Brandon Lasher
CSIT 565

Homework 3

Task 3: Output for 11-25: (See rating_task3.sql)

```
mysql> source ratings_test.sql
ERROR 1062 (23000): Duplicate entry '102' for key 'Movie.PRIMARY'
ERROR 1062 (23000): Duplicate entry 'Titanic-1997' for key 'Movie.title'
ERROR 1062 (23000): Duplicate entry '201' for key 'Reviewer.PRIMARY'
ERROR 1062 (23000): Duplicate entry '205-104-2011-01-22' for key 'Rating.rID'
ERROR 1048 (23000): Column 'name' cannot be null
ERROR 1048 (23000): Column 'stars' cannot be null
ERROR 3819 (HY000): Check constraint 'year_chk' is violated.
ERROR 3819 (HY000): Check constraint 'star_chk' is violated.
ERROR 3819 (HY000): Check constraint 'date_chk' is violated.
ERROR 3819 (HY000): Check constraint 'ss_chk' is violated.
ERROR 3819 (HY000): Check constraint 'jc_chk' is violated.
Query OK, 1 row affected (0.01 sec)

Query OK, 14 rows affected (0.01 sec)
Rows matched: 14  Changed: 14  Warnings: 0

Query OK, 5 rows affected (0.01 sec)
Rows matched: 5  Changed: 5  Warnings: 0

Query OK, 14 rows affected (0.01 sec)
Rows matched: 14  Changed: 14  Warnings: 0
```

Task 6: Output for 28-31: (See rating_task6.sql)

```
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`csit565`.`Rating`, CONSTRAINT `rat_fk_0` FOREIGN KEY (`rID`) REFERENCES `Review
er` (`rID`) ON DELETE SET NULL ON UPDATE CASCADE)
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`csit565`.`Rating`, CONSTRAINT `rat_fk_0` FOREIGN KEY (`rID`) REFERENCES `Review
er` (`rID`) ON DELETE SET NULL ON UPDATE CASCADE)
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`csit565`.`Rating`, CONSTRAINT `rat_fk_1` FOREIGN KEY (`mID`) REFERENCES `Movie`
 (`mID`) ON DELETE CASCADE)
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`csit565`.`Rating`, CONSTRAINT `rat_fk_1` FOREIGN KEY (`mID`) REFERENCES `Mo
vie` (`mID`) ON DELETE CASCADE)
Query OK, 0 rows affected (0.01 sec)
Rows matched: 0  Changed: 0  Warnings: 0

Query OK, 8 rows affected (0.01 sec)
Rows matched: 8  Changed: 8  Warnings: 0

Query OK, 3 rows affected (0.00 sec)

Query OK, 4 rows affected (0.01 sec)
```

Part2 (Also see rating_part2.sql)

36:

```sql
SELECT MAX(year) FROM Movie WHERE mID IN( SELECT mID FROM Rating WHERE
rID=(SELECT rID FROM Reviewer WHERE name='Chris Jackson') );
```

Output:

```
+-----------+
| MAX(year) |
+-----------+
|      1982 |
+-----------+
1 row in set (0.00 sec)
```

37:

Brandon Lasher
CSIT 565

```sql
SELECT title FROM Movie WHERE mID IN( SELECT mID FROM Rating GROUP by mID HAVING
count(mID) > 2 );
```

Output:


```
+------------------------+
| title                  |
+------------------------+
| Gone with the Wind     |
| Raiders of the Lost Ark |
+------------------------+
2 rows in set (0.00 sec)
```

Part3 (Also see social.sql)
38:

```sql
drop table if exists Student_summary_trigger;

/* Create table and populate with initial dataset */
create table Student_summary_trigger(grade int, numStudents int DEFAULT 0);

INSERT INTO  Student_summary_trigger values ( 9,0);
INSERT INTO  Student_summary_trigger values ( 10,0);
INSERT INTO  Student_summary_trigger values ( 11,0);
INSERT INTO  Student_summary_trigger values ( 12,0);

drop TRIGGER if exists Student_summary_trigger_ai;
DELIMITER //
create TRIGGER Student_summary_trigger_ai
    AFTER INSERT ON Highschooler
    for each row
    UPDATE Student_summary_trigger  SET numStudents = numStudents+1 WHERE grade =
NEW.grade;//
DELIMITER ;

drop TRIGGER if exists Student_summary_trigger_ad;
DELIMITER //
create TRIGGER Student_summary_trigger_ad
    AFTER DELETE ON Highschooler
    for each row
    UPDATE Student_summary_trigger  SET numStudents = numStudents-1 WHERE grade =
OLD.grade;//
DELIMITER ;

drop TRIGGER if exists Student_summary_trigger_au;
```

Brandon Lasher
CSIT 565

```sql
DELIMITER //
create TRIGGER Student_summary_trigger_au
    AFTER UPDATE ON Highschooler
    for each row
    begin
        if NEW.grade != OLD.grade THEN
            UPDATE Student_summary_trigger  SET numStudents = numStudents-1 WHERE
grade = OLD.grade;
            UPDATE Student_summary_trigger  SET numStudents = numStudents+1 WHERE
grade = NEW.grade;
        end if;
    end;//
DELIMITER ;
```

39:

```sql
drop TRIGGER if exists Highschooler_bi;
DELIMITER //
create TRIGGER Highschooler_bi
    BEFORE INSERT ON Highschooler
    for each row
    begin
        if NEW.grade is NULL THEN
            SET NEW.grade = 9;
        ELSEIF NEW.grade < 9 OR NEW.grade > 12 THEN
            SET NEW.grade = NULL;
        end if;
    end;//

DELIMITER ;
```