

PROJECT 3

ASHISH SANGHVI

ashishsa

To check whether the given relation is in 3rd Normal Form we check each of the tables are in the 3rd Normal Form.

In general the properties of tables in third normal form are the following:

- 1) It has to be in the 2nd Normal Forms
- 2) The Transitive Functional Dependency Of Non-Prime Attributes on any super key must be removed.

TABLE Users:- Its in 3rd NF as non key attributes display name, account password and user type is dependent on email_id.

TABLE Professor,Staff and Student:- Its in 3rd NF as key attributes email_id is unique and referenced from user. The department number is referenced from the departments table.

TABLE department:- Its in 3NF as it contains unique department number(Primary Key) and the non key attribute department name is dependent on the department number.

TABLE program:- Its in 3NF as it contains unique program name and id. The program name is dependent on the program id. There is unique dept no referenced from departments table.

TABLE majors:- Its in 3NF as it contains unique student email id. The major department is unique dept no referenced from departments table.

TABLE prog_pursues_majors:- Its in 3NF as it contains (student email id, major dept) referenced from majors, program id from program and the combination forms a super key with no non-prime attributes. The majordept might be different from the dept major of program as student might be pursuing program that's different from his majors dept.

TABLE semester:- Its in 3NF as it contains unique combination of year and season offered which is unique and forms super key.

TABLE registers:- Its in 3NF. The register_id is Primary key and unique. There are no transitive relationships between the foreign keys and they are not dependent on each other thus does not violate the 3NF rule.

TABLE course:- Its in 3NF as course id is Primary Key and course name is dependent on course id. course dept is foreign key.

TABLE pre_req_course:- Its in 3NF. Course id and pre req course id are dependent on each other.

TABLE instructor:- Its in 3NF. Contains Instructor id which is referenced from professor_id as Foreign key and is a Primary key in this table.

TABLE courses_available:- Its in 3NF. Contains (course_id,year_offered,season_offered,session_id) as Primary key and capacity is dependent on this super key.

TABLE enroll:- Its in 3NF. Contains (register_id,course_id,year_offered,season_offered,session_id) as Primary key and subject passed and subject grade is dependent on the super key and there are no transitive relationships.

TABLE instructs:- Its in 3NF. Contains (instructor_email_id,course_id,year_offered,season_offered,session_id) all of which are foreign keys and a combination forms the Primary key and there are no transitive relationships.

TABLE feedback:- Its in 3NF. Contains (instructor_email_id,course_id,year_offered,season_offered,session_id,register_id)) all of which are foreign keys and a combination forms the Primary key and there are no transitive relationships. The feedback is dependent on the combination of these primary keys.

TABLE ta:-Its in 3NF.Contains (student_email_id,course_id,year_offered,season_offered,session_id) all of which are foreign keys and combination forms a primary key. There are no transitive relationships.

TABLE exam:-Its in 3NF. exam_id is primary key and (course_id,year_offered,season_offered,session_id) are foreign keys and are unique combination. Thus there are no transitive relationships.

TABLE student_takes_exam:-Its in 3NF. Contains (exam_id,register_id,course_id,year_offered,season_offered,session_id)) all of which are foreign keys and a combination forms the Primary key and there are no transitive relationships. The final_grade is dependent on the combination of the primary keys.

TABLE problems:- Its in 3NF. Contains problem_id as primary key. exam_id is foreign key. Scores is dependent on problem_id and there are no transitive relationships.

TABLE result:- Its in 3NF. Contains (problem_id,exam_id,register_id,course_id,year_offered,season_offered,session_id) all of which are foreign keys and a combination forms the primary key and there are no transitive relationships. The scores is dependent on the combination of the primary keys.

TABLE book:- Its in 3NF. Contains (ISBN) as primary key and (title,pages,publication_date) are dependent on the ISBN and there are no transitive relationships

TABLE author:-Its in 3NF. author_id is primary key and author_name depends on author_id.

TABLE library:- Its in 3NF. university_id is primary key.

TABLE physical_library:- library_id is primary key. university_id is foreign key. There are no transitive relationships.

TABLE copies:- book_id is primary key.(ISBN,date_of_purchase,book_id,price) are non-key attributes and its in 3NF.

TABLE writes:- Combination of (author_id,ISBN) is the primary key.

TABLE borrow:- Combination of (book_id,borrower_email_id) is primary key. The combination of (issue_date, return_date, extended_date ,extension_requested ,returned) is non key attribute and there are no transitive relationships. Its in 3NF.

Thus the Relational Data Base Table in PROJECT1 is in 3rd NORMAL FORM.

2A) An unregistered student enrolls CSE 560 in 2020 Fall

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' panel with a list of databases: employees, pa3, sakila, sys, and world. The main window shows a SQL script with the following statements:

```
7 • urse_dept VALUES('CSE360','INTRO TO DATA MINING QUERY LANGUAGE','1');
8 • D THE DEPARTMENT NAME IN THE COURSE TABLE
9 • urse_dept VALUES('CSE560','ADVANCED DATA MINING QUERY LANGUAGE','1');
10 • REQUISITE COURSE TABLE
11 • q_course_id VALUES('CSE560','CSE360');
12
13 • r_offered,season_offered,capacity,session_id VALUES('CSE360','2020','FALL','150','SESSION1');
14 •
15
16 • ount_password,user_type VALUES('bartsam@buffalo.edu','BART101','USER101','Student');
17 • ENT TABLE
```

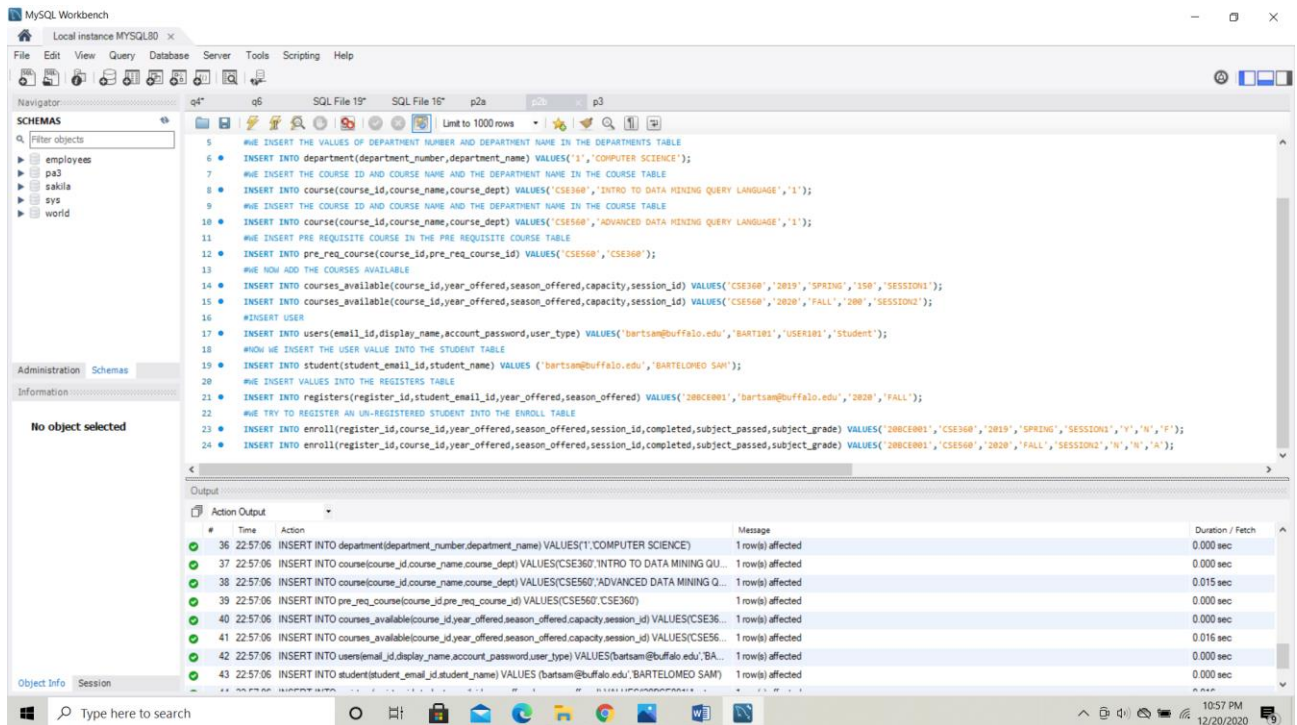
The 'Output' panel shows the execution results:

#	Time	Action	Message	Duration / Fetch
45	03:57:31	INSERT INTO course(course_id,course_name,course_dept) VALUES(CSE360,INTRO TO DATA MINING Q...	1 row(s) affected	0.032 sec
46	03:58:51	INSERT INTO course(course_id,course_name,course_dept) VALUES(CSE560,ADVANCED DATA MINING ...	1 row(s) affected	0.000 sec
47	03:59:13	SELECT * FROM course LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
48	04:05:10	INSERT INTO pre_req_course(course_id,pre_req_course_id) VALUES(CSE560,CSE360)	1 row(s) affected	0.000 sec
49	04:05:26	SELECT * FROM pre_req_course LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
50	04:18:51	INSERT INTO users(email_id,display_name,account_password,user_type) VALUES(bartsam@buffalo.edu,'BA...	1 row(s) affected	0.015 sec
51	04:19:04	SELECT * FROM users LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
52	04:32:56	INSERT INTO registers(register_id,student_email_id,year_offered,season_offered) VALUES(20BCE001,'bartsa...	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL se...	0.000 sec
53	04:35:21	INSERT INTO registers(register_id,student_email_id,year_offered,season_offered) VALUES(20BCE001,'bartsa...	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('grp_proj'.'registers', CONS...	0.031 sec
54	04:40:02	INSERT INTO student(student_email_id,student_name) VALUES (bartsam@buffalo.edu,'BARTELOMEO SA...	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL se...	0.000 sec
55	04:40:15	INSERT INTO student(student_email_id,student_name) VALUES (bartsam@buffalo.edu,'BARTELOMEO SAM)	1 row(s) affected	0.015 sec
56	04:40:25	INSERT INTO registers(register_id,student_email_id,year_offered,season_offered) VALUES(20BCE001,'bartsa...	1 row(s) affected	0.000 sec
57	05:05:38	INSERT INTO courses_available(course_id,year_offered,season_offered,capacity,session_id) VALUES(CSES...	1 row(s) affected	0.063 sec
58	05:06:55	SELECT * FROM courses_available LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
59	05:11:08	INSERT INTO enroll(register_id,course_id) VALUES(20BCE001,CSE560)	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('grp_proj'.'enroll', CONSTRAINT 'register_fk' FOREIGN KEY ('register_id') REFERENCES 'registers' ('register_id'))	

The error message at the bottom of the output panel states: 'Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('grp_proj'.'enroll', CONSTRAINT 'register_fk' FOREIGN KEY ('register_id') REFERENCES 'registers' ('register_id'))'.

WE GET ERROR CODE 1452 WHICH STATES THAT FOREIGN KEY RELATIONSHIP CONSTRAINT FAILS IF WE TRY TO REGISTER AN UNENROLLED STUDENT.

2B) A registered student enrolls CSE 560 in 2020 Fall but does not pass the prerequisite course, CSE 360, of CSE 560.



THE STUDENT IS ABLE TO REGISTER THE COURSE CSE 560 AS LONG AS WE HAVE REGISTERED FOR CSE 360 EVEN IF HE/SHE HASNOT PASSED THE EXAM. INCASE IF YOU ADD USER THAT HAS NOT TAKEN THE PRE REQUISITE COURSE THEN THE RELATIONAL TABLE SHOULD GIVE AN ERROR.

3) DELETE TO REMOVE USER FROM USER TABLE.

The DELETE USER IS NOT SUPPORTED IN OUR RELATIONAL DATABASE SCHEMA AND IT THROWS ERROR AS WE HAVE NOT SPECIFIED THE DELETE SEQUENCE IN THE DESIGN OF THE SCHEMA. BY ADDING (ON DELETE CASCADE) IT HELPS YOU TO DELETE THE INSTANCES OF THE USER FROM ALL THE TABLES WHERE THE ELEMENTS FROM THE PARENT TABLE ARE REFERENCED AS A FOREIGN KEY.

CURRENT ERROR:- CANNOT UPDATE OR DELETE A PARENT ROW AS A FOREIGN KEY CONSTRAINT FAILS.

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' panel with a tree view containing 'employees', 'pa3', 'sakila', 'sys', and 'world'. The main editor window shows a SQL script with the following queries:

```
15 • INSERT INTO courses_available(course_id,year_offered,season_offered,capacity,session_id) VALUES('CSE560','2020','FALL','200','SESSION2');
16 #INSERT USER
17 • INSERT INTO users(email_id,display_name,account_password,user_type) VALUES('bartsam@buffalo.edu','BART101','USER101','Student');
18 #NOW WE INSERT THE USER VALUE INTO THE STUDENT TABLE
19 • INSERT INTO student(student_email_id,student_name) VALUES ('bartsam@buffalo.edu','BARTELOMEO SAM');
20 #WE INSERT VALUES INTO THE REGISTERS TABLE
21 • INSERT INTO registers(register_id,student_email_id,year_offered,season_offered) VALUES('20BCE001','bartsam@buffalo.edu','2020','FALL');
22 #WE TRY TO REGISTER AN UN-REGISTERED STUDENT INTO THE ENROLL TABLE
23 • INSERT INTO enroll(register_id,course_id,year_offered,season_offered,session_id,completed,subject_passed,subject_grade) VALUES('20BCE001','CSE360','2019','SPRING','SESSION1','Y');
24 • INSERT INTO enroll(register_id,course_id,year_offered,season_offered,session_id,completed,subject_passed,subject_grade) VALUES('20BCE001','CSE560','2020','FALL','SESSION2','N');
25 #DELETE THE USER FROM THE USER DATA BASE
26 • SELECT * FROM users;
27 • DELETE FROM users WHERE display_name='BART101';
28
```

The 'Output' panel at the bottom shows the execution results of these queries. The first six queries (lines 15-24) are successful. Query 25 (SELECT) returns 1 row. Query 26 (DELETE) fails with Error Code 1451: 'Cannot delete or update a parent row: a foreign key constraint fails ('grp_proj1'.student, CONSTRAINT 'student_fk' FOREIGN KEY ('student_email_id') REFERENCES 'users' ('email_id'))'. Query 27 (SELECT) returns 1 row. Query 28 (DELETE) also fails with the same Error Code 1451.

#	Time	Action	Message	Duration / Fetch
386	06:03:27	INSERT INTO courses_available(course_id,year_of...	1 row(s) affected	0.000 sec
387	06:03:28	INSERT INTO users(email_id,display_name,accoun...	1 row(s) affected	0.000 sec
388	06:03:30	INSERT INTO student(student_email_id,student_na...	1 row(s) affected	0.000 sec
389	06:03:31	INSERT INTO registers(register_id,student_email_id...	1 row(s) affected	0.015 sec
390	06:03:32	INSERT INTO enroll(register_id,course_id,year_offe...	1 row(s) affected	0.000 sec
391	06:03:36	INSERT INTO enroll(register_id,course_id,year_offe...	Error Code: 1062. Duplicate entry '20BCE001-CSE360-2019-SPRING-SESSION1' for key 'enroll PRIMARY'	0.000 sec
392	06:03:39	INSERT INTO enroll(register_id,course_id,year_offe...	1 row(s) affected	0.000 sec
393	06:04:15	select * from users LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
394	06:06:27	DELETE FROM users WHERE email_id=bartsam@...	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails ('grp_proj1'.student, CONSTRAINT 'student_fk' FOREIGN KEY ('student_email_id') REFERENCES 'users' ('email_id'))	0.016 sec
395	06:08:11	SELECT * FROM users LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
396	06:08:36	DELETE FROM users WHERE(Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails ('grp_proj1'.student, CONSTRAINT 'student_fk' FOREIGN KEY ('student_email_id') REFERENCES 'users' ('email_id'))	