



**⚠** To see course content, [sign in](#) or [register](#).

# Automatic View Modifications Exercises

You will create (virtual) views over the movie-ratings database that was also used for the SQL Movie-Rating Query Exercises, and you will experiment with how modifications over the views behave when they are handled by the underlying database system automatically. A SQL file to set up the schema and data for the movie-ratings database is downloadable [here](#). This schema and data can be loaded as specified in the file into SQLite, MySQL, or PostgreSQL. However, currently only MySQL supports automatic view modifications. (SQLite and PostgreSQL instead require users to implement triggers or rules for view modifications, as explored in the View Modifications Using Triggers Exercises.) Thus, you must currently use MySQL for these exercises. See our [quick guide](#) for installing and using all three systems.

## Schema:

*Movie* ( *mID*, *title*, *year*, *director* )

English: There is a movie with ID number *mID*, a *title*, a release *year*, and a *director*.

*Reviewer* ( *rID*, *name* )

English: The reviewer with ID number *rID* has a certain *name*.

*Rating* ( *rID*, *mID*, *stars*, *ratingDate* )

English: The reviewer *rID* gave the movie *mID* a number of *stars* rating (1-5) on a certain *ratingDate*.

Each exercise has the same form: It asks you to create a view, and then experiment with modifications over the view, with and without WITH CHECK OPTION. Sample correct answers can be seen by pressing the button at the bottom of the page. Sample answers assume that each exercise is solved using the original data in the movie-rating database.

1. Create a view called *Post80* that contains the title and year of movies made after 1980. Do not use WITH CHECK OPTION in the CREATE VIEW statement.

(a) Is it possible to have a "mishandled insertion," where an insert into view *Post80* is accepted, it causes a modification to the underlying database, but the insertion is not reflected properly in the view afterward? (For this problem and all others in this set, assume insertion commands specify values for all attributes.) If so, give an example.

(b) Is it possible to have a "mishandled deletion," where a delete command on view *Post80* is accepted, it causes a modification to the underlying database, but the deletion is not reflected properly in the view afterward? If so, give an example.

(c) Is it possible to have a "mishandled update," where an update command on view *Post80* is accepted, it causes a modification to the underlying database, but the update is not reflected properly in the view afterward? If so, give an example.

(d) Drop view *Post80* and create it again, this time using WITH CHECK OPTION. For each of (a), (b), and (c) where you gave an example of a mishandled modification, is the modification still permitted?

**2.** Create a view called *Above3* that contains the mID and title of movies that have at least one rating of more than 3 stars. Write the view in a form that makes it modifiable according to MySQL and the SQL standard. Do not use WITH CHECK OPTION in the CREATE VIEW statement.

(a) Is it possible to have a "mishandled insertion" into view *Above3* as described in Exercise 1(a)? If so, give an example.

(b) Is it possible to have a "mishandled deletion" from view *Above3* as described in Exercise 1(b)? If so, give an example.

(c) Is it possible to have a "mishandled update" on view *Above3* as described in Exercise 1(c)? If

so, give an example.

(d) Drop view *Above3* and create it again, this time using WITH CHECK OPTION. For each of (a), (b), and (c) where you gave an example of a mishandled modification, is the modification still permitted?

3. Create a view called *NoDate*(mID,rID,title,name) that contains movie-reviewer pairs where the reviewer didn't give a date for the review. Do not use WITH CHECK OPTION in the CREATE VIEW statement.

(a) Is it possible to have a "mishandled insertion" into view *NoDate*, as described in Exercise 1(a)? If so, give an example.

(b) Is it possible to have a "mishandled deletion" from view *NoDate*, as described in Exercise 1(b)? If so, give an example.

(c) Is it possible to have a "mishandled update" on view *NoDate*, as described in Exercise 1(c)? If so, give an example.

(d) Drop view *NoDate* and create it again, this time using WITH CHECK OPTION. For each of (a), (b), and (c) where you gave an example of a mishandled modification, is the modification still permitted?

4. Create a view called *Liked*(title,name) that contains title-name pairs where the reviewer (name) gave the movie (title) more than 3 stars.

(a) Is it possible to have a "mishandled insertion" into view *Liked*, as described in Exercise 1(a)? If so, give an example.

(b) Is it possible to have a "mishandled deletion" from view *Liked*, as described in Exercise 1(b)? If so, give an example.

(c) Is it possible to have a "mishandled update" on view *Liked*, as described in Exercise 1(c)? If so, give an example.

(d) Drop view *Liked* and create it again, this time using WITH CHECK OPTION. For each of (a), (b), and (c) where you gave an example of a mishandled modification, is the modification still permitted?

**Hide Sample Answers**

**1.**

```
create view Post80 as
  select title, year from Movie where year > 1980
```

*Answers assume the original movie-rating data:*

- (a) Yes - "insert into Post80 values ('Jaws', 1975)"
- (b) No
- (c) Yes - "update Post80 set year = year - 5"
- (d) Mishandled insertions and updates like the ones in (a) and (c) are now disallowed.

**2.**

```
create view Above3 as
  select mID, title from Movie M
  where mID in (select mID from Rating where stars > 3)
```

*Answers assume the original movie-rating data:*

- (a) Yes - "insert into Above3 values (109, 'Rain Man')"
- (b) No
- (c) Yes - "update Above3 set mID = 110 where mID = 101"
- (d) Mishandled insertions and updates like the ones in (a) and (c) are now disallowed.

**3.**

```
create view NoDate as
  select M.mID, R.rID, title, name from Movie M, Reviewer R, Rating T
  where M.mID = T.mID and R.rID = T.rID and ratingDate is null
```

*Answers assume the original movie-rating data:*

- (a) No - system disallows inserts into views with joins
- (b) No - system disallows deletions from views with joins
- (c) Yes - "update NoDate set mID = 110 where mID = 106"

(d) Mishandled updates like the one in (c) are now disallowed.

**4.**

```
create view Liked as
  select title, name from Movie M, Reviewer R, Rating T
  where M.mID = T.mID and R.rID = T.rID and stars > 3
```

*Answers assume the original movie-rating data:*

- (a) No - system disallows inserts into views with joins
- (b) No - system disallows deletions from views with joins
- (c) Yes - "update Liked set title = 'Jaws' where name = 'Daniel Lewis'"
- (d) Mishandled updates like the one in (c) are still allowed.