**Case Study: School Management System**

**Section 1: Python Standalone Console Application**

Design and implement a standalone console application for a School Management System using Python. The application should utilize collections, object-oriented programming (OOP), and exception handling to manage students, teachers, courses, and grades.

**Requirements:**

1. **Student Management:**
   * Implement the functionality to add, update, and delete student records.
   * Each student should have attributes such as student\_id, name, date\_of\_birth, and grade\_level.
2. **Teacher Management:**
   * Implement the functionality to manage teachers.
   * Each teacher should have attributes such as teacher\_id, name, contact\_number, and subject\_taught.
3. **Course Management:**
   * Implement the functionality to handle courses.
   * Each course should have attributes such as course\_id, name, teacher\_id, and maximum\_students.
4. **Grade Management:**
   * Implement the functionality to manage student grades for courses.
   * Each grade should have attributes such as student\_id, course\_id, and grade.

**Business Functionalities:**

1. **Manage Students:**
   * Create a class Student with attributes student\_id, name, date\_of\_birth, and grade\_level.
   * Implement methods to add a new student, update student details, and delete a student from the system.
2. **Manage Teachers:**
   * Create a class Teacher with attributes teacher\_id, name, contact\_number, and subject\_taught.
   * Implement methods to add a new teacher, update teacher details, and delete a teacher.
3. **Manage Courses:**
   * Create a class Course with attributes course\_id, name, teacher\_id, and maximum\_students.
   * Implement methods to add a new course, update course details, and delete a course.
4. **Manage Grades:**
   * Create a class Grade with attributes student\_id, course\_id, and grade.
   * Implement methods to assign grades to students for courses.

**Section 2: MySQL Database Management**

Design a MySQL database schema to support the School Management System and provide solutions for the problem statements.

**Table Structures:**

1. **Students Table:**

* student\_id: INT, Primary Key
* name: VARCHAR(255)
* date\_of\_birth: DATE
* grade\_level: INT

1. **Teachers Table:**

* teacher\_id: INT, Primary Key
* name: VARCHAR(255)
* contact\_number: VARCHAR(15)
* subject\_taught: VARCHAR(255)

1. **Courses Table:**

* course\_id: INT, Primary Key
* name: VARCHAR(255)
* teacher\_id: INT, Foreign Key References Teachers(teacher\_id)
* maximum\_students: INT

1. **Grades Table:**

* grade\_id: INT, Primary Key
* student\_id: INT, Foreign Key References Students(student\_id)
* course\_id: INT, Foreign Key References Courses(course\_id)
* grade: VARCHAR(2)

**Problem Statements:**

* Write a query to find the students who have the highest grade in a specific course.
* Write a query to find the teachers who are teaching the maximum number of courses.
* Write a query to find the courses that have not reached the maximum number of students.
* Write a query to find the average grade for each course.
* Write a query to find the students who have failed a course.