---------------------------------------------------------------------------\n");
}
}
//------------------------------------------------------------------------------------------------------------------------
```

This function searches in the FIFO buffer for a student with specified first name and displays their details if found , otherwise displays an error message.



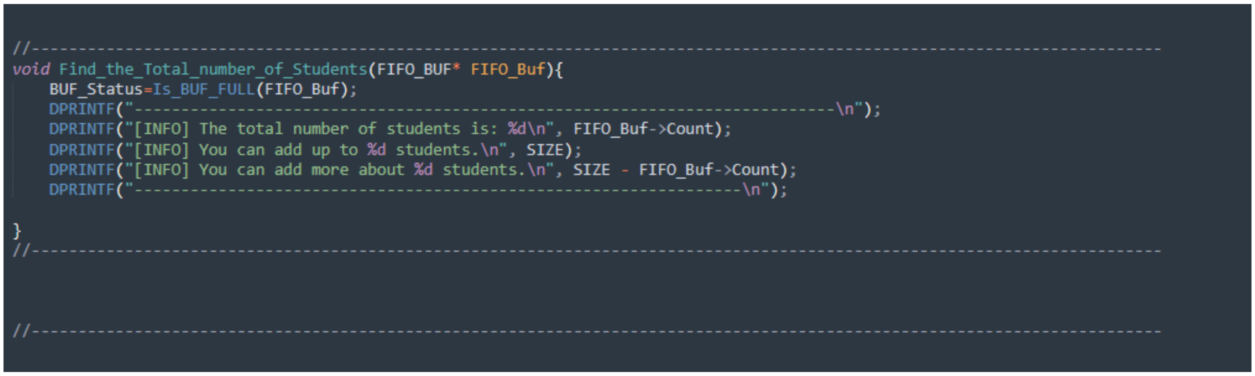
```c
//--------------------------------------------------------------------------------------------------------------------------
void Find_the_Student_Details_by_First_Name(FIFO_BUF* FIFO_Buf){
    int i=0 , a=0;
    S_Std_Data *current_student = FIFO_Buf->Base;
    BUF_Status=Is_BUF_FULL(FIFO_Buf);
    DPRINTF("----------------------------------------------------------------------------\n");
    DPRINTF("Enter student first name: ");
    gets(multi_useAarray);

    while (i < FIFO_Buf->Count)
    {
        if (strcmp(current_student->first_name,multi_useAarray) == 0)  // compare strings equality
        {
            DPRINTF("\n----------------------------------------------------------------------------\n");
            DPRINTF("\t Student Roll Number: %d\n",current_student->roll);
            DPRINTF("\t Student First Name : %s\n",current_student->first_name);
            DPRINTF("\t Student Last Name  : %s\n",current_student->last_name);
            DPRINTF("\t Student GPA  : %.1f\n",current_student->GPA);
            for ( a=0; a <5; a++)
            {
                DPRINTF("\t course %d id : %d\n",a+1,current_student->course_id[a]);

            }


        }
        i++;
        current_student++;

    }
    if( a == 0){
        DPRINTF("\n----------------------------------------------------------------------------\n");
        DPRINTF("[ERROR] First Name is not found in the FIFO\n");
        DPRINTF("\n----------------------------------------------------------------------------\n");
    }}
//--------------------------------------------------------------------------------------------------------------------------
```

This function searches in the FIFO buffer for a student that registered in specified ID course using ID course and displays their details if found and if the ID course not valid displays Cannot find id course .



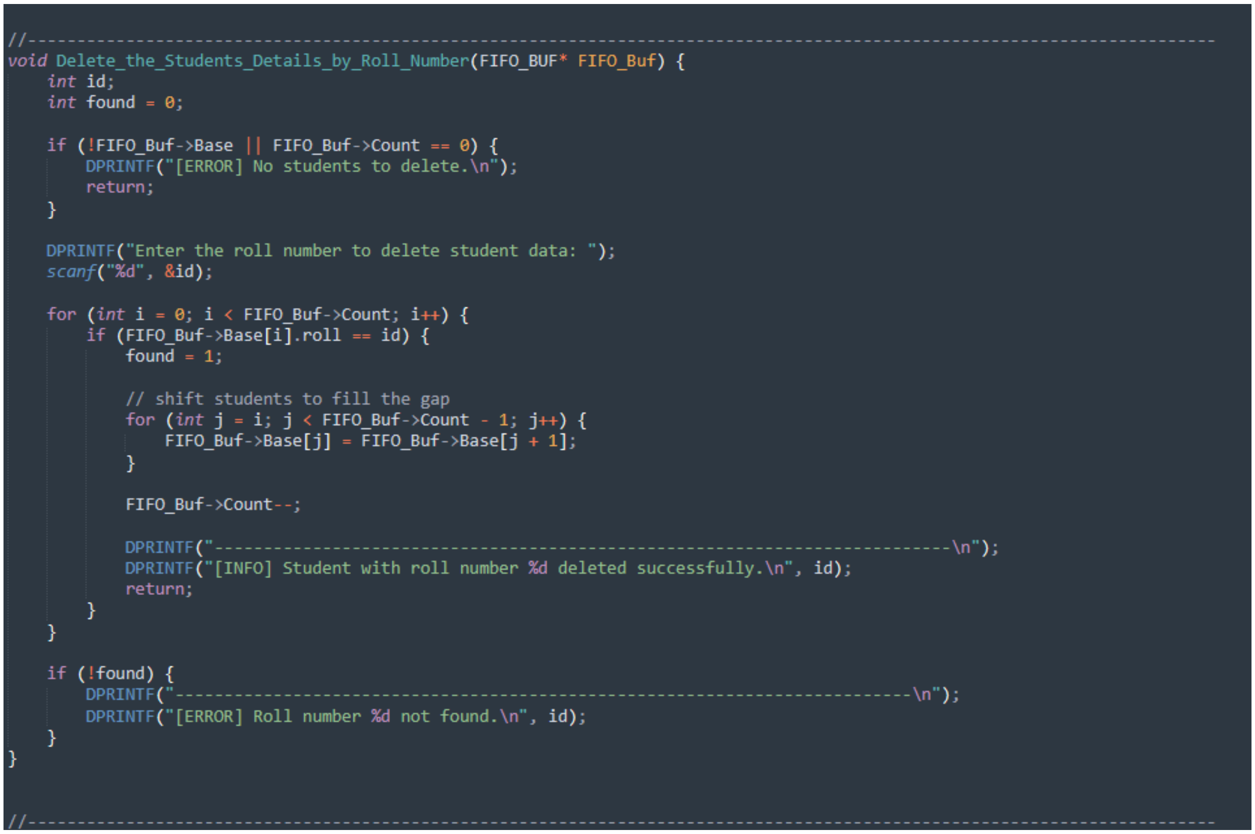
```c
//------------------------------------------------------------------------------------------------------------------------
void Find_the_Student_Details_by_Course(FIFO_BUF* FIFO_Buf){
    int i=0, counter=0;
    S_Std_Data *current_student = FIFO_Buf->Base;
    BUF_Status=Is_BUF_FULL(FIFO_Buf);
    DPRINTF("---------------------------------------------------------------------------\n");
    DPRINTF("Enter ID Course to find registered student: ");
    gets(multi_useAarray);
    if( atoi(multi_useAarray) > 0 && atoi(multi_useAarray) <= 10){
    while (i < FIFO_Buf->Count)
    {
        for(int a=0 ; a<5 ; a++){
            if (current_student->course_id[a] == atoi(multi_useAarray))
            {
                DPRINTF("\n---------------------------------------------------------------------------\n");
                DPRINTF("\t Student Roll Number: %d\n",current_student->roll);
                DPRINTF("\t Student First Name : %s\n",current_student->first_name);
                DPRINTF("\t Student Last Name  : %s\n",current_student->last_name);
                DPRINTF("\t Student GPA  : %.2f\n",current_student->GPA);
                counter++;
                break; // we found student so out of loop
            }
        }
        i++;
        current_student++;  // if we don't find student go to next student
    }
    DPRINTF("---------------------------------------------------------------------------\n");
    DPRINTF("Number of Student who registered in course %d id : %d ",atoi(multi_useAarray),counter);
    DPRINTF("\n---------------------------------------------------------------------------\n");
}else{
    DPRINTF("---------------------------------------------------------------------------\n");
    DPRINTF("Cann't find id cource : %d ",atoi(multi_useAarray));
    DPRINTF("\n---------------------------------------------------------------------------\n");
}

    // if counter value doesn't change that mean we don't find student
    if (!counter)
    {
        DPRINTF("\n---------------------------------------------------------------------------\n");
        DPRINTF("[ERROR] NO students registered in this Course\n");
        DPRINTF("\n---------------------------------------------------------------------------\n");
    }

}
//------------------------------------------------------------------------------------------------------------------------
```

This function displays the total number of students stored in the FIFO buffer and returns the remaining available space in FIFO buffer .



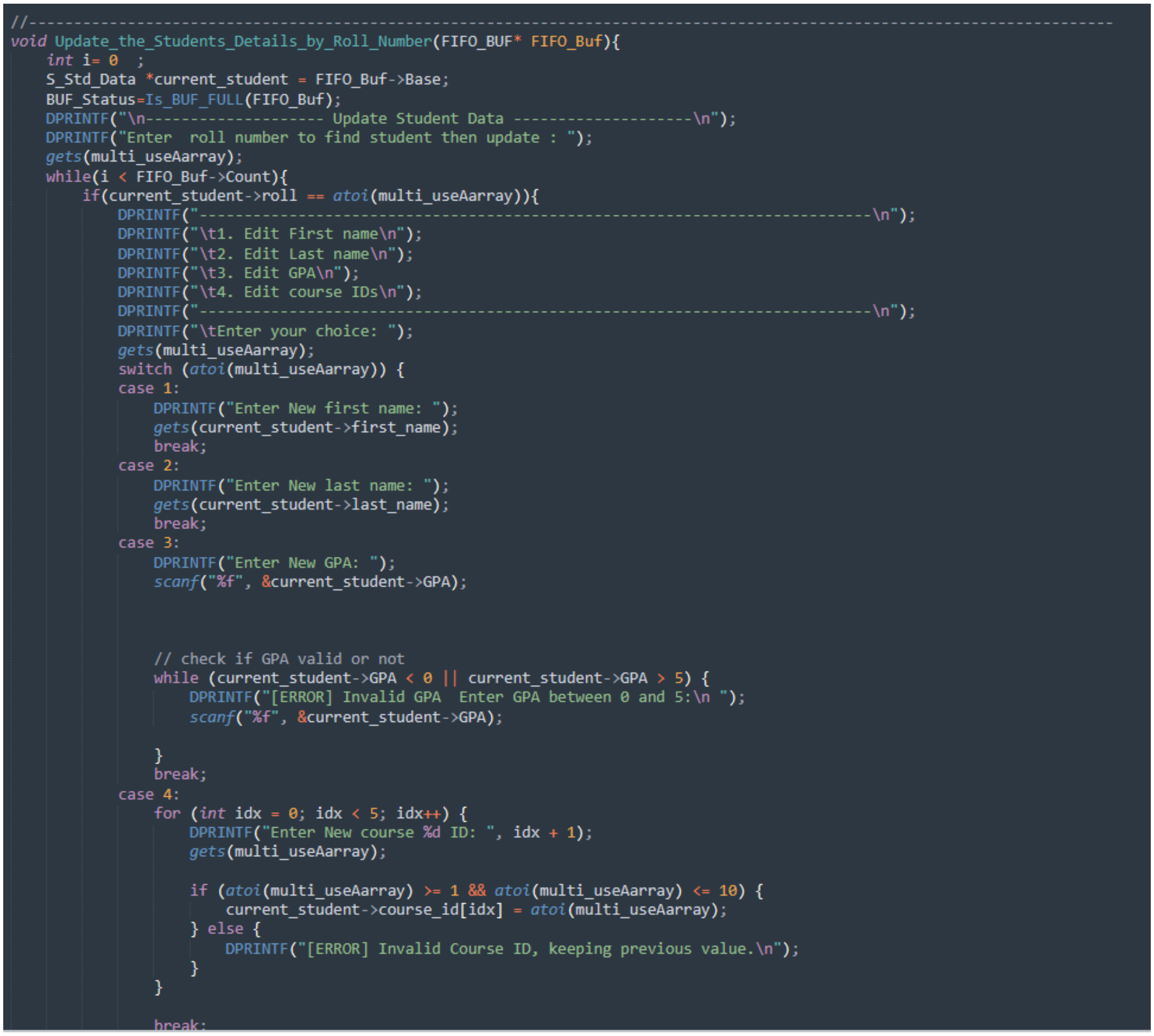
```c
//-------------------------------------------------------------------------------------------------------------------------
void Find_the_Total_number_of_Students(FIFO_BUF* FIFO_Buf){
    BUF_Status=Is_BUF_FULL(FIFO_Buf);
    DPRINTF("---------------------------------------------------------------------------\n");
    DPRINTF("[INFO] The total number of students is: %d\n", FIFO_Buf->Count);
    DPRINTF("[INFO] You can add up to %d students.\n", SIZE);
    DPRINTF("[INFO] You can add more about %d students.\n", SIZE - FIFO_Buf->Count);
    DPRINTF("-----------------------------------------------------------------\n");

}
//-------------------------------------------------------------------------------------------------------------------------



//-------------------------------------------------------------------------------------------------------------------------
```

This function deletes a student from the FIFO buffer by roll number , shifting remaining students to the left to fill the gap and updating the total count or displays an error if the roll number is not found.

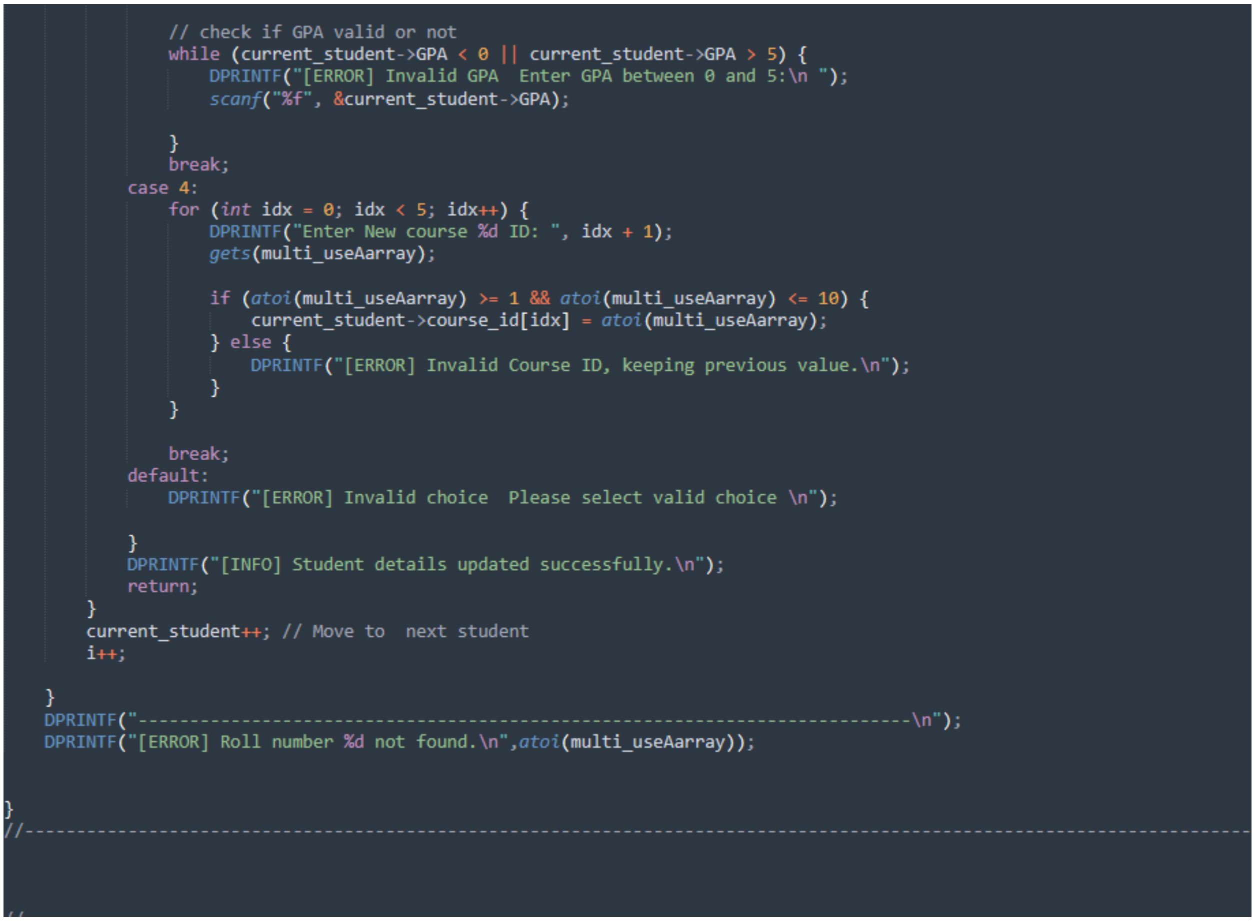


```c
//-------------------------------------------------------------------------------------------------------------------
void Delete_the_Students_Details_by_Roll_Number(FIFO_BUF* FIFO_Buf) {
    int id;
    int found = 0;

    if (!FIFO_Buf->Base || FIFO_Buf->Count == 0) {
        DPRINTF("[ERROR] No students to delete.\n");
        return;
    }

    DPRINTF("Enter the roll number to delete student data: ");
    scanf("%d", &id);

    for (int i = 0; i < FIFO_Buf->Count; i++) {
        if (FIFO_Buf->Base[i].roll == id) {
            found = 1;

            // shift students to fill the gap
            for (int j = i; j < FIFO_Buf->Count - 1; j++) {
                FIFO_Buf->Base[j] = FIFO_Buf->Base[j + 1];
            }

            FIFO_Buf->Count--;

            DPRINTF("---------------------------------------------------------------------------------\n");
            DPRINTF("[INFO] Student with roll number %d deleted successfully.\n", id);
            return;
        }
    }

    if (!found) {
        DPRINTF("---------------------------------------------------------------------------------\n");
        DPRINTF("[ERROR] Roll number %d not found.\n", id);
    }
}


//-------------------------------------------------------------------------------------------------------------------
```

This function updates a student detail in the FIFO buffer by roll number, allowing access of his name, GPA and course IDs , while ensuring that GPA and course ID are unique.

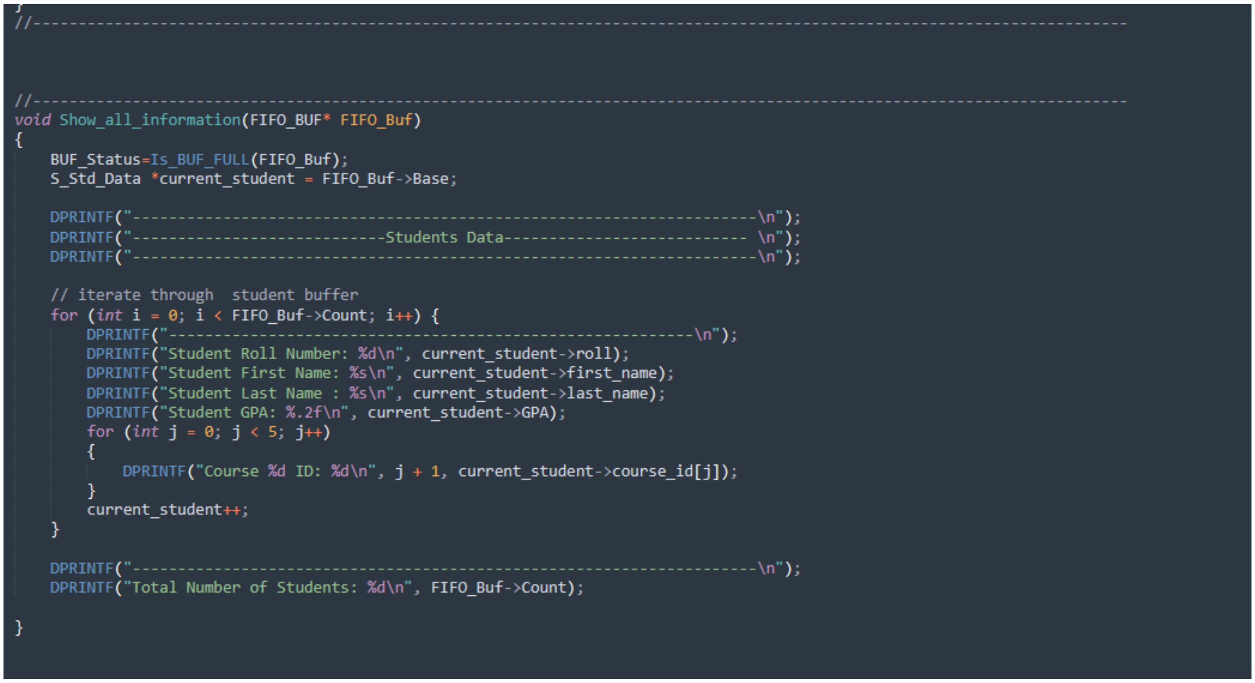


```c
//------------------------------------------------------------------------------------------------------------------------
void Update_the_Students_Details_by_Roll_Number(FIFO_BUF* FIFO_Buf){
    int i= 0  ;
    S_Std_Data *current_student = FIFO_Buf->Base;
    BUF_Status=Is_BUF_FULL(FIFO_Buf);
    DPRINTF("\n-------------------- Update Student Data --------------------\n");
    DPRINTF("Enter  roll number to find student then update : ");
    gets(multi_useAarray);
    while(i < FIFO_Buf->Count){
        if(current_student->roll == atoi(multi_useAarray)){
            DPRINTF("---------------------------------------------------------------------------\n");
            DPRINTF("\t1. Edit First name\n");
            DPRINTF("\t2. Edit Last name\n");
            DPRINTF("\t3. Edit GPA\n");
            DPRINTF("\t4. Edit course IDs\n");
            DPRINTF("---------------------------------------------------------------------------\n");
            DPRINTF("\tEnter your choice: ");
            gets(multi_useAarray);
            switch (atoi(multi_useAarray)) {
            case 1:
                DPRINTF("Enter New first name: ");
                gets(current_student->first_name);
                break;
            case 2:
                DPRINTF("Enter New last name: ");
                gets(current_student->last_name);
                break;
            case 3:
                DPRINTF("Enter New GPA: ");
                scanf("%f", &current_student->GPA);



                // check if GPA valid or not
                while (current_student->GPA < 0 || current_student->GPA > 5) {
                    DPRINTF("[ERROR] Invalid GPA  Enter GPA between 0 and 5:\n ");
                    scanf("%f", &current_student->GPA);

                }
                break;
            case 4:
                for (int idx = 0; idx < 5; idx++) {
                    DPRINTF("Enter New course %d ID: ", idx + 1);
                    gets(multi_useAarray);

                    if (atoi(multi_useAarray) >= 1 && atoi(multi_useAarray) <= 10) {
                        current_student->course_id[idx] = atoi(multi_useAarray);
                    } else {
                        DPRINTF("[ERROR] Invalid Course ID, keeping previous value.\n");
                    }
                }

                break;
```



```c
                // check if GPA valid or not
                while (current_student->GPA < 0 || current_student->GPA > 5) {
                    DPRINTF("[ERROR] Invalid GPA  Enter GPA between 0 and 5:\n ");
                    scanf("%f", &current_student->GPA);

                }
                break;
            case 4:
                for (int idx = 0; idx < 5; idx++) {
                    DPRINTF("Enter New course %d ID: ", idx + 1);
                    gets(multi_useAarray);

                    if (atoi(multi_useAarray) >= 1 && atoi(multi_useAarray) <= 10) {
                        current_student->course_id[idx] = atoi(multi_useAarray);
                    } else {
                        DPRINTF("[ERROR] Invalid Course ID, keeping previous value.\n");
                    }
                }

                break;
            default:
                DPRINTF("[ERROR] Invalid choice  Please select valid choice \n");

            }
            DPRINTF("[INFO] Student details updated successfully.\n");
            return;
        }
        current_student++; // Move to  next student
        i++;

    }
    DPRINTF("---------------------------------------------------------------------------\n");
    DPRINTF("[ERROR] Roll number %d not found.\n",atoi(multi_useAarray));


}
//------------------------------------------------------------------------------------------------------------------------


//
```

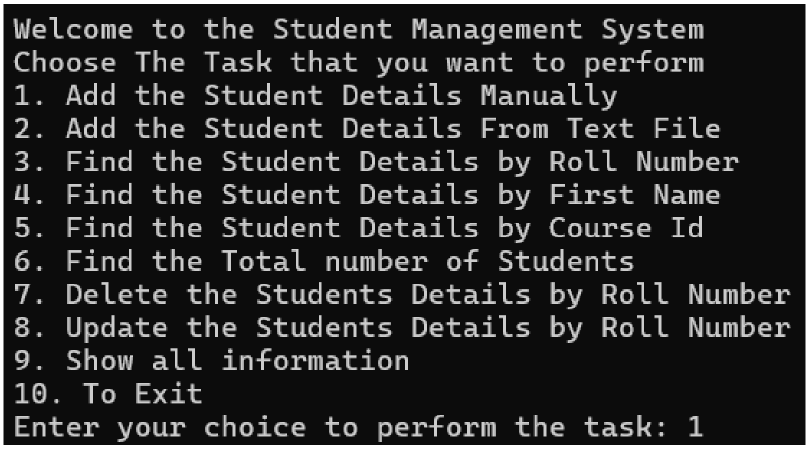
This function displays all students Data stored in the FIFO buffer with their roll number, first and last name , GPA and course IDs with the total number of students.



```c
}
//--------------------------------------------------------------------------------------------------------------------------



//--------------------------------------------------------------------------------------------------------------------------
void Show_all_information(FIFO_BUF* FIFO_Buf)
{
    BUF_Status=Is_BUF_FULL(FIFO_Buf);
    S_Std_Data *current_student = FIFO_Buf->Base;

    DPRINTF("---------------------------------------------------------------------\n");
    DPRINTF("----------------------------Students Data-------------------------- \n");
    DPRINTF("---------------------------------------------------------------------\n");

    // iterate through  student buffer
    for (int i = 0; i < FIFO_Buf->Count; i++) {
        DPRINTF("----------------------------------------------------------\n");
        DPRINTF("Student Roll Number: %d\n", current_student->roll);
        DPRINTF("Student First Name: %s\n", current_student->first_name);
        DPRINTF("Student Last Name : %s\n", current_student->last_name);
        DPRINTF("Student GPA: %.2f\n", current_student->GPA);
        for (int j = 0; j < 5; j++)
        {
            DPRINTF("Course %d ID: %d\n", j + 1, current_student->course_id[j]);
        }
        current_student++;
    }

    DPRINTF("---------------------------------------------------------------------\n");
    DPRINTF("Total Number of Students: %d\n", FIFO_Buf->Count);

}
```

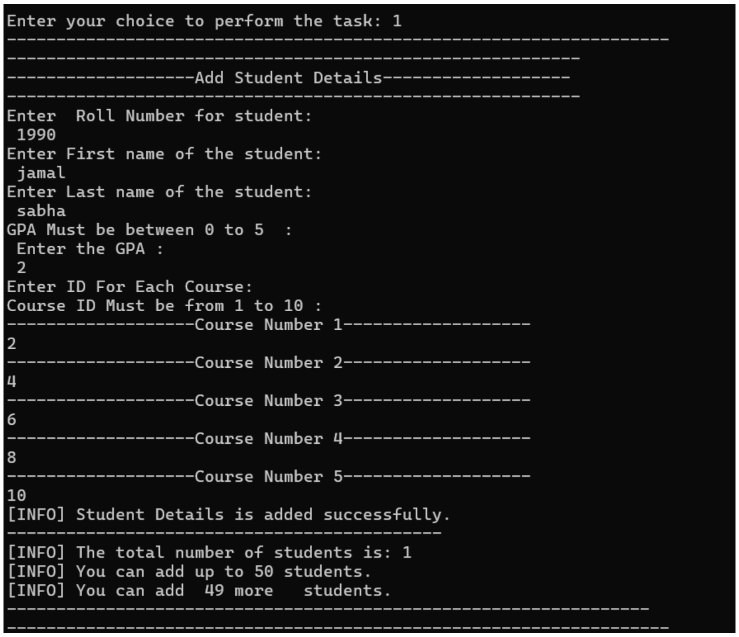
Run the code:

Display Menu:



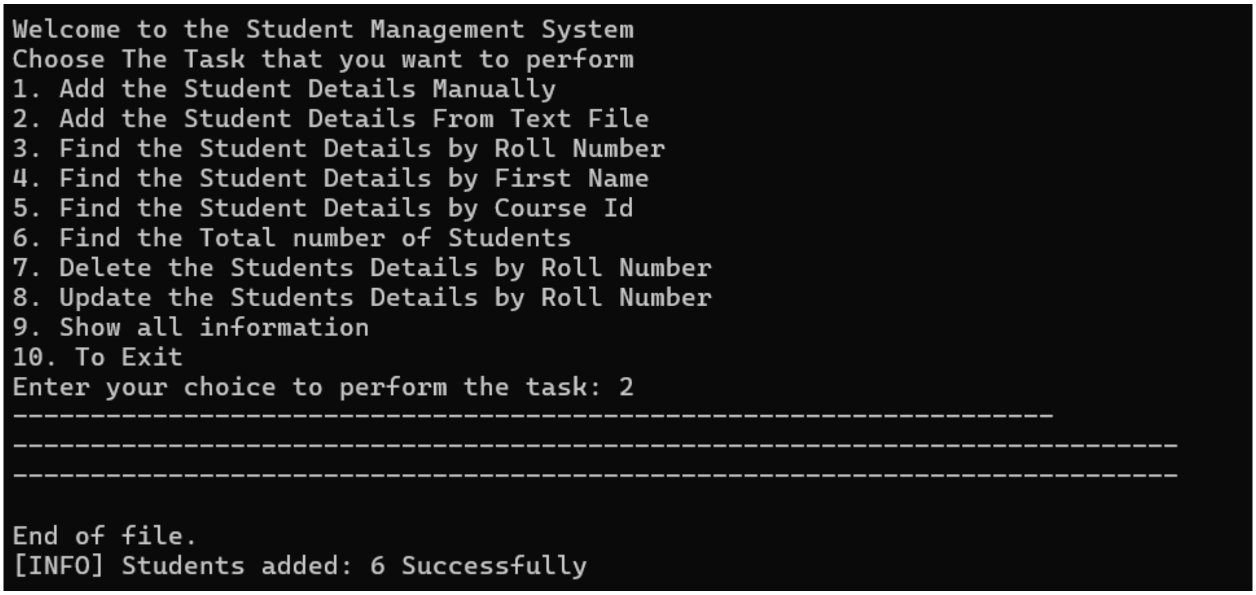
```
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 1
```

Add Student Manually:



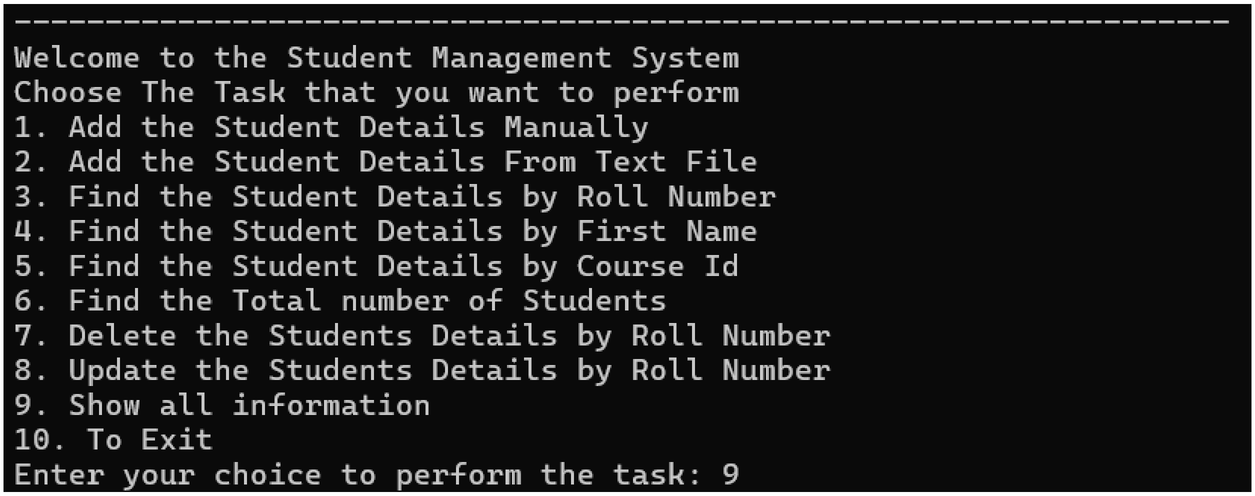
```
Enter your choice to perform the task: 1
-------------------------------------------------------------------
----------------------------------------------------------
-------------------Add Student Details-------------------
----------------------------------------------------------
Enter  Roll Number for student:
 1990
Enter First name of the student:
 jamal
Enter Last name of the student:
 sabha
GPA Must be between 0 to 5  :
 Enter the GPA :
 2
Enter ID For Each Course:
Course ID Must be from 1 to 10 :
-------------------Course Number 1-------------------
2
-------------------Course Number 2-------------------
4
-------------------Course Number 3-------------------
6
-------------------Course Number 4-------------------
8
-------------------Course Number 5-------------------
10
[INFO] Student Details is added successfully.
--------------------------------------------
[INFO] The total number of students is: 1
[INFO] You can add up to 50 students.
[INFO] You can add  49 more   students.
-----------------------------------------------------------------
-------------------------------------------------------------------
```

Add Student from text file :

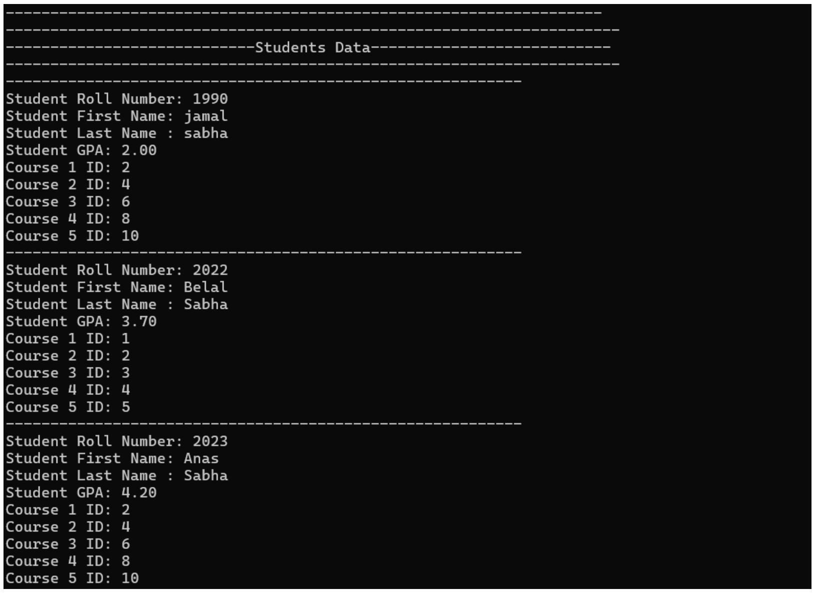


```
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 2
-------------------------------------------------------------------
---------------------------------------------------------------------------
---------------------------------------------------------------------------

End of file.
[INFO] Students added: 6 Successfully
```

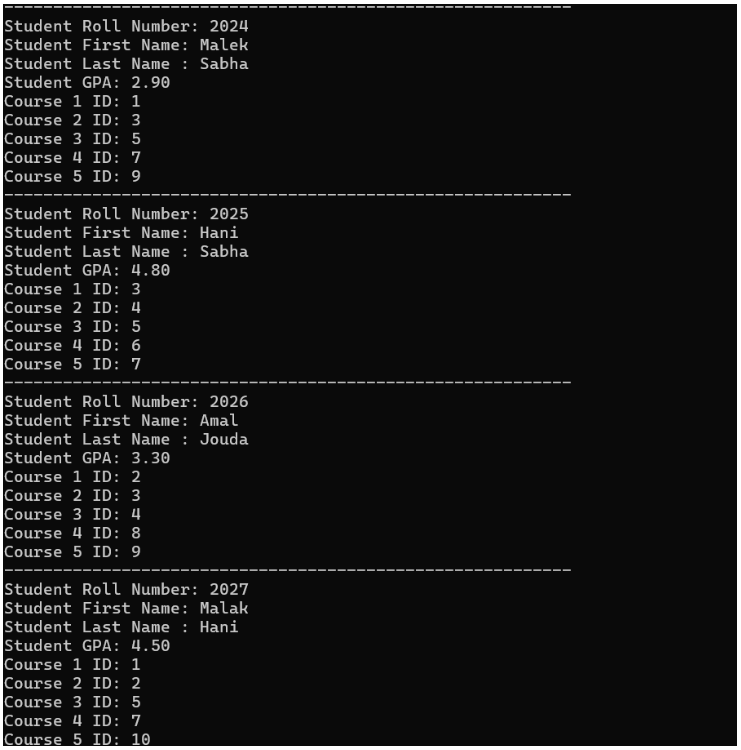
Show all Students data :



```
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 9
```



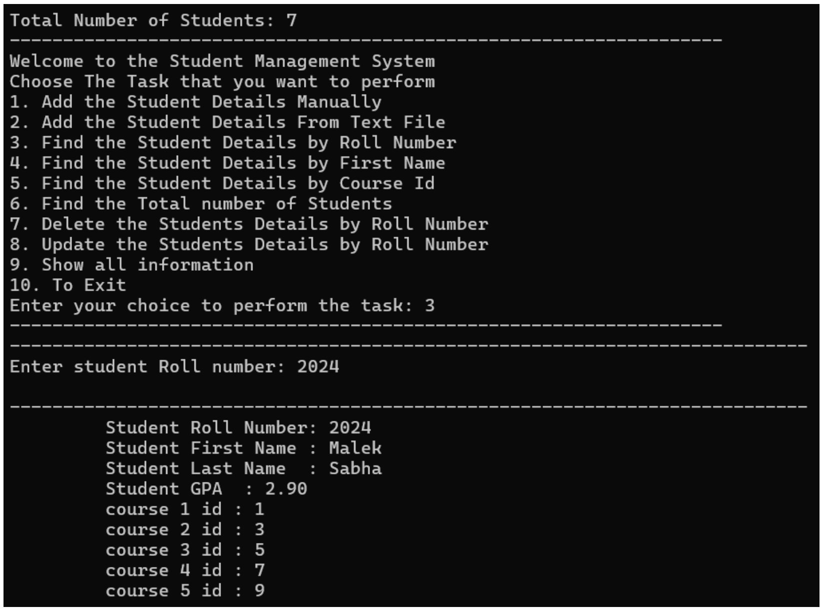
```
-------------------------------------------------------------------
---------------------------------------------------------------------
----------------------------Students Data---------------------------
---------------------------------------------------------------------
----------------------------------------------------------
Student Roll Number: 1990
Student First Name: jamal
Student Last Name : sabha
Student GPA: 2.00
Course 1 ID: 2
Course 2 ID: 4
Course 3 ID: 6
Course 4 ID: 8
Course 5 ID: 10
----------------------------------------------------------
Student Roll Number: 2022
Student First Name: Belal
Student Last Name : Sabha
Student GPA: 3.70
Course 1 ID: 1
Course 2 ID: 2
Course 3 ID: 3
Course 4 ID: 4
Course 5 ID: 5
----------------------------------------------------------
Student Roll Number: 2023
Student First Name: Anas
Student Last Name : Sabha
Student GPA: 4.20
Course 1 ID: 2
Course 2 ID: 4
Course 3 ID: 6
Course 4 ID: 8
Course 5 ID: 10
```



```
----------------------------------------------------------
Student Roll Number: 2024
Student First Name: Malek
Student Last Name : Sabha
Student GPA: 2.90
Course 1 ID: 1
Course 2 ID: 3
Course 3 ID: 5
Course 4 ID: 7
Course 5 ID: 9
----------------------------------------------------------
Student Roll Number: 2025
Student First Name: Hani
Student Last Name : Sabha
Student GPA: 4.80
Course 1 ID: 3
Course 2 ID: 4
Course 3 ID: 5
Course 4 ID: 6
Course 5 ID: 7
----------------------------------------------------------
Student Roll Number: 2026
Student First Name: Amal
Student Last Name : Jouda
Student GPA: 3.30
Course 1 ID: 2
Course 2 ID: 3
Course 3 ID: 4
Course 4 ID: 8
Course 5 ID: 9
----------------------------------------------------------
Student Roll Number: 2027
Student First Name: Malak
Student Last Name : Hani
Student GPA: 4.50
Course 1 ID: 1
Course 2 ID: 2
Course 3 ID: 5
Course 4 ID: 7
Course 5 ID: 10
```

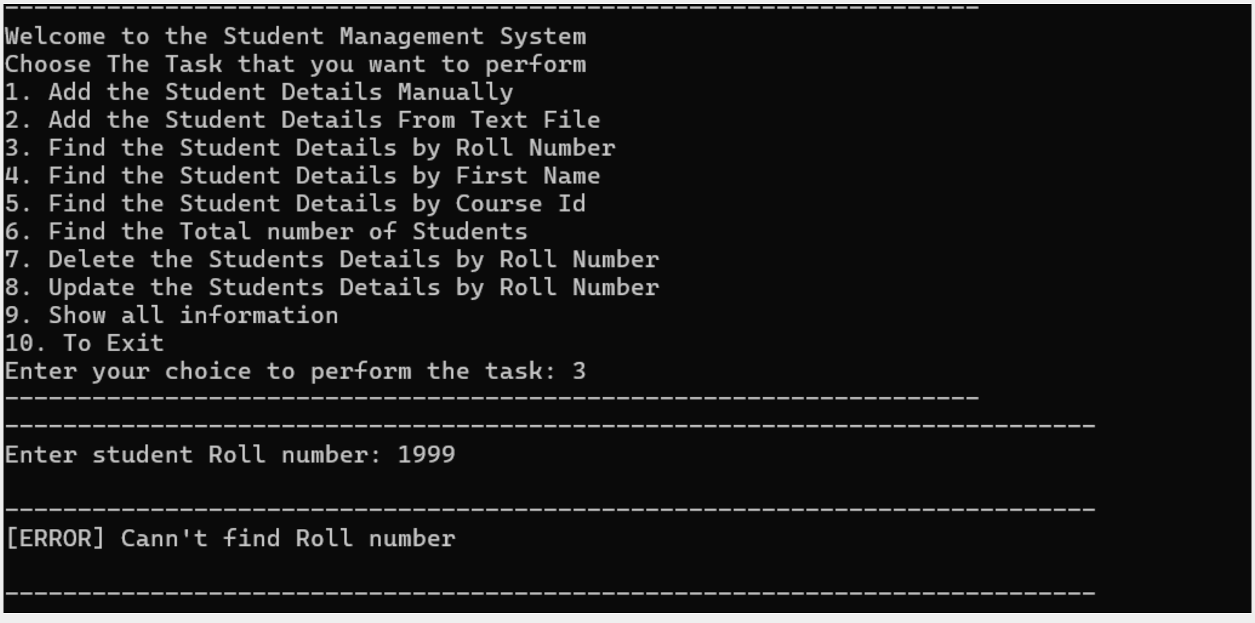
Find Student Details By roll number:

1-if roll number exist in FIFO.



```
Total Number of Students: 7
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 3
-------------------------------------------------------------------
---------------------------------------------------------------------------
Enter student Roll number: 2024

---------------------------------------------------------------------------
         Student Roll Number: 2024
         Student First Name : Malek
         Student Last Name  : Sabha
         Student GPA  : 2.90
         course 1 id : 1
         course 2 id : 3
         course 3 id : 5
         course 4 id : 7
         course 5 id : 9
```

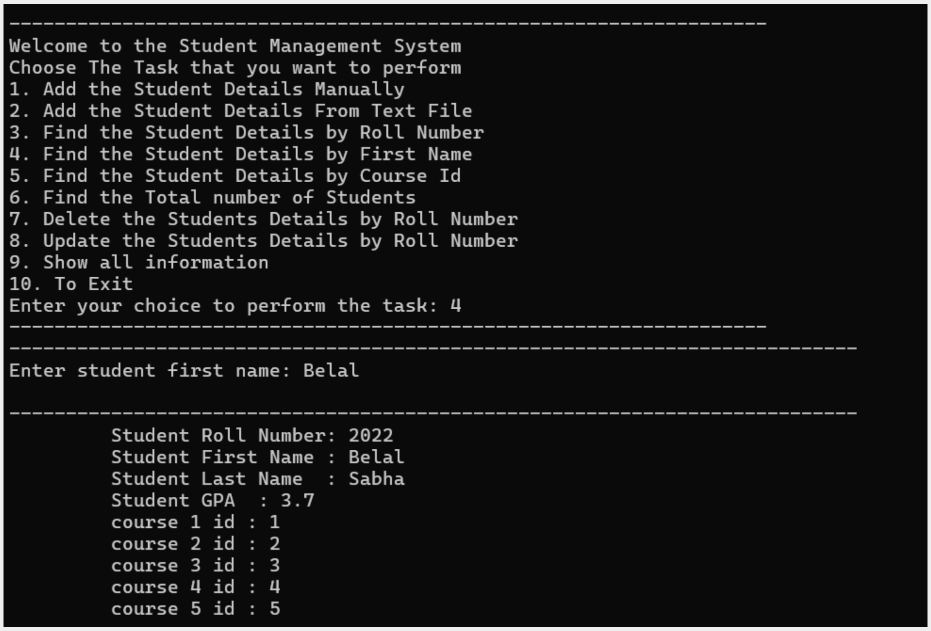
2-if roll number doesn’t exist in FIFO.



```
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 3
-------------------------------------------------------------------
---------------------------------------------------------------------------
Enter student Roll number: 1999

---------------------------------------------------------------------------
[ERROR] Cann't find Roll number

---------------------------------------------------------------------------
```

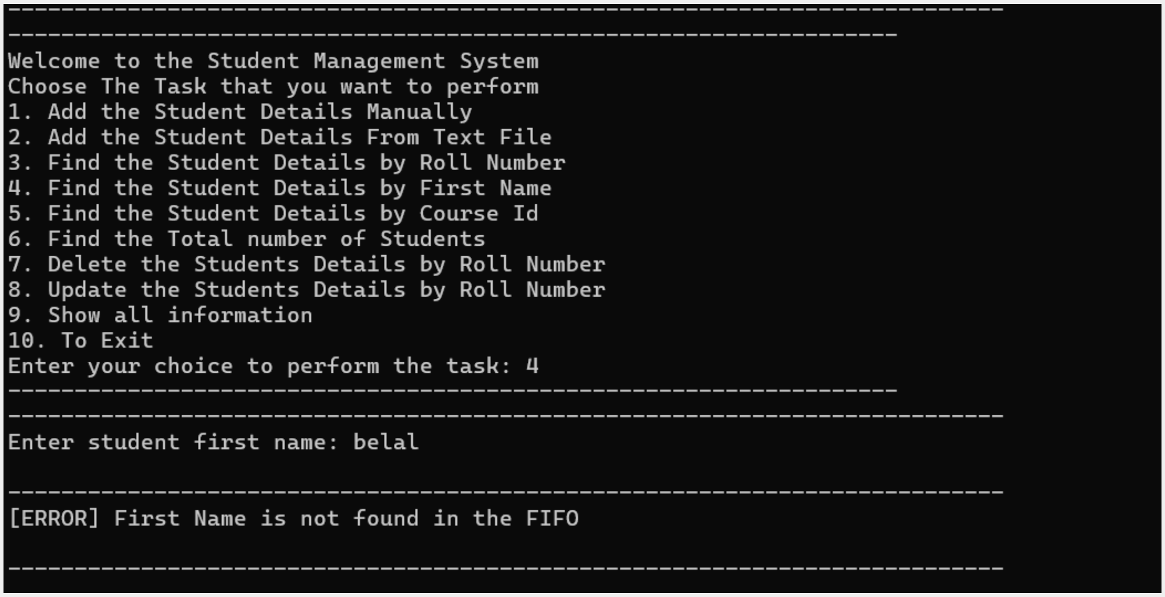
Find Student Details By first name:

1-if first name exist in FIFO.



```
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 4
-------------------------------------------------------------------
---------------------------------------------------------------------------
Enter student first name: Belal

---------------------------------------------------------------------------
         Student Roll Number: 2022
         Student First Name : Belal
         Student Last Name  : Sabha
         Student GPA   : 3.7
         course 1 id : 1
         course 2 id : 2
         course 3 id : 3
         course 4 id : 4
         course 5 id : 5
```

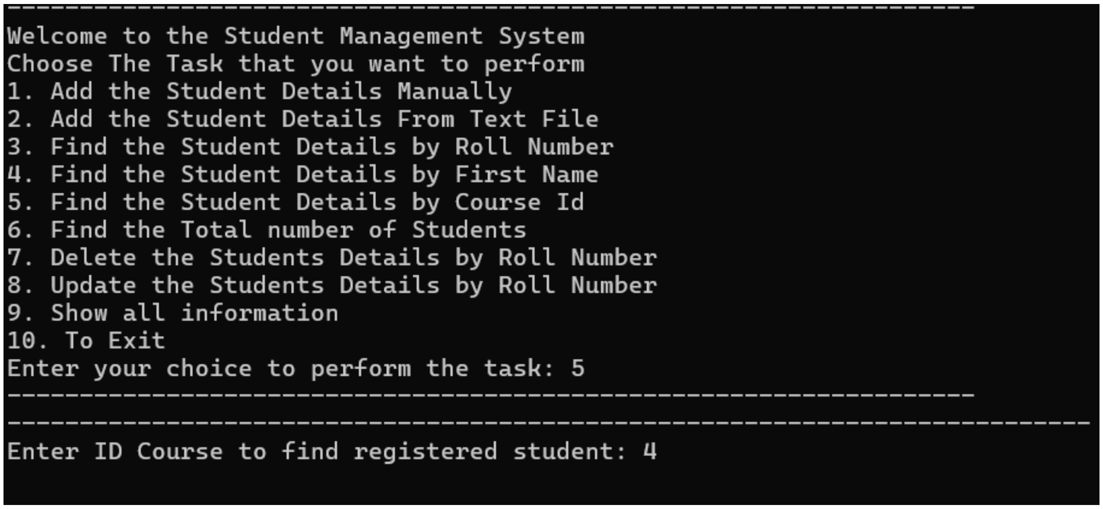
2-if first name doesn’t exist in FIFO.



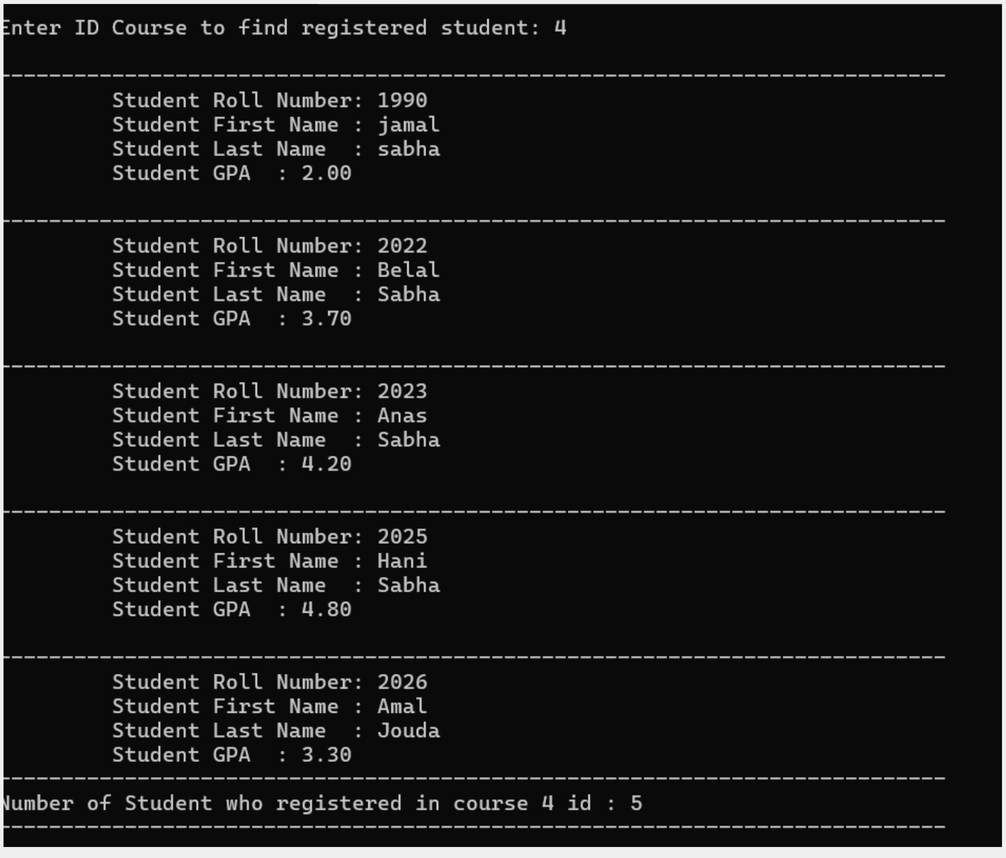
```
---------------------------------------------------------------------------
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 4
-------------------------------------------------------------------
---------------------------------------------------------------------------
Enter student first name: belal


---------------------------------------------------------------------------
[ERROR] First Name is not found in the FIFO

---------------------------------------------------------------------------
```

Find Student Details By ID course:

1-if ID Course exist in FIFO.

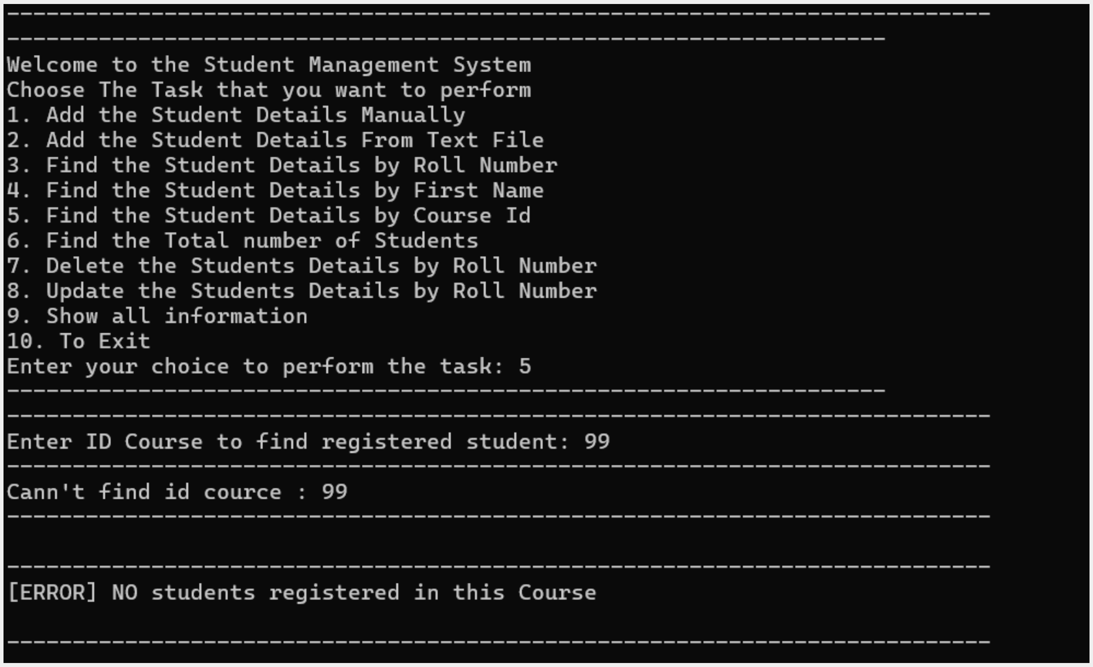


```
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 5
-------------------------------------------------------------------
---------------------------------------------------------------------------

Enter ID Course to find registered student: 4
```



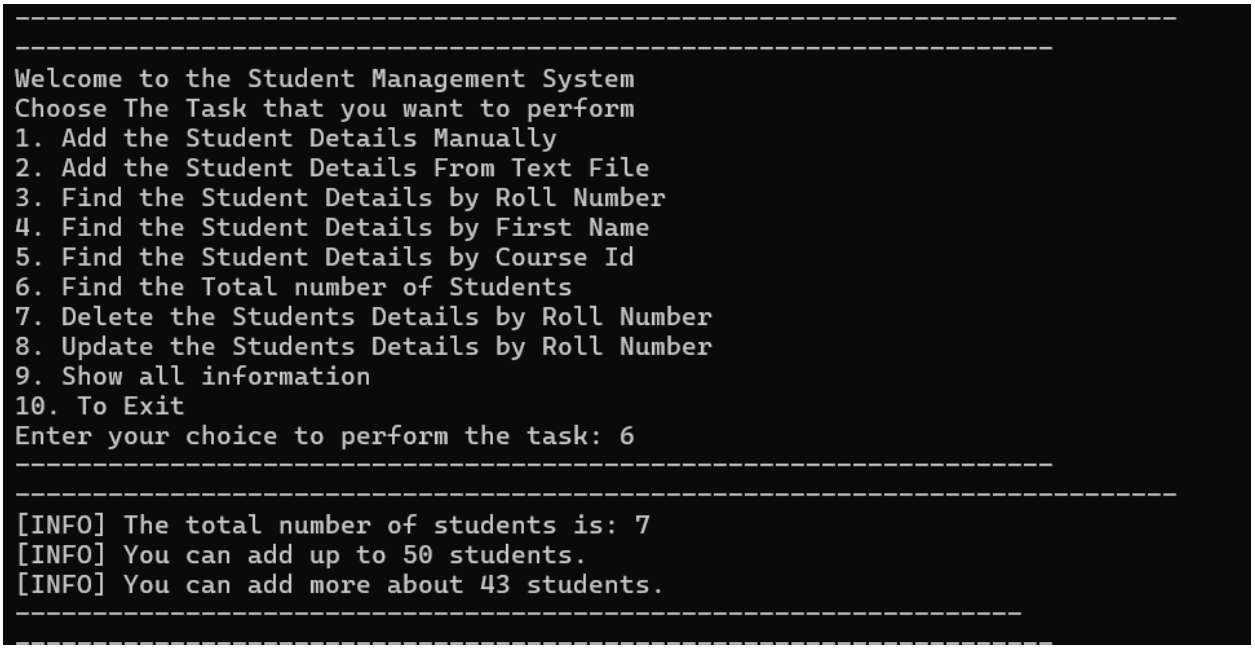
```
Enter ID Course to find registered student: 4

--------------------------------------------------------------------------
        Student Roll Number: 1990
        Student First Name : jamal
        Student Last Name  : sabha
        Student GPA  : 2.00

--------------------------------------------------------------------------
        Student Roll Number: 2022
        Student First Name : Belal
        Student Last Name  : Sabha
        Student GPA  : 3.70

--------------------------------------------------------------------------
        Student Roll Number: 2023
        Student First Name : Anas
        Student Last Name  : Sabha
        Student GPA  : 4.20

--------------------------------------------------------------------------
        Student Roll Number: 2025
        Student First Name : Hani
        Student Last Name  : Sabha
        Student GPA  : 4.80

--------------------------------------------------------------------------
        Student Roll Number: 2026
        Student First Name : Amal
        Student Last Name  : Jouda
        Student GPA  : 3.30
--------------------------------------------------------------------------
Number of Student who registered in course 4 id : 5
--------------------------------------------------------------------------
```

2-if ID course doesn’t exist in FIFO.



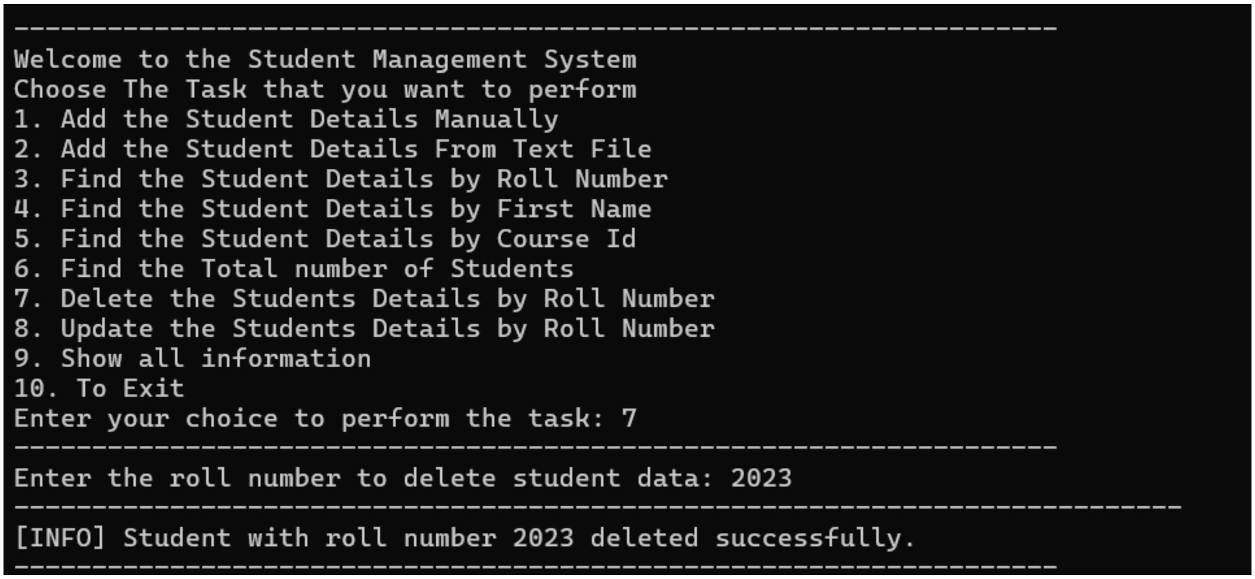
```
---------------------------------------------------------------------------
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 5
-------------------------------------------------------------------
---------------------------------------------------------------------------
Enter ID Course to find registered student: 99
---------------------------------------------------------------------------
Cann't find id cource : 99
---------------------------------------------------------------------------

---------------------------------------------------------------------------
[ERROR] NO students registered in this Course

---------------------------------------------------------------------------
```

Find Total Number of Students in FIFO:



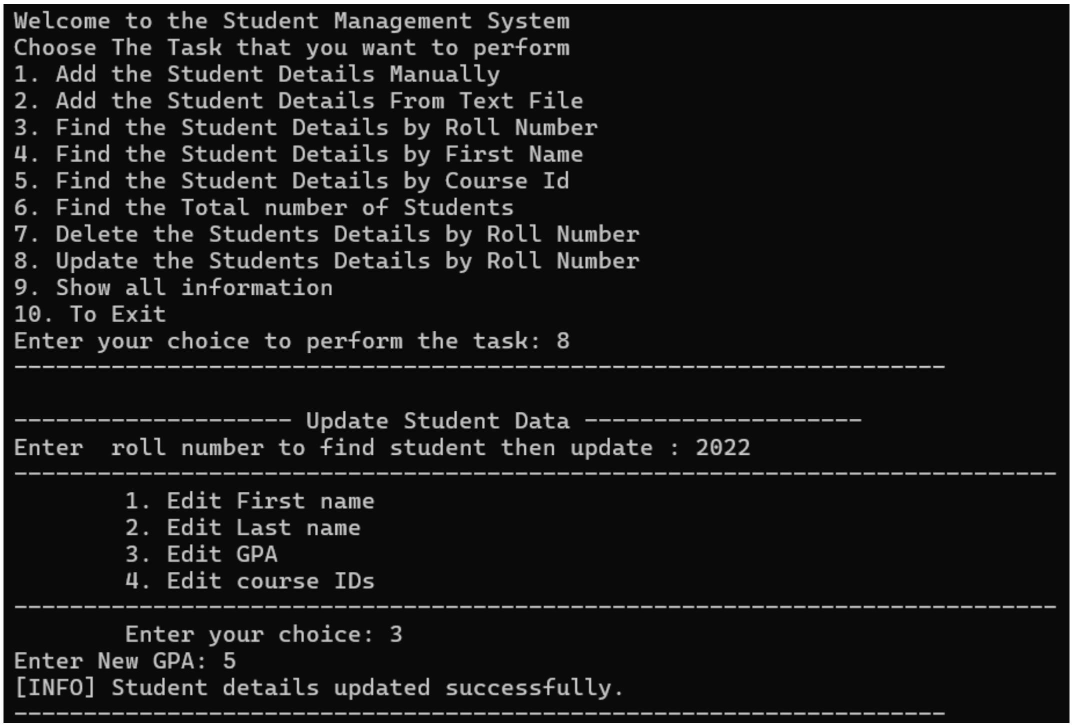
```
---------------------------------------------------------------------------
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 6
-------------------------------------------------------------------
---------------------------------------------------------------------------
[INFO] The total number of students is: 7
[INFO] You can add up to 50 students.
[INFO] You can add more about 43 students.
-----------------------------------------------------------------
-------------------------------------------------------------------
```

Delete student :



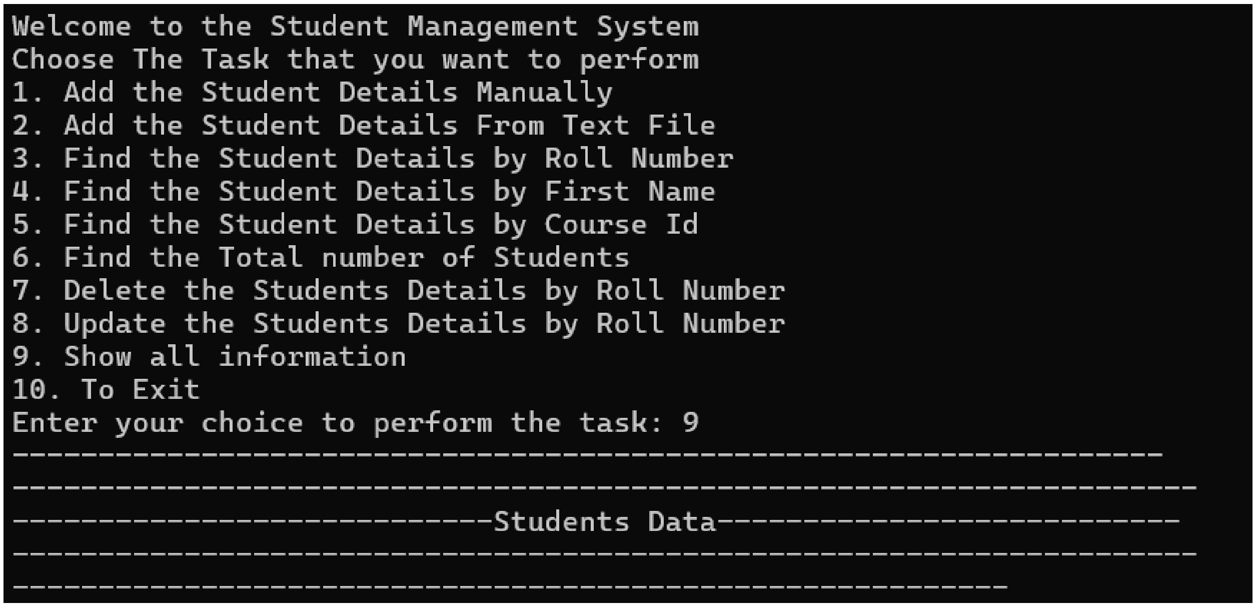
```
-------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 7
-------------------------------------------------------------------
Enter the roll number to delete student data: 2023
---------------------------------------------------------------------------
[INFO] Student with roll number 2023 deleted successfully.
-------------------------------------------------------------------
```

Update student Data :

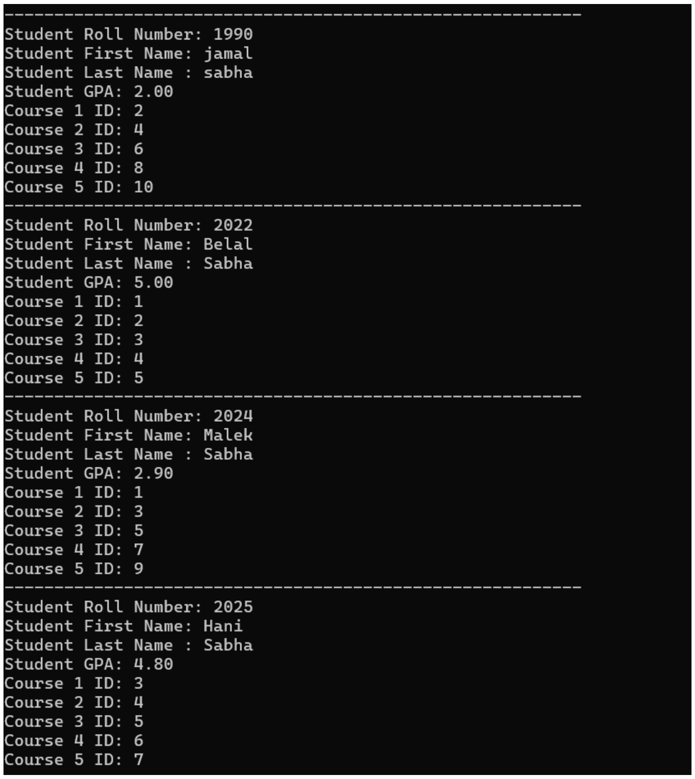


```
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 8
-------------------------------------------------------------------

-------------------- Update Student Data --------------------
Enter  roll number to find student then update : 2022
---------------------------------------------------------------------------
        1. Edit First name
        2. Edit Last name
        3. Edit GPA
        4. Edit course IDs
---------------------------------------------------------------------------
        Enter your choice: 3
Enter New GPA: 5
[INFO] Student details updated successfully.
-------------------------------------------------------------------
```

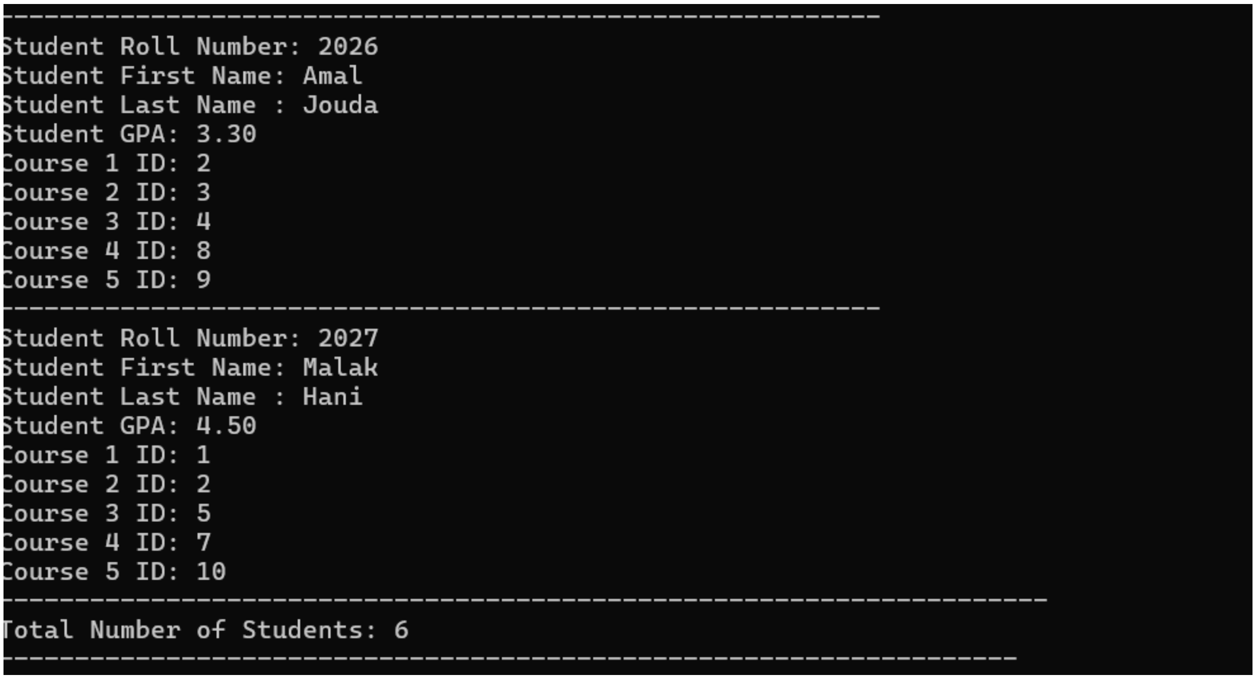
Displaying students Data after deleting and updating :



```
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 9
-------------------------------------------------------------------
---------------------------------------------------------------------
----------------------------Students Data---------------------------
---------------------------------------------------------------------
----------------------------------------------------------
```

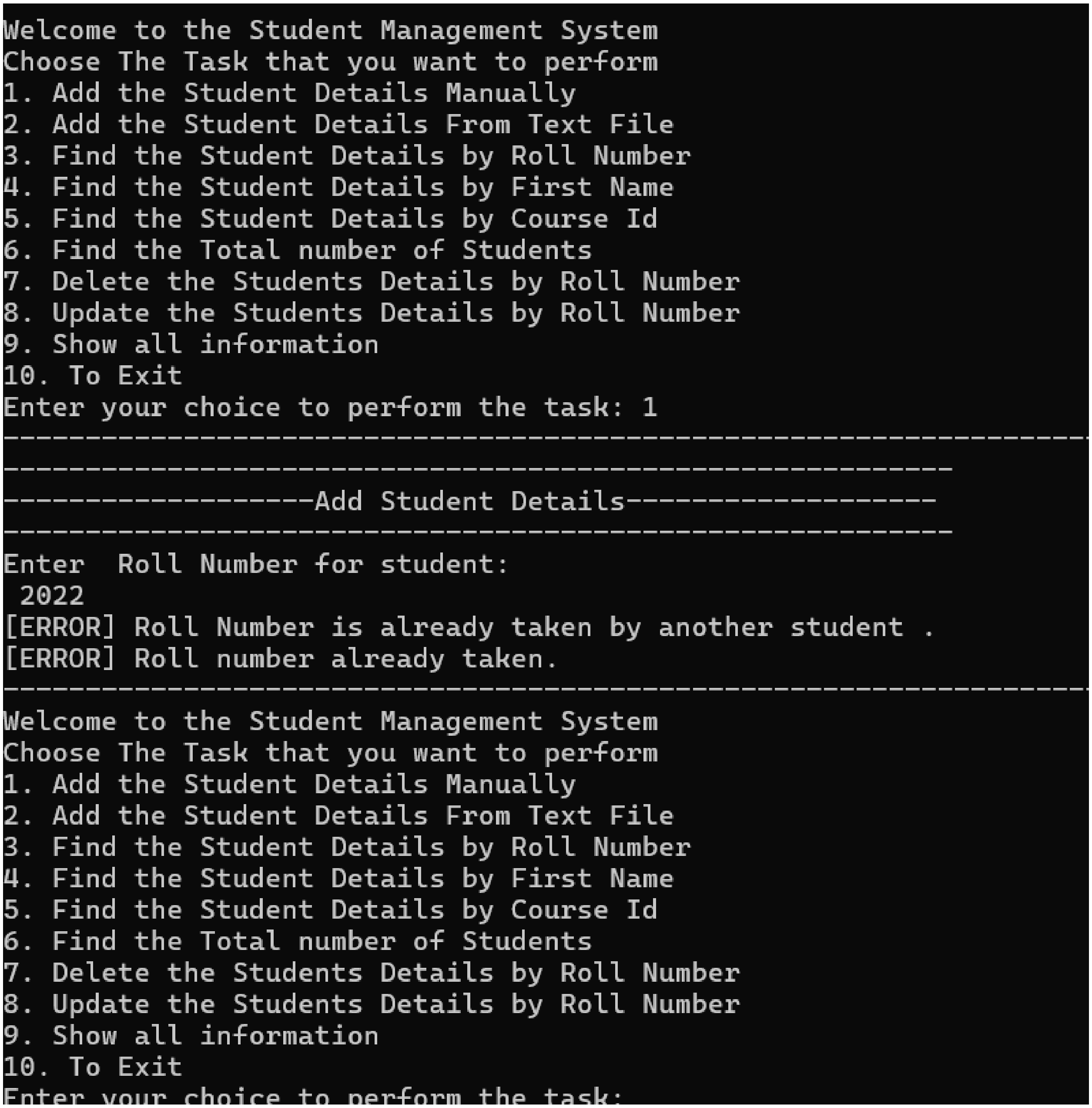


```
----------------------------------------------------------
Student Roll Number: 1990
Student First Name: jamal
Student Last Name : sabha
Student GPA: 2.00
Course 1 ID: 2
Course 2 ID: 4
Course 3 ID: 6
Course 4 ID: 8
Course 5 ID: 10
----------------------------------------------------------
Student Roll Number: 2022
Student First Name: Belal
Student Last Name : Sabha
Student GPA: 5.00
Course 1 ID: 1
Course 2 ID: 2
Course 3 ID: 3
Course 4 ID: 4
Course 5 ID: 5
----------------------------------------------------------
Student Roll Number: 2024
Student First Name: Malek
Student Last Name : Sabha
Student GPA: 2.90
Course 1 ID: 1
Course 2 ID: 3
Course 3 ID: 5
Course 4 ID: 7
Course 5 ID: 9
----------------------------------------------------------
Student Roll Number: 2025
Student First Name: Hani
Student Last Name : Sabha
Student GPA: 4.80
Course 1 ID: 3
Course 2 ID: 4
Course 3 ID: 5
Course 4 ID: 6
Course 5 ID: 7
```



```
----------------------------------------------------------
Student Roll Number: 2026
Student First Name: Amal
Student Last Name : Jouda
Student GPA: 3.30
Course 1 ID: 2
Course 2 ID: 3
Course 3 ID: 4
Course 4 ID: 8
Course 5 ID: 9
----------------------------------------------------------
Student Roll Number: 2027
Student First Name: Malak
Student Last Name : Hani
Student GPA: 4.50
Course 1 ID: 1
Course 2 ID: 2
Course 3 ID: 5
Course 4 ID: 7
Course 5 ID: 10
---------------------------------------------------------------------
Total Number of Students: 6
-------------------------------------------------------------------
```

Add Student with roll number taken by another student :



```
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task: 1
------------------------------------------------------------------
----------------------------------------------------------
-------------------Add Student Details-------------------
----------------------------------------------------------
Enter  Roll Number for student:
 2022
[ERROR] Roll Number is already taken by another student .
[ERROR] Roll number already taken.
------------------------------------------------------------------
Welcome to the Student Management System
Choose The Task that you want to perform
1. Add the Student Details Manually
2. Add the Student Details From Text File
3. Find the Student Details by Roll Number
4. Find the Student Details by First Name
5. Find the Student Details by Course Id
6. Find the Total number of Students
7. Delete the Students Details by Roll Number
8. Update the Students Details by Roll Number
9. Show all information
10. To Exit
Enter your choice to perform the task:
```