### \*\*Detailed Report on Student Management System\*\*

#### \*\*Overview\*\*

The Student Management System is a comprehensive application designed to manage and streamline student data, course information, and grading details. It includes functionalities for adding, displaying, and updating student information, course details, and grades, catering to both administrative and teaching roles.

### \*\*Class Definitions and Functionalities\*\*

#### \*\*1. Student Class\*\*

- \*\*Attributes:\*\*

- \*\*Protected:\*\*

- `name`: List of student names.

- `father\_name`: List of student fathers' names.

- `roll\_no`: List of student roll numbers.

- `degree`: List of student degree programs.

- `semester`: List of student semesters.

- `total\_students`: Total number of students.

- \*\*Methods:\*\*

- \*\*Public:\*\*

- `getTotalStudents()`: Returns the total number of students.

- `getRollNoByIndex(index)`: Returns the roll number of the student at the given index.

- `getNameByIndex(index)`: Returns the name of the student at the given index.

- `getFatherNameByIndex(index)`: Returns the father's name of the student at the given index.

- `getDegreeByIndex(index)`: Returns the degree of the student at the given index.

- `getSemesterByIndex(index)`: Returns the semester of the student at the given index.

- `display()`: Displays the details of all students.

- \*\*Constructor:\*\*

- Prompts the user to input the total number of students and their respective details.

- \*\*Friend Class:\*\*

- `UpdateManager` class, which has access to private and protected members of the `Student` class for updating student information.

#### \*\*2. Course Class (Derived from Student)\*\*

- \*\*Attributes:\*\*

- \*\*Protected:\*\*

- `student\_total\_subjects`: List of total subjects per student.

- `subject\_name`: Nested list of subject names for each student.

- `course\_code`: Nested list of course codes for each subject of each student.

- \*\*Methods:\*\*

- \*\*Public:\*\*

- `gettotal\_subjects(index)`: Returns the total number of subjects for the student at the given index.

- `getsubject\_name(index1, index2)`: Returns the subject name for the given student and subject indices.

- `getcourse\_code(index1, index2)`: Returns the course code for the given student and subject indices.

- \*\*Constructor:\*\*

- Prompts the user to input the total number of subjects, subject names, and course codes for each student.

- \*\*Friend Class:\*\*

- `UpdateManager` class.

#### \*\*3. Grades Class (Derived from Course)\*\*

- \*\*Attributes:\*\*

- \*\*Protected:\*\*

- `grade`: Nested list of grades for each subject of each student.

- `obtained\_number`: Nested list of obtained marks for each subject of each student.

- `total\_number`: Nested list of total marks for each subject of each student.

- `percentage`: Nested list of percentages for each subject of each student.

- \*\*Methods:\*\*

- \*\*Public:\*\*

- `get\_grades(index1, index2)`: Returns the grade for the given student and subject indices.

- `displayGrades()`: Displays the grades for all students.

- \*\*Constructor:\*\*

- Prompts the user to input the obtained and total marks for each subject of each student, calculates the percentage, and assigns grades based on the percentage.

- \*\*Friend Class:\*\*

- `UpdateManager` class.

#### \*\*4. UpdateManager Class\*\*

- \*\*Static Methods:\*\*

- `updateStudentData(Student& student)`: Updates student details (name, father's name, roll number, degree, semester) for a student based on the roll number.

- `updateCourseData(Course& course)`: Updates course details (subjects and course codes) for a student based on the roll number.

- `updateStudentNumbers(Grades& grades)`: Updates marks and grades for a student based on the roll number.

### \*\*Program Flow and User Interaction\*\*

#### \*\*Main Functionality\*\*

The main function initializes the program, displaying a welcome message and entering an interactive loop that prompts the user to choose between administrative or teaching roles. The user is presented with a menu of options depending on their role.

#### \*\*Admin Options:\*\*

1. \*\*Add Student Data:\*\*

- The admin inputs details for each student, including their personal information and course details.

2. \*\*Display Student Data:\*\*

- The admin can display details for all students or a specific student by roll number.

3. \*\*Update Student Data:\*\*

- The admin updates a student's personal information using their roll number.

4. \*\*Update Course Data:\*\*

- The admin updates the course details for a student using their roll number.

5. \*\*Exit:\*\*

- Exits the admin menu.

#### \*\*Teacher Options:\*\*

1. \*\*Add Student Grades:\*\*

- The teacher inputs marks for each subject of a student, and grades are calculated and stored.

2. \*\*Display Student Grades:\*\*

- The teacher can display grades for all students or a specific student by roll number.

3. \*\*Update Student Numbers:\*\*

- The teacher updates the marks and grades for a student using their roll number.

4. \*\*Exit:\*\*

- Exits the teacher menu.

### \*\*Detailed Functionalities\*\*

#### \*\*Adding Student Data:\*\*

- The admin enters the number of students and their personal details, such as name, father's name, roll number, degree, and semester.

- Subsequently, the admin enters the course details for each student, including the total number of subjects, subject names, and course codes.

#### \*\*Displaying Student Data:\*\*

- The admin can display the details of all students or a specific student by entering the roll number.

- This includes personal details like name, father's name, roll number, degree, and semester.

#### \*\*Updating Student Data:\*\*

- The admin can update a student's details by entering the roll number.

- Options are provided to update specific details like name, father's name, roll number, degree, and semester.

#### \*\*Updating Course Data:\*\*

- The admin can update the course details for a student by entering the roll number.

- This includes entering new subject names and course codes for the student.

#### \*\*Adding Student Grades:\*\*

- The teacher enters the obtained and total marks for each subject of a student.

- Grades are calculated based on the percentage of marks obtained and stored in the system.

#### \*\*Displaying Student Grades:\*\*

- The teacher can display grades for all students or a specific student by entering the roll number.

- The grades for each subject are shown along with the corresponding percentage and marks.

#### \*\*Updating Student Numbers:\*\*

- The teacher can update the obtained marks and grades for a student by entering the roll number.

### \*\*Conclusion\*\*

The Student Management System provides a robust framework for managing student information, course details, and grades. It ensures a streamlined and user-friendly experience for both administrative staff and teachers. By incorporating structured classes and interactive functionalities, the system effectively handles the complex tasks of student data management, offering a comprehensive solution for educational institutions.