SQL – Relational Assignment

Before you begin please do the following:

1. Watch the video lecture posted. Many of the commands I will show you on the video.
2. You can also look at the SQLCheatsheet tab for many of the common SQL commands.
3. The below exercises corresponds to the lecture titled Relational Model.

Submission: Save a new Word document which will contain all SQL code and anything else asked for and be submitted in the D2L assignment tab folder titled SQL Primary Keys and More.

I Primary Keys / Foreign Keys / Default

Bring up your previous 2 assignments with the sql code:

1. Write the statements to remove (**drop)** the previous tables: Actor, Castings, and Movie
2. Modify your Create Table statements (from the last two assignments SQL Basics and SQL Continued) to use:
   a. A **primary key** for **each** of the three tables. *Hint: One of them is a composite key!*
   b. A **default** salary in table Castings.
   c. **Foreign keys** where appropriate*: actorId in the Castings table must refer to an actorId in the Actor table and movieId in the Castings table must refer to a movieId from the Movie table.*

   Paste the 3 new create statements.
   Note: When running multiple sql statements you need semicolons (;) separating them. Otherwise run them one at a time.
3. Rerun your insert statements (from the previous 2 weeks' assignments) to insert the rows you did before in the 3 tables. Did they work (yes or no)? If not, explain what happened and fix your errors (paste only the updated inserts below).

4. Write the SQL statement to insert a row into the Movie table where the movieId is NULL. *Note: Specify the word NULL instead of a number.*
5. Was there an error? If so, paste it. Describe what happened.
6. Write the SQL to insert a row into Movie where the movieId is 1.
7. Was there an error? If so, paste it. Describe what happened.
8. Write a statement to insert a new entry into the Castings table, but use a movieId or actorId that is not currently in that appropriate table.
9. What error did you receive? Describe what happened.
10. Write a statement to insert a new entry into the Castings table (with the appropriate movieId, actorId, characterRole), but do not specify the salary.

Note: In the insert specify the columns that you do want to add.

11. Check out what value got entered into the salary, Paste the results of a select * from Castings. Explain what happened.
12. When you run the same command from #10 again what happens? What error message do you receive? Explain.

II Alter

13. Write one statement to alter the table Movie to add a column, rating, which is an integer. (Example: 5 would be a great movie – 5 stars!)
14. Paste the result of Select * from Movie. Check out the rating attribute. What happened?
15. Write a statement to insert a new entry into the Movie table without specifying a rating. You will need to specify the columns you are inserting values into.
16. What happened? What do you see in the rating field for that row in the table?
17. Write the statement to drop the column rating from the Movie table.
18. Paste the result of select * from Movie.
19. Write **one** statement to alter the table Movie to add a column, rating, which is an integer. This time give it a default value of 5.
20. Paste the results of select * from Movie and explain what happened.
21. Write two sql statements to insert new movies the first without specifying a rating and the second with a rating specified (choose a number that's not 5).
22. Paste the results of a Select * from Movie. What happened to the entry you specified a rating for? What happened to the one you did not specify a rating for?