CS313, Lab – 5

Rishit Saiya, 180010027

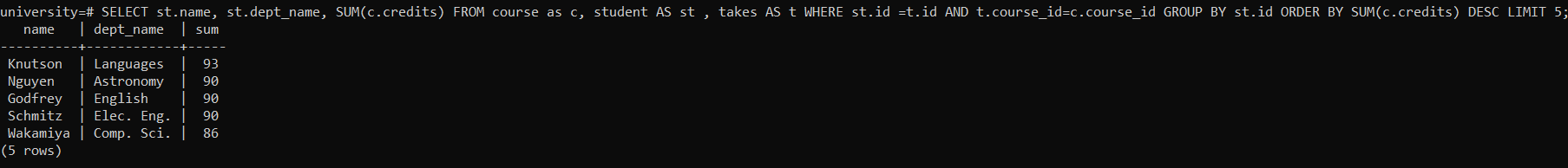
October 3, 2020

**Q.2]**

***Query:***

SELECT st.name, st.dept\_name, SUM(c.credits) FROM course as c, student AS st, takes AS t WHERE st.id =t.id AND t.course\_id=c.course\_id GROUP BY st.id ORDER BY SUM(c.credits) DESC LIMIT 5;

***Output:***



**Q.5]**

i]

***Query:*** Putting COMMIT after all three INSERT statements.

BEGIN TRANSACTION;

INSERT INTO course VALUES ('060', 'Capture The Flag', 'Comp. Sci.', 6);

INSERT INTO section VALUES ('060', '1', 'Spring', '2019', 'Gates', '707', 'K');

INSERT INTO teaches VALUES ('3200', '060', '1', 'Spring', '2019');

COMMIT TRANSACTION;

**SQL file Screenshot:**

BEGIN TRANSACTION;

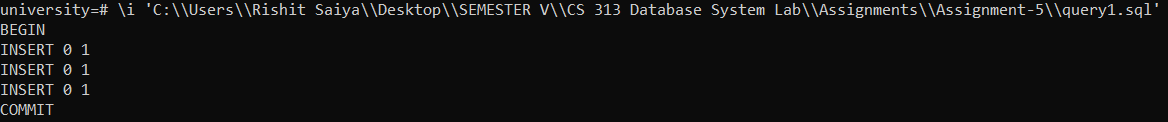
INSERT INTO course VALUES ('060', 'Capture The Flag', 'Comp. Sci.', 6);

INSERT INTO section VALUES ('060', '1', 'Spring', '2019', 'Gates', '707', 'K');

INSERT INTO teaches VALUES ('3199', '060', '1', 'Spring', '2019');

COMMIT TRANSACTION;

***Output:***



We do not get any errors here and all the tuples are inserted, because we just INSERT the tuples here and finally we commit them.

ii]

***Query:*** Put ROLLBACK after second INSERT statement and COMMIT after third INSERT statement.

BEGIN TRANSACTION;

INSERT INTO course VALUES ('061', 'Reverse Engineering', 'Comp. Sci.', 6);

INSERT INTO section VALUES ('061', '1', 'Spring', '2019', 'Gates', '707', 'K');

ROLLBACK TRANSACTION;

INSERT INTO teaches VALUES ('3199', '061', '1', 'Spring', '2019');

COMMIT TRANSACTION;

**SQL file Screenshot:**

BEGIN TRANSACTION;

INSERT INTO course VALUES ('061', 'Reverse Engineering', 'Comp. Sci.', 6);

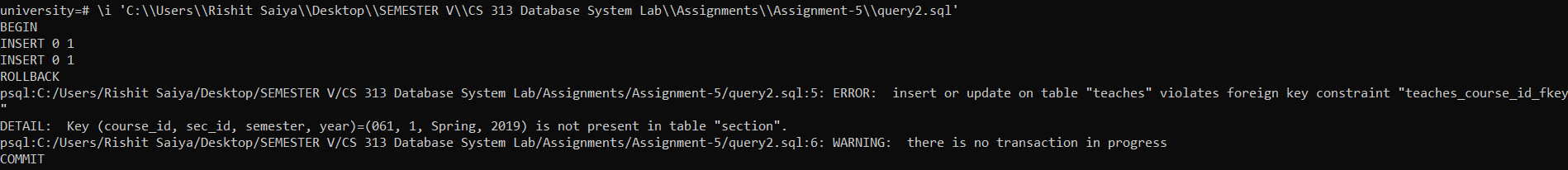
INSERT INTO section VALUES ('061', '1', 'Spring', '2019', 'Gates', '707', 'K');

ROLLBACK TRANSACTION;

INSERT INTO teaches VALUES ('3199', '061', '1', 'Spring', '2019');

COMMIT TRANSACTION;

***Output:***



Here after INSERT of first 2 tuples, we give a ROLLBACK statement, which instates an error because 3rd tuple has a foreign key-id of Course Code (061) referring to first 2 tuples which we just removed. Hence we get error for the 3rd INSERT query.

iii]

***Query:*** Put ROLLBACK after all three INSERT statements.

BEGIN TRANSACTION;

INSERT INTO course VALUES ('062', 'Binary Exploitation', 'Comp. Sci.', 6);

INSERT INTO section VALUES ('062', '1', 'Spring', '2019', 'Gates', '314', 'K');

INSERT INTO teaches VALUES ('79081', '062', '1', 'Spring', '2019');

ROLLBACK TRANSACTION;

**SQL file Screenshot:**

BEGIN TRANSACTION;

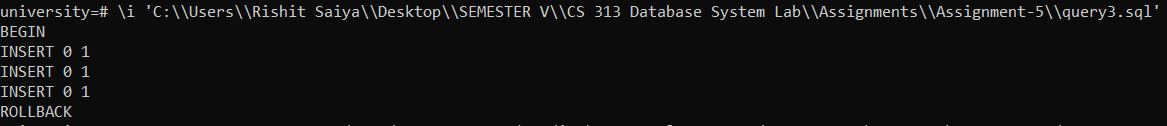
INSERT INTO course VALUES ('062', 'Binary Exploitation', 'Comp. Sci.', 6);

INSERT INTO section VALUES ('062', '1', 'Spring', '2019', 'Gates', '314', 'K');

INSERT INTO teaches VALUES ('79081', '062', '1', 'Spring', '2019');

ROLLBACK TRANSACTION;

***Output:***



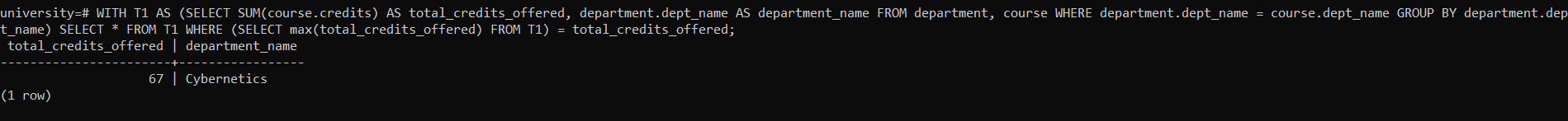
Here we don’t get any errors because we INSERT all the tuples sequentially and then remove them at the end, so there is no scope of NULL referencing as it happened in previous query. So finally all of them were added and removed at the end.

**Q.QUIZ]**

***Base Table Query:***

WITH T1 AS (SELECT SUM(course.credits) AS total\_credits\_offered, department.dept\_name AS department\_name FROM department, course WHERE department.dept\_name = course.dept\_name GROUP BY department.dept\_name) SELECT \* FROM T1 WHERE (SELECT max(total\_credits\_offered) FROM T1) = total\_credits\_offered;

**Ouput:**

******

***Materialized View:***

CREATE MATERIALIZED VIEW TotalCredits AS SELECT dept\_name, SUM(credits) AS total\_credits\_offered FROM course GROUP BY dept\_name;

***Materialized View Query:***

SELECT \* FROM TotalCredits WHERE total\_credits\_offered IN (SELECT MAX(total\_credits\_offered) FROM TotalCredits);

***Output:***

