**Name: Aditya Sharad Deshmukh**

**MIS: 612203036**

**Assignment 3: SQL DDL and updates**

Statement : Write the DDL and DML statements for the following.

* Each offering of a course (i.e. a section) can have many Teaching assistants; each teaching assistant is a student. Extend the existing schema(Add/Alter tables) to accommodate this requirement.

->ALTER TABLE teaching\_assistant DROP FOREIGN KEY teaching\_assistant\_ibfk\_1;

->ALTER TABLE teaching\_assistant DROP FOREIGN KEY teaching\_assistant\_ibfk\_2;

->ALTER TABLE teaching\_assistant DROP PRIMARY KEY;

->ALTER TABLE teaching\_assistant

-> ADD CONSTRAINT teaching\_assistant\_fk

-> FOREIGN KEY (course\_id, sec\_id, semester, year)

-> REFERENCES section(course\_id, sec\_id, semester, year)

-> ON DELETE CASCADE;

-> ALTER TABLE teaching\_assistant

->ADD CONSTRAINT teaching\_assistant\_fk2

->FOREIGN KEY (ID) REFERENCES student(ID) ON DELETE CASCADE;

mysql> select \* from teaching\_assistant;

+-------+-----------+--------+----------+------+

| ID | course\_id | sec\_id | semester | year |

+-------+-----------+--------+----------+------+

| 00128 | BIO-101 | 1 | Summer | 2009 |

| 12345 | BIO-101 | 1 | Summer | 2009 |

+-------+-----------+--------+----------+------+

According to the existing schema, one student can have only one advisor.

* Alter the schema to allow a student to have multiple advisors and make sure that you are able to insert multiple advisors for a student.
* ->SHOW CREATE TABLE advisor;
* ->ALTER TABLE advisor DROP FOREIGN KEY advisor\_ibfk\_2;
* → ALTER TABLE advisor DROP PRIMARY KEY;
* ->ALTER TABLE advisor ADD CONSTRAINT advisor\_ibfk\_2 FOREIGN KEY (s\_ID) REFERENCES student(ID) ON DELETE CASCADE;

mysql> INSERT INTO advisor(s\_ID,i\_ID) values("00128","10101");

Query OK, 1 row affected (0.02 sec)

mysql> select \* from advisor;

+-------+-------+

| s\_ID | i\_ID |

+-------+-------+

| 00128 | 45565 |

| 12345 | 10101 |

| 23121 | 76543 |

| 44553 | 22222 |

| 45678 | 22222 |

| 76543 | 45565 |

| 76653 | 98345 |

| 98765 | 98345 |

| 98988 | 76766 |

| 00128 | 10101 |

+-------+-------+

10 rows in set (0.00 sec)

Write SQL queries on the modified schema. You will need to insert data to ensure the query results are not empty.

* Find all students who have more than 3 advisors
* →select s\_ID FROM advisor group by s\_ID having count > 3;

mysql> select s\_ID FROM advisor group by s\_ID having count > 3;

ERROR 1054 (42S22): Unknown column 'count' in 'having clause'

mysql> select s\_ID FROM advisor group by s\_ID having count(s\_ID) > 3;

+-------+

| s\_ID |

+-------+

| 00128 |

+-------+

1 row in set (0.00 sec)

mysql>

* Find all students who are co-advised by Prof. Srinivas and Prof. Ashok.
* ->mysql> SELECT S.name
* FROM student S
* LEFT JOIN
* advisor A on S.ID = A.s\_ID WHERE A.i\_ID = "10101" || A.i\_ID = "99999" GROUP BY S.ID HAVING COUNT(A.i\_ID) = 2;
* +-------+
* | name |
* +-------+
* | Zhang |
* +-------+
* Find students advised by instructors from different departments. Etc.
* mysql> SELECT S.name, COUNT(DISTINCT I.dept\_name) AS dept\_count
* -> FROM student S
* -> JOIN advisor A ON S.ID = A.s\_ID
* -> JOIN instructor I ON A.i\_ID = I.ID
* -> GROUP BY S.ID
* -> HAVING COUNT(DISTINCT I.dept\_name) >= 2;
* +---------+------------+
* | name | dept\_count |
* +---------+------------+
* | Zhang | 3 |
* | Shankar | 2 |
* +---------+------------+

Write SQL queries for the following:

* Delete all information in the database which is more than 10 years old. Add data as necessary to verify your query.

->DELETE FROM teaches WHERE year <= 2008;

DELETE FROM takes WHERE year <= 2008;

DELETE FROM section WHERE year <= 2008;

DELETE FROM teaching\_assistant WHERE year <= 2008;

* Delete the course CS 101.  Any course which has CS 101 as a prereq should remove CS 101 from its prereq set.  Create a cascade constraint to enforce the above rule, and verify that it is working.

->ALTER TABLE prereq DROP CONSTRAINT prereq\_course\_id\_fkey;

->ALTER TABLE prereq ADD CONSTRAINT prereq\_course\_id\_fkey FOREIGN KEY (course\_id) REFERENCES course(course\_id) ON DELETE CASCADE;

->DELETE FROM course WHERE course\_id = 'CS101';