DBMS_LAB_2 dated 25/07/24

Question 1

Develop an implementation package using 'C' program to process a FILE containing studentdetails for the given queries.
A student record has the following format:
Std_rollno, Std_name, Dept, C1, C1_c, C1_g, C2, C2_c, C2_g, C3, C3_c, C3_g
Note: C1 refers to Course1, C1_c refers to credit of the course, C1_g refers to the grade in that course and so on.
Every student should have a unique rollno.
A student should have at least 3 courses and maximum four.
A grade point is in integer: S - 10; A - 9; B - 8; C - 7; D - 6; E - 5; F – 0.
Create a file and develop a menu driven system for the following queries.
a. Insert at least 5 student records.
b. Create a column 'GPA' for all the students.
c. For a student with four courses, delete(deregister) a course name.
d. For the same student you deleted in 'c', insert a new course name.
e. Update the name of a course for two different students.
f. Calculate GPA of all students using the GPA formula. Refer the following:
https://www.nitt.edu/home/academics/rules/BTech_Regulations_2019.pdf
g. Upgrade the grade point of a student who has secured '7' in a course.
h. Calculate the updated GPA of the student in 'g'.
i. Generate a Grade report of a student given the roll no. or name.

Answers :

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define FILENAME "students.dat"
#define LEN 50
#define NCOURSES 4

typedef struct {
   char rollno[20];
   char name[LEN];
   char dept[LEN];
   char courses[NCOURSES][LEN];
   int credits[NCOURSES];
   int grades[NCOURSES];
   int num_courses;
   float gpa;
} Student;
```

```c
int gradeToPoint(char grade) {
    switch (grade) {
        case 'S': return 10;
        case 'A': return 9;
        case 'B': return 8;
        case 'C': return 7;
        case 'D': return 6;
        case 'E': return 5;
        case 'F': return 0;
        default: return 0;
    }
}

void addStudent() {
    FILE *file = fopen(FILENAME, "ab");
    if (!file) {
        perror("Failed to open file");
        return;
    }

    Student student;
    printf("Enter roll number: ");
    scanf("%s", student.rollno);
    printf("Enter name: ");
    scanf(" %[^\n]", student.name);
    printf("Enter department: ");
    scanf(" %[^\n]", student.dept);

    printf("Enter number of courses (3 or 4): ");
    scanf("%d", &student.num_courses);
    if (student.num_courses < 3 || student.num_courses > NCOURSES)
{
        printf("Invalid number of courses.\n");
        fclose(file);
        return;
    }

    for (int i = 0; i < student.num_courses; i++) {
        printf("Enter course %d name: ", i + 1);
        scanf(" %[^\n]", student.courses[i]);
        printf("Enter credits for course %d: ", i + 1);
        scanf("%d", &student.credits[i]);
```

```
        char grade;
        printf("Enter grade for course %d (S/A/B/C/D/E/F): ", i +
1);
        scanf(" %c", &grade);
        student.grades[i] = gradeToPoint(grade);
    }

    fwrite(&student, sizeof(Student), 1, file);
    fclose(file);
    printf("Student added successfully!\n");
}

void calculateGPA(Student *student) {
    int totalCredits = 0;
    int weightedSum = 0;
    for (int i = 0; i < student->num_courses; i++) {
        totalCredits += student->credits[i];
        weightedSum += (student->credits[i] )*
(student->grades[i]);
    }
    student->gpa = (float)weightedSum / totalCredits;
}

void displayStudent(const Student *student) {
    printf("Roll No: %s\n", student->rollno);
    printf("Name: %s\n", student->name);
    printf("Department: %s\n", student->dept);
    for (int i = 0; i < student->num_courses; i++) {
        printf("Course %d: %s, Credits: %d, Grade: %d\n", i + 1,
student->courses[i], student->credits[i], student->grades[i]);
    }
    printf("GPA: %.2f\n", student->gpa);
}

void updateStudentGPA(FILE *file) {
    fseek(file, 0, SEEK_SET);
    Student student;
    while (fread(&student, sizeof(Student), 1, file)) {
        calculateGPA(&student);
        fseek(file, -sizeof(Student), SEEK_CUR);
        fwrite(&student, sizeof(Student), 1, file);
    }
}
```

```
void deleteCourse(const char *rollno, int courseIndex) {
    FILE *file = fopen(FILENAME, "r+b");
    if (!file) {
        perror("Failed to open file");
        return;
    }

    FILE *tempFile = fopen("temp.dat", "wb");
    if (!tempFile) {
        perror("Failed to open temp file");
        fclose(file);
        return;
    }

    Student student;
    while (fread(&student, sizeof(Student), 1, file)) {
        if (strcmp(student.rollno, rollno) == 0) {
            for (int i = courseIndex; i < student.num_courses - 1;
i++) {
                strcpy(student.courses[i], student.courses[i +
1]);
                student.credits[i] = student.credits[i + 1];
                student.grades[i] = student.grades[i + 1];
            }
            student.num_courses--;
        }
        fwrite(&student, sizeof(Student), 1, tempFile);
    }

    fclose(file);
    fclose(tempFile);
    remove(FILENAME);
    rename("temp.dat", FILENAME);
    printf("Course deleted successfully!\n");
}

void addCourse(const char *rollno) {
    FILE *file = fopen(FILENAME, "r+b");
    if (!file) {
        perror("Failed to open file");
        return;
    }
```

```
    FILE *tempFile = fopen("temp.dat", "wb");
    if (!tempFile) {
        perror("Failed to open temp file");
        fclose(file);
        return;
    }

    Student student;
    while (fread(&student, sizeof(Student), 1, file)) {
        if (strcmp(student.rollno, rollno) == 0) {
            if (student.num_courses >= NCOURSES) {
                printf("Cannot add more courses.\n");
                fclose(file);
                fclose(tempFile);
                return;
            }
            int index = student.num_courses;
            (student.num_courses)++;
            printf("Enter new course name: ");
            scanf(" %[^\n]", student.courses[index]);
            printf("Enter credits for new course: ");
            scanf("%d", &student.credits[index]);
            char grade;
            printf("Enter grade for new course (S/A/B/C/D/E/F):
");
            scanf(" %c", &grade);
            student.grades[index] = gradeToPoint(grade);
        }
        fwrite(&student, sizeof(Student), 1, tempFile);
    }

    fclose(file);
    fclose(tempFile);
    remove(FILENAME);
    rename("temp.dat", FILENAME);
    printf("Course added successfully!\n");
}

void updateCourseName(const char *oldName, const char *newName) {
    FILE *file = fopen(FILENAME, "r+b");
    if (!file) {
        perror("Failed to open file");
```

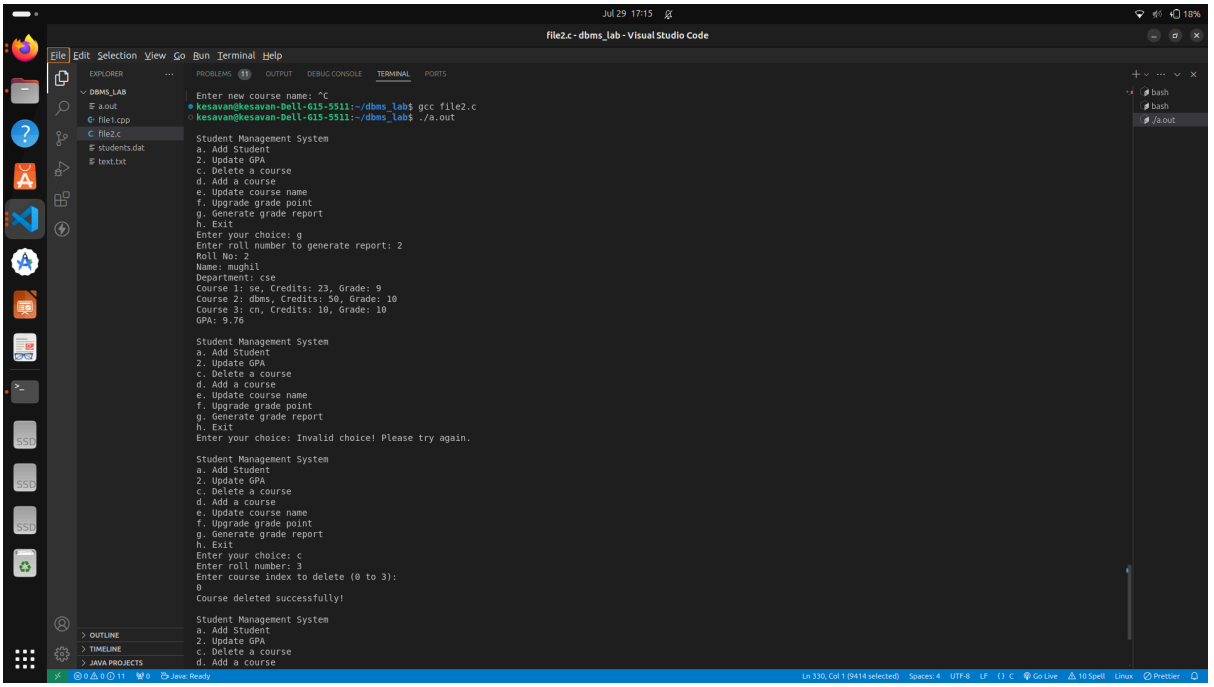
```
        return;
    }

    Student student;
    while (fread(&student, sizeof(Student), 1, file)) {
        for (int i = 0; i < student.num_courses; i++) {
            if (strcmp(student.courses[i], oldName) == 0) {
                strcpy(student.courses[i], newName);
                fseek(file, -sizeof(Student), SEEK_CUR);
                fwrite(&student, sizeof(Student), 1, file);
                break;
            }
        }
    }

    fclose(file);
    printf("Course name updated successfully!\n");
}

void upgradeGradePoint(const char *rollno) {
    FILE *file = fopen(FILENAME, "r+b");
    if (!file) {
        perror("Failed to open file");
        return;
    }

    FILE *tempFile = fopen("temp.dat", "wb");
    if (!tempFile) {
        perror("Failed to open temp file");
        fclose(file);
        return;
    }

    Student student;
    while (fread(&student, sizeof(Student), 1, file)) {
        if (strcmp(student.rollno, rollno) == 0) {
            for (int i = 0; i < student.num_courses; i++) {
                if (student.grades[i] == 7) {
                    student.grades[i] = 8; // Upgrade from 7 to 8
                }
            }
        }
        fwrite(&student, sizeof(Student), 1, tempFile);
```

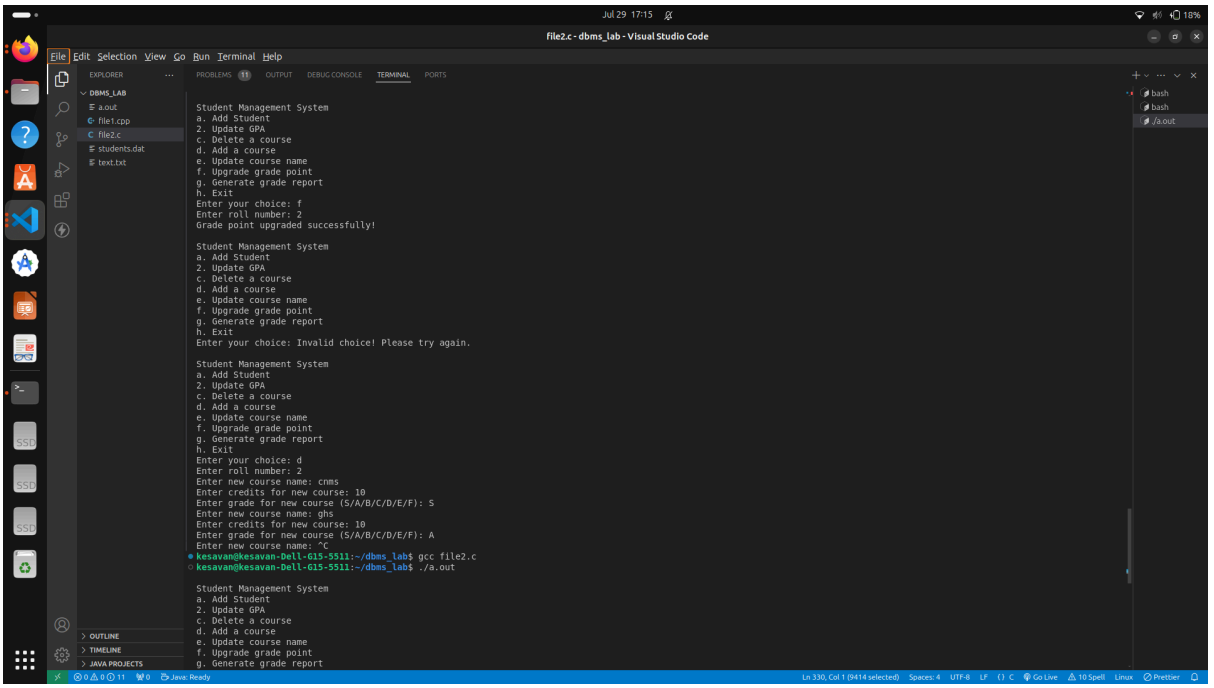
```
    }

    fclose(file);
    fclose(tempFile);
    remove(FILENAME);
    rename("temp.dat", FILENAME);
    printf("Grade point upgraded successfully!\n");
}

void generateGradeReport(const char *rollno) {
    FILE *file = fopen(FILENAME, "rb");
    if (!file) {
        perror("Failed to open file");
        return;
    }

    Student student;
    while (fread(&student, sizeof(Student), 1, file)) {
        if (strcmp(student.rollno, rollno) == 0) {
            displayStudent(&student);
            fclose(file);
            return;
        }
    }

    fclose(file);
    printf("Student with roll number %s not found.\n", rollno);
}

void menu() {
    char choice;
    char rollno[20];
    char oldCourseName[LEN];
    char newCourseName[LEN];

    while (1) {
        printf("\nStudent Management System\n");
        printf("a. Add Student\n");
        printf("2. Update GPA \n");
        printf("c. Delete a course\n");
        printf("d. Add a course\n");
        printf("e. Update course name\n");
        printf("f. Upgrade grade point\n");
```

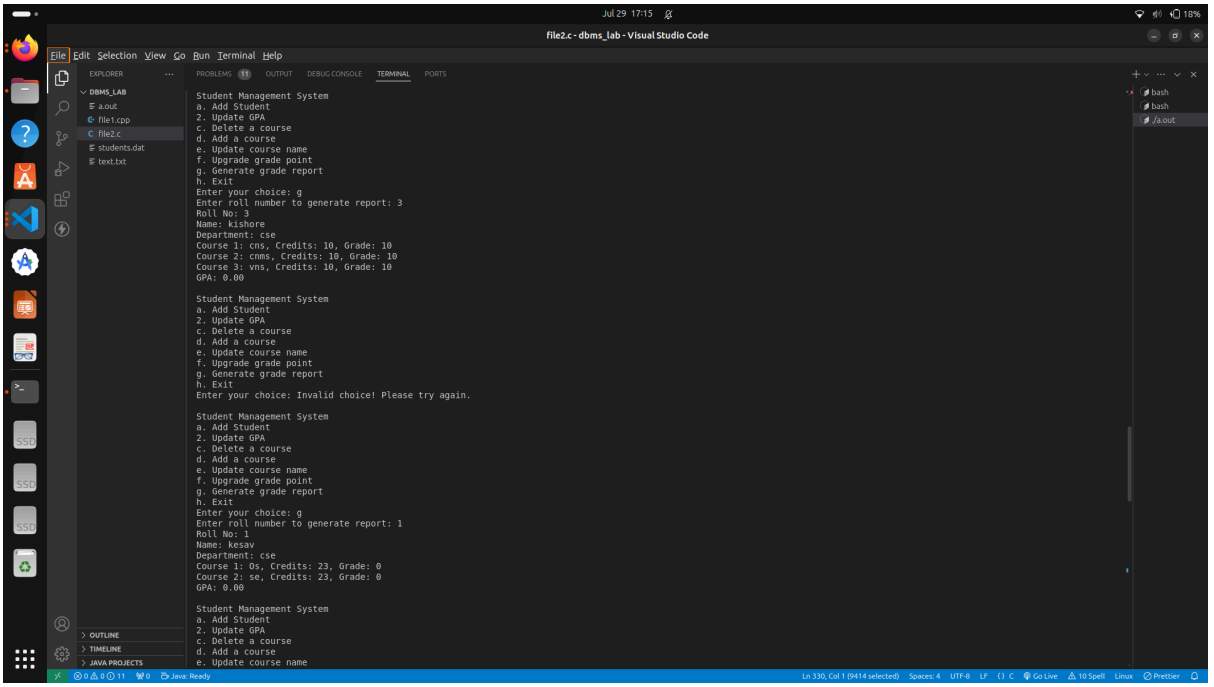
```
        printf("g. Generate grade report\n");
        printf("h. Exit\n");
        printf("Enter your choice: ");
        scanf("%c", &choice);

        switch (choice) {
            case 'a':
                addStudent();
                break;
            case 'b':
                {
                    FILE *file = fopen(FILENAME, "r+b");
                    if (file) {
                        updateStudentGPA(file);
                        fclose(file);
                    }
                }
                break;
            case 'c':
                printf("Enter roll number: ");
                scanf("%s", rollno);
                printf("Enter course index to delete (0 to 3): ");
                int index;
                scanf("%d", &index);
                deleteCourse(rollno, index);
                break;
            case 'd':
                printf("Enter roll number: ");
                scanf("%s", rollno);
                addCourse(rollno);
                break;
            case 'e':
                printf("Enter old course name: ");
                scanf(" %[^\n]", oldCourseName);
                printf("Enter new course name: ");
                scanf(" %[^\n]", newCourseName);
                updateCourseName(oldCourseName, newCourseName);
                break;
            case 'f':
                printf("Enter roll number: ");
                scanf("%s", rollno);
                upgradeGradePoint(rollno);
                break;
```

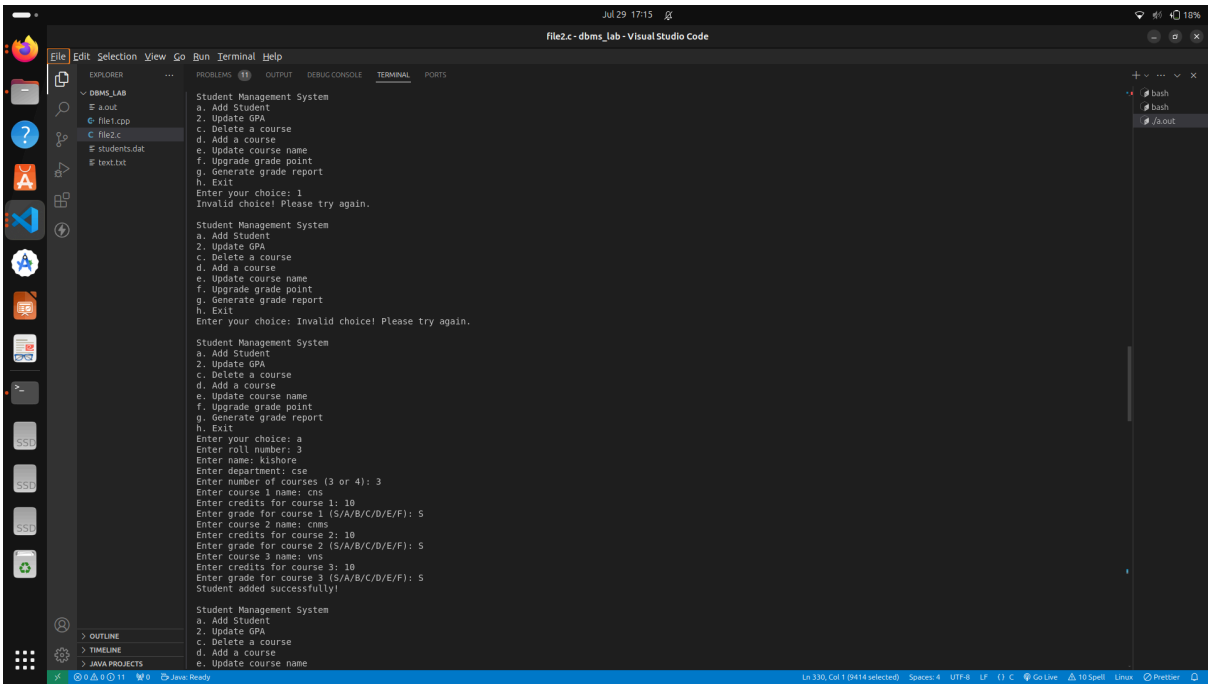
```
            case 'g':
                printf("Enter roll number to generate report: ");
                scanf("%s", rollno);
                generateGradeReport(rollno);
                break;
            case 'h':
                exit(0);
            default:
                printf("Invalid choice! Please try again.\n");
        }
    }
}

int main() {
    menu();
    return 0;
}
```

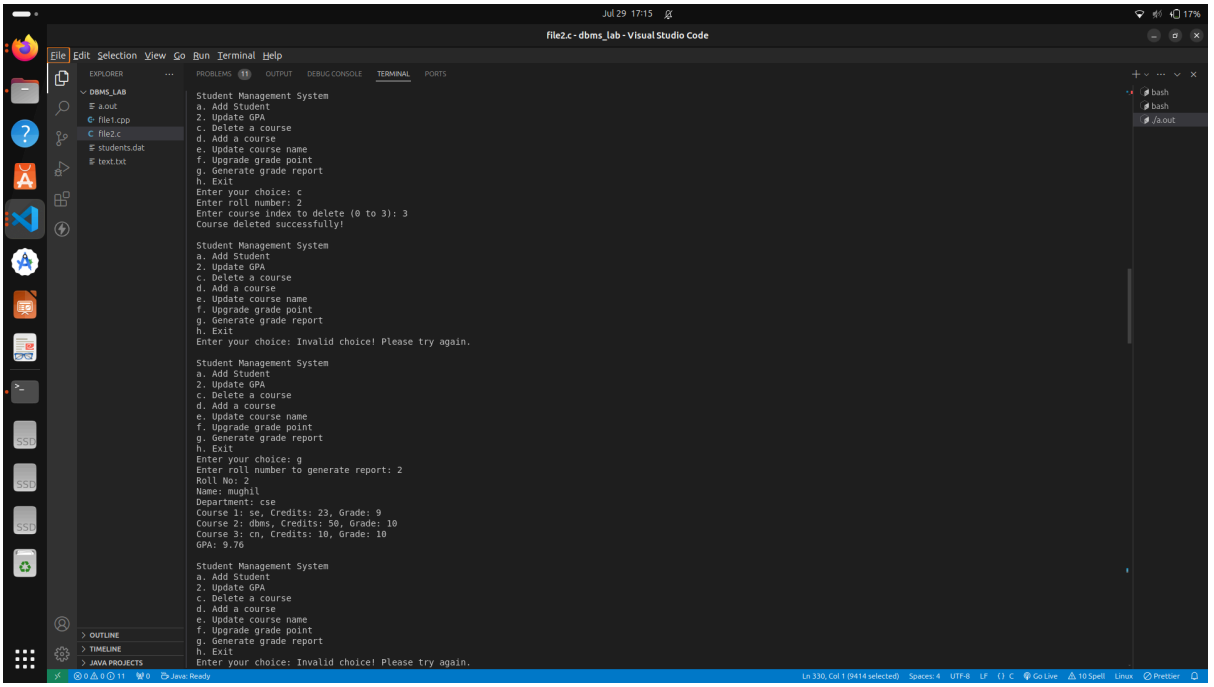
Outputs :

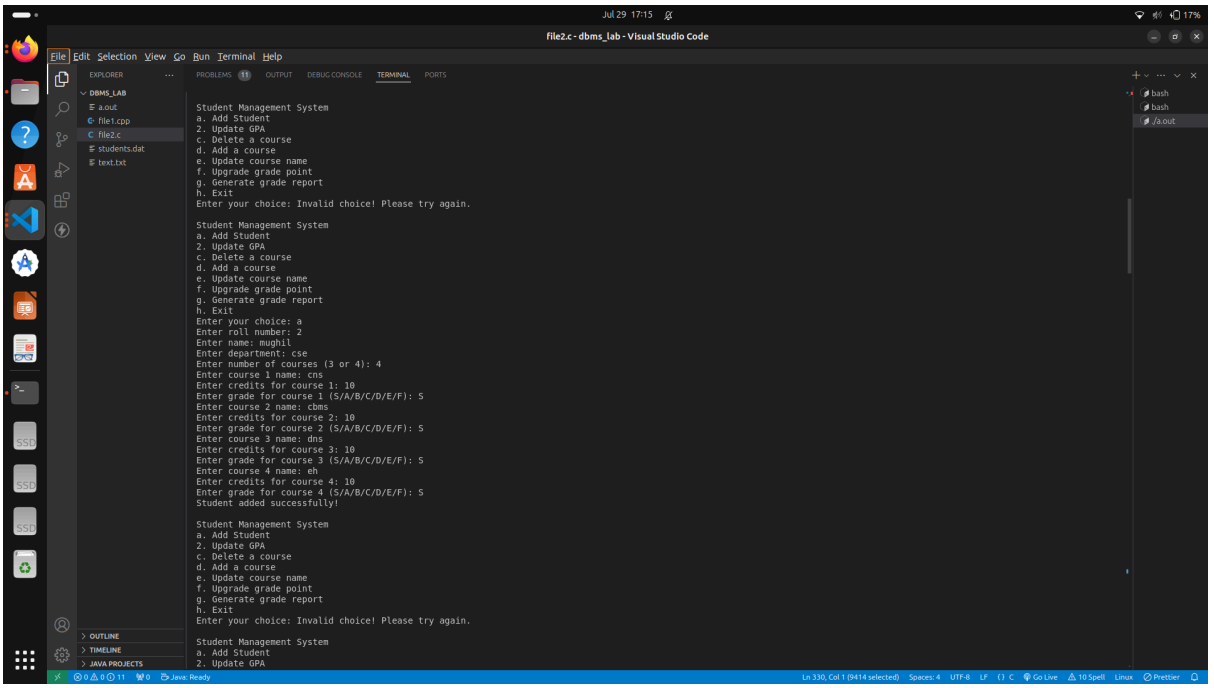

Jul 29 17:15
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Enter new course name: ^C
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ gcc file2.c
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ ./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: g
Enter roll number to generate report: 2
Roll No: 2
Name: mughil
Department: cse
Course 1: se, Credits: 23, Grade: 9
Course 2: dbms, Credits: 50, Grade: 10
Course 3: cn, Credits: 10, Grade: 10
GPA: 9.76
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: c
Enter roll number: 3
Enter course index to delete (0 to 3):
0
Course deleted successfully!
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
bash
bash
./a.out
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected) Spaces: 4 UTF-8 LF C Go Live 10 Spell Linux Prettier

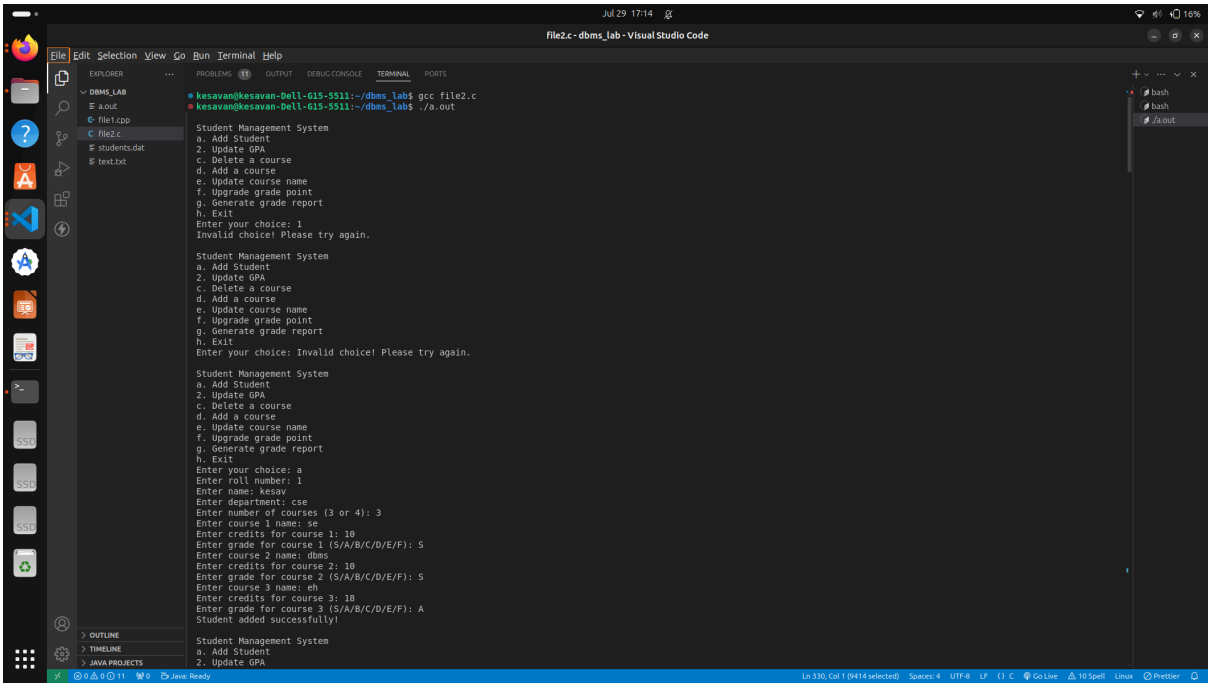

Jul 29 17:15
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: f
Enter roll number: 2
Grade point upgraded successfully!
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: d
Enter roll number: 2
Enter new course name: cnms
Enter credits for new course: 10
Enter grade for new course (S/A/B/C/D/E/F): S
Enter new course name: ghs
Enter credits for new course: 10
Enter grade for new course (S/A/B/C/D/E/F): A
Enter new course name: ^C
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ gcc file2.c
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ ./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
bash
bash
./a.out
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected) Spaces: 4 UTF-8 LF C Go Live 10 Spell Linux Prettier

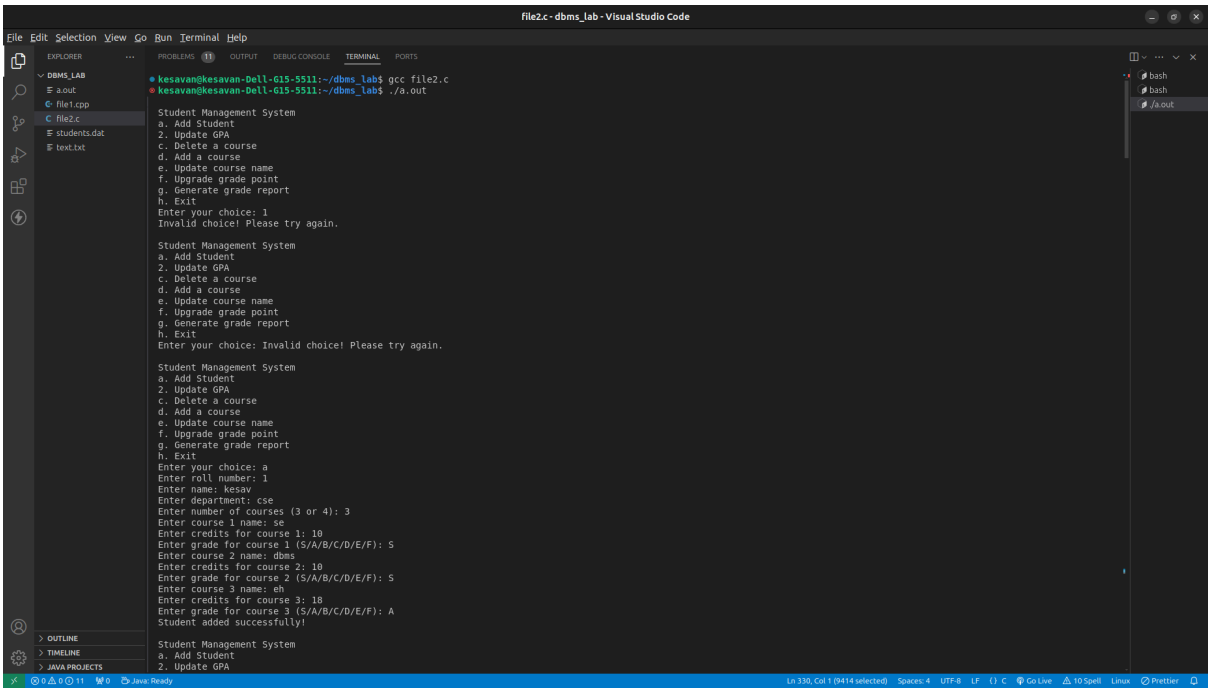

Jul 29 17:15
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash
bash
./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: g
Enter roll number to generate report: 3
Roll No: 3
Name: kishore
Department: cse
Course 1: cns, Credits: 10, Grade: 10
Course 2: cnms, Credits: 10, Grade: 10
Course 3: vns, Credits: 10, Grade: 10
GPA: 0.00
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: g
Enter roll number to generate report: 1
Roll No: 1
Name: kesav
Department: cse
Course 1: Os, Credits: 23, Grade: 0
Course 2: se, Credits: 23, Grade: 0
GPA: 0.00
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected) Spaces: 4 UTF-8 LF C Go Live 10 Spell Linux Prettier


Jul 29 17:15
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash
bash
./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: 1
Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: a
Enter roll number: 3
Enter name: kishore
Enter department: cse
Enter number of courses (3 or 4): 3
Enter course 1 name: cns
Enter credits for course 1: 10
Enter grade for course 1 (S/A/B/C/D/E/F): S
Enter course 2 name: cnms
Enter credits for course 2: 10
Enter grade for course 2 (S/A/B/C/D/E/F): S
Enter course 3 name: vns
Enter credits for course 3: 10
Enter grade for course 3 (S/A/B/C/D/E/F): S
Student added successfully!
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected) Spaces: 4 UTF-8 LF C Go Live 10 Spell Linux Prettier


Jul 29 17:15
17%
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
bash
bash
./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: c
Enter roll number: 2
Enter course index to delete (0 to 3): 3
Course deleted successfully!
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: g
Enter roll number to generate report: 2
Roll No: 2
Name: mughil
Department: cse
Course 1: se, Credits: 23, Grade: 9
Course 2: dbms, Credits: 50, Grade: 10
Course 3: cn, Credits: 10, Grade: 10
GPA: 9.76
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected)
Spaces: 4
UTF-8
LF
C
Go Live
10 Spell
Linux
Prettier


Jul 29 17:15
17%
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
bash
bash
./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: a
Enter roll number: 2
Enter name: mughil
Enter department: cse
Enter number of courses (3 or 4): 4
Enter course 1 name: cns
Enter credits for course 1: 10
Enter grade for course 1 (S/A/B/C/D/E/F): S
Enter course 2 name: cbms
Enter credits for course 2: 10
Enter grade for course 2 (S/A/B/C/D/E/F): S
Enter course 3 name: dns
Enter credits for course 3: 10
Enter grade for course 3 (S/A/B/C/D/E/F): S
Enter course 4 name: eh
Enter credits for course 4: 10
Enter grade for course 4 (S/A/B/C/D/E/F): S
Student added successfully!
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected)
Spaces: 4
UTF-8
LF
C
Go Live
10 Spell
Linux
Prettier


Jul 29 17:14
16%
file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ gcc file2.c
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ ./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: 1
Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: a
Enter roll number: 1
Enter name: kesav
Enter department: cse
Enter number of courses (3 or 4): 3
Enter course 1 name: se
Enter credits for course 1: 10
Enter grade for course 1 (S/A/B/C/D/E/F): S
Enter course 2 name: dbms
Enter credits for course 2: 10
Enter grade for course 2 (S/A/B/C/D/E/F): S
Enter course 3 name: eh
Enter credits for course 3: 18
Enter grade for course 3 (S/A/B/C/D/E/F): A
Student added successfully!
Student Management System
a. Add Student
2. Update GPA
bash
bash
./a.out
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected)
Spaces: 4
UTF-8
LF
C
Go Live
10 Spell
Linux
Prettier


file2.c - dbms_lab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DBMS_LAB
a.out
file1.cpp
file2.c
students.dat
text.txt
PROBLEMS 11
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ gcc file2.c
kesavan@kesavan-Dell-G15-5511:~/dbms_lab$ ./a.out
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: 1
Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: Invalid choice! Please try again.
Student Management System
a. Add Student
2. Update GPA
c. Delete a course
d. Add a course
e. Update course name
f. Upgrade grade point
g. Generate grade report
h. Exit
Enter your choice: a
Enter roll number: 1
Enter name: kesav
Enter department: cse
Enter number of courses (3 or 4): 3
Enter course 1 name: se
Enter credits for course 1: 10
Enter grade for course 1 (S/A/B/C/D/E/F): S
Enter course 2 name: dbms
Enter credits for course 2: 10
Enter grade for course 2 (S/A/B/C/D/E/F): S
Enter course 3 name: eh
Enter credits for course 3: 18
Enter grade for course 3 (S/A/B/C/D/E/F): A
Student added successfully!
Student Management System
a. Add Student
2. Update GPA
bash
bash
./a.out
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Ln 330, Col 1 (9414 selected)
Spaces: 4
UTF-8
LF
C
Go Live
10 Spell
Linux
Prettier

2.

Create a Student schema using the student details given in Q.No.1 and execute the following
basic queries.
Note: When defining the schema, exclude the following columns: Course_credit and Course_grade for all the courses.
Make sure you have the following constraints: Course is declared in char datatype.
DoB should be in date (dd/mm/yyyy) format. Provide a not-null constraint for dob.
Email should have the following format: xxx@nitt.edu

Insert at least 5 student records into the Student table.
b. Delete Course2 and Course3 attributes from the Student table.
c. Insert two new columns DoB and email into the Student table.
d. Change Course1 datatype to varchar2.
e. Update the column name ‘Std_rollno’ to ‘Std_rno’.
f. Update all student records who pursue a course named “DBMS” to “OS”.
g. Delete a student record with student name starting with letter ‘S’.
h. Display all records in which a student has born after the year 2005.
i. Simulate RENAME, COMMENT, TRUNATE and DROP.

Answers :
a)

```
CREATE TABLE Student (
        Std_rollno INT PRIMARY KEY,
        Std_name VARCHAR(50),
        Dept VARCHAR(10),
        Course1 CHAR(10),
        Course2 CHAR(10),
        Course3 CHAR(10),
        Course4 CHAR(10),
        dob DATE NOT NULL,
        email VARCHAR(50) CHECK (email LIKE '%@nitt.edu')
);
```

b)

```
ALTER TABLE Student DROP COLUMN Course2;
ALTER TABLE Student DROP COLUMN Course3;
```

c)

```
ALTER TABLE Student ADD COLUMN dob;
ALTER TABLE Student ADD COLUMN email;
```

d)

```
ALTER TABLE Student MODIFY COLUMN Course1 VARCHAR(2);
```

e)

```
ALTER TABLE Student MODIFY COLUMN Course1 VARCHAR(2);
```

f)

```
UPDATE Student SET Course1 = 'OS' WHERE Course1 = 'DBMS';
```

g)

```
DELETE FROM Student WHERE Std_name LIKE 'S%';
```

h)

```
SELECT * FROM Student WHERE YEAR(dob) > 2005;
```

i)

```
DROP TABLE Student;

TRUNCATE TABLE Student;
```