**Case Study: Employee Performance Tracking System**

**Section 1: Python Standalone Console Application**

Design and implement a standalone console application for an Employee Performance Tracking System using Python. The application should utilize collections, object-oriented programming (OOP), and exception handling to manage employee performance records.

**Requirements:**

1. **Employee Management**:
   * Implement the functionality to add, update, and delete employee records.
   * Each employee should have attributes such as employee\_id, name, department, designation, and performance\_score.
2. **Performance Tracking**:
   * Implement the functionality to update an employee’s performance score.
   * Implement the functionality to generate performance reviews for employees.
3. **Reporting**:
   * Implement the functionality to generate a report of top-performing employees (performance\_score > 85).

**Business Functionalities:**

1. **Add/Update/Delete Employees**:
   * Create a class Employee with attributes employee\_id, name, department, designation, and performance\_score.
   * Implement methods to add a new employee, update existing employee details, and delete an employee from the system.
2. **Update Performance Score**:
   * Implement a method to update the performance score of an employee by employee\_id.
3. **Top Performers Report**:
   * Implement a method to generate a list of employees with performance scores greater than 85.

**Section 2: MySQL Database Management**

Design a MySQL database schema to support the Employee Performance Tracking System and provide solutions for the problem statements.

**Table Structures:**

1. **Employees Table**:
   * employee\_id: INT, Primary Key
   * name: VARCHAR(100)
   * department: VARCHAR(50)
   * designation: VARCHAR(50)
   * performance\_score: INT
2. **PerformanceReviews Table**:
   * review\_id: INT, Primary Key
   * employee\_id: INT, Foreign Key References Employees(employee\_id)
   * review\_date: DATE
   * review\_comments: TEXT

**Problem Statements:**

1. Write a query to find the average performance score of employees in each department.
2. Write a query to find the names and performance scores of employees along with their departments.
3. Write a query to find the names of employees who have never had a performance review.
4. Write a query to find the departments that have more than 5 employees with performance scores greater than 85.
5. Write a query to find the names and designations of employees who have had more than 3 performance reviews.