Instruction and important Note:

1. **Teamwork and Composition:**
   - This project is a collaborative effort, and each team can consist of a maximum of three students.
   - The work presented must be original, and all code will undergo scrutiny for plagiarism and cheating.
2. **Presentation and Class Attendance:**
   - Students are required to present their work in class on **Thursday, December 07**. The presentation should include a technical explanation of specific parts of the code, detailing the implementation process.
   - Prepare presentation slides to visually support and enhance your explanation.
   - Teams that do not present their work will receive a 30% deduction from their project grade. Additionally, individual penalties will be imposed for absent team members on the presentation day.
3. **Project Grading:**
   - The project contributes 10% to the total grade. However, teams that exhibit exceptional implementation and design may earn an additional 5% as extra credit.
4. **Submission Guidelines:**
   - Source code files that are compiled and run without errors.
   - A concise report explaining the code, including sample outputs and function descriptions.
5. **Submission Deadline:**
   - The final submission deadline is **December 10**. However, an initial version of the project, comprising compiled and runnable code, must be submitted by **December 03**. This initial submission should be made even if the project is not yet complete.
6. **Team Registration:**
   - Students must register their teams on Georgia View under the "Project" section by the end of this week.

Project Detail

Objective:

This project aims to apply the concepts covered in our course to address real-world challenges.

Problem Scenario:

You have been tasked with developing a Java program to manage student information, schedules, and grades. The primary objectives include designing two classes: one for Student Personal Information and another for Course Information.

Student Personal Information Class:

This class should include the following attributes:

1. Student ID
2. First Name
3. Last Name
4. Birth Date

5. Mobile Number
6. Email ID
7. Gender
8. Address

Course Information Class:

This class should have these attributes:

1. Course Number
2. Course Name
3. Credit Number
4. Prerequisite Course

The project also requires the use of two text files to store relevant data:

1. Student Schedule Text File: This file should store this data:
    1. Semester
    2. Student ID
    3. Course Number
    4. Days
2. Student Grade Text File: This file should store this data:
    1. Student ID
    2. Course Number
    3. Grade (in letter form)

Project Tasks:

To accomplish this project, you should follow these steps:

1. Create Java Classes:
   Implement the necessary methods and variables for the Student Personal Information and Course Information classes.
1. Generate Sample Data:
   Create objects for each class to represent sample data.
2. Create Text Files:
   Populate the text files with sample data.
3. Develop a Java Program:
   Create a Java program that offers the following functionalities:
    a. View and Search Data:
        - Display all student information.
        - Search and display specific student(s) information using criteria such as ID, Names, or Gender.
        - Display all course information.
        - Search and display specific courses using criteria like Course Number or Credit.
        - Export all student information to a file.
        - Export all course information to a file.
        - Read and process Student Schedule and Student Grade files.

- Print Student Schedule information.
- Search and print specific student using student ID.
- Print Student Grade information.
- Search and print specific student's grades using Student ID.

b. Add New Information:
   - Enter new student information, ensuring each student has a unique student ID and email address.
   - Enter new course information, ensuring each course has a unique course number.
   - Enter student schedules, ensuring that both student and course objects exist.
   - Enter student grades, ensuring that both student and course objects exist.

c. Delete Data:
   - Allow users to delete a student's information.
   - Allow users to delete course information.
   - Allow users to remove entries from the student schedule.

d. Edit Information:
   - Enable users to edit student information.
   - Allow users to modify course information.

Note:

Use two-dimensional arrays of size 100 to store objects as tables to manage the data effectively.

Sample Data Provided in below tables:

- Student Personal information Sample Data:

Student ID	First Name	Last Name	Birth Date	Mobile Number	Email ID	Gender	Address
202301	John	Doe	3/12/1995	123-456-7890	john@email.com	Male	123 Main St, Atlanta, GA
202302	Alice	Johnson	8/25/1997	987-654-3210	alice@email.com	Female	456 Elm St, Atlanta, GA
202303	Bob	Smith	11/5/1996	555-123-7890	bob@email.com	Male	789 Oak St, Atlanta, GA

- Course Information Sample Data:

Course Number	Course Name	Credit Number	Prerequisite Course
M101	Calculus I	3	None
M102	Calculus II	3	M101
P201	Physics I	4	Math 101
P202	Physics II	4	P201
C201	Chemistry I	3	Math 101
C301	Advance Chemistry	3	M101, C201

- Student Schedule Sample Data:

Semester	Student ID	Course Number	Date

Fall 2023	202301	M102	MW
Fall 2023	202301	P201	TH
Fall 2023	202302	M102	MW
Fall 2023	202303	C301	MW

- Student Grade Sample Data:

Student ID	Course Number	Grade (letter)
202301	M101	A
202301	P201	C
202302	M101	A
202303	M101	B