Points: 100/100   **Due Date:** Nov. 7th, 2018

A software development firm had been asked to develop access control mechanism for a Database Management System (DBMS). In this assignment you are asked to perform the task of the security designer for that firm, i.e., designing and implementing the access control mechanism for the DBMS.

The DBMS Access Control Mechanism (DACM) will receive SQL commands and either accept and perform the command or deny the command. SQL commands can be divided into four categories. The following are the four categories and the minimal commands that you need for each category:

- **DDL (Data Definition Language):** 1) Create; and 2) Drop.
- **DML (Data Manipulation Language):** 1) Select; 2) Insert; and 3) Delete.
- **TCL (Transaction Control Language):** 1) Commit; and 2) Rollback.
- **DCL (Data Control Language):** 1) Grant; and 2) Revoke.

The database system has three types of users:
1. **Security Officers:** Manges users and passwords. In addition, they perform backup and recovery operations.
2. **Regular Users:** Users who are allowed to perform only DML operations.
3. **Administrators:** Perform the tasks of all the above.

Requirements
1. Implement a generic Access Control Matrix class that implements all the Access Control System Commands in Table 4.3 of the book (slides).
2. Create the required objects and/or structures to implement the system described above.
3. Show the required initialization for the object of the Access Control Matric class.
4. Determine the type of this access control mechanism (DAC, MAC, RBAC, etc).
5. Demonstrate the abilities of the system to:
   a. Add/remove users.
   b. Authenticate users.
   c. Create and drop databases and tables (Table structures and contents are not important).
   d. Accept/Deny the SQL command received by the users.
6. It is essential to utilize the class implemented in requirement #1.

Note 1: You are not required to implement the whole DBMS system, only the sufficient part to perform the demonstration.

Note 2: This is a design assignment, so different correct implementations are possible.

Note 3: You can use any language that supports OOP.