

Eksaminatorier DM534

- Remember to read the relevant slides before attempting to solve the exercises
- I made available the SQL commands to create a database with the Movies relation and all its tuples here: <https://www.db-fiddle.com/f/oEjzriS8QbnNTbpT4yuigf/5>
- You can use the DB fiddle website from above as a starting point to write and test your own SQL commands.

Uge 43

1. Given the following relation schema:

Band(name: CHAR(20), formed_in: INTEGER)

Which of the following are valid tuples of the Band relation?

- ('Foo Fighters', 1994)
 - (1991, 'Incubus')
 - ('Massive Attack')
 - ('Disturbed', '1996')
2. The relation schema of task 1 together with the valid tuples define a relation instance.

Visualize this relation instance as a table.

3. Given the following relation instance:

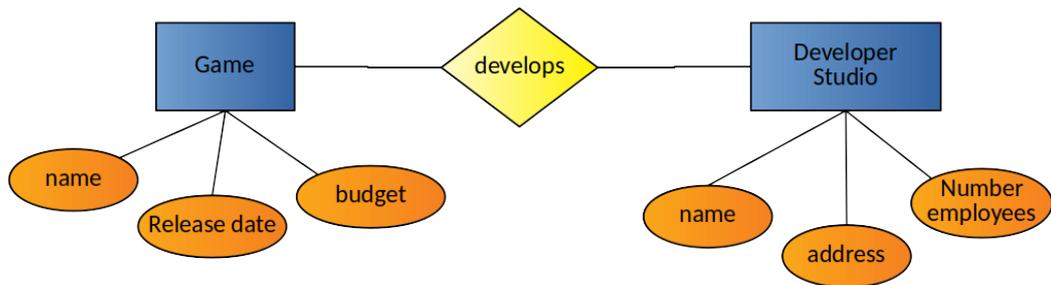
Movies

mid	title	director	production_year	budget_usd
0	'Matrix'	'The Wachowskis'	1999	63000000
1	'Raiders of the Lost Ark'	'Steven Spielberg'	1981	20000000
2	'The Shawshank Redemption'	'Frank Darabont'	1994	25000000
3	'Dead Poets Society'	'Peter Weir'	1989	16400000
4	'Django Unchained'	'Quentin Tarantino'	2012	100000000
5	'Pulp Fiction'	'Quentin Tarantino'	1994	8500000

Which of the following attribute sets are possible primary keys?

- {mid}
- {title}
- {director}
- {title, director}
- {director, production_year}

4. Given the following Entity-Relationship diagram:



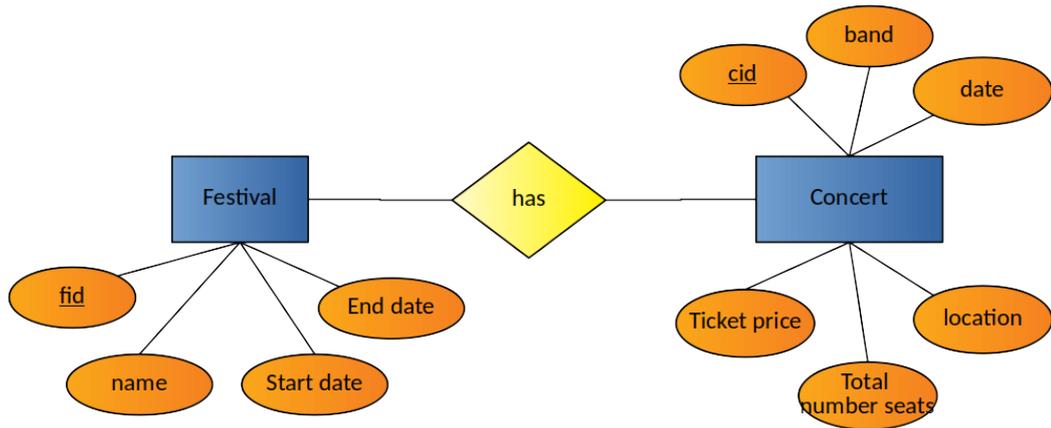
How could this ER diagram be modeled as a relational data model? Provide the relation schemas.

5. Joins are compound operators we did not cover in the lecture. They can join (combine) the data contained in multiple relations together. In relational algebra, the condition join \bowtie_C (also called θ -join) operator is defined as

$$R1 \bowtie_C R2 = \sigma_C(R1 \times R2)$$

In words: A condition join can be calculated by first computing the cross product of the two relations $R1$ and $R2$, followed by a selection using the condition C .

You are given the following ER-diagram:



and the corresponding relations:

- Concert(cid: INTEGER, band: CHAR(20), date: CHAR(20), location: CHAR(20), total_number_seats: INTEGER, ticket_price: FLOAT)
- Festival(fid: INTEGER, name: CHAR(20), start_date: CHAR(20), end_date: CHAR(20))
- FestivalHasConcert(festival: INTEGER, concert: INTEGER)

Here, underlined attributes are primary keys of the relation and dashed underlined attributes are foreign keys.

We now look at the following condition join in particular:

$$Festival \bowtie_{fid=festival} FestivalHasConcert$$

This join combines the two relations Festival and FestivalHasConcert using the attribute fid of the Festival relation, and the festival attribute of the FestivalHasConcert relation. Specifically, it combines those tuples of the two relations for which the equality condition holds.

What is the relation schema of the result relation?

6. You are given the relation instance defined in task 3.

How many tuples does the result relation of the following relational operations contain?

- $\sigma_{true}(Movies)$

- $\pi_{mid,title,director}(Movies)$
 - $\pi_{director,production_year}(Movies)$
 - $\pi_{director}(Movies)$
7. Specify the SQL command that creates the table corresponding to the Concert relation of task 5.
 8. Given the relations of task 5, specify the SQL command that creates a corresponding table for the FestivalHasConcert relation.

Note: FestivalHasConcert's festival and concert attributes are foreign keys to the festival and the concert relation respectively.
 9. Specify an SQL command that deletes all movies from the Movies table, that were not produced in 1994.

Note: You can solve this task with the comparison operators you saw in the lecture. Alternatively, use the "!=" operator that is also defined in SQL.
 10. Specify an SQL command that retrieves all movies from the Movies table of task 3, that were produced in 1994 and directed by Quentin Tarantino.
 11. Formulate the query of task 10 with nested relational operators.
 12. You are given a database with a table that corresponds to the relation instance of task 3. That is, the table has the same name as the relation, 'Movies', and contains the same tuples.

Provide the INSERT and DELETE commands (in total: 2) that need to be executed to end up with a table that corresponds to the following relation instance:

Movies

mid	title	director	production_year	budget_usd
0	'Matrix'	'The Wachowskis'	1999	63000000
1	'Raiders of the Lost Ark'	'Steven Spielberg'	1981	20000000
5	'Pulp Fiction'	'Quentin Tarantino'	1994	8500000
6	'The Lord of the Rings'	'Peter Jackson'	2001	93000000

13. Specify an SQL command without set operations (UNION or EXCEPT) that retrieves all movies from the Movie table, that have been produced before 1990 or that have a budget of at least 30 million USD.
14. Solve task 13 with SQL set operations (UNION or EXCEPT).
15. Specify an SQL command without set operations (UNION or EXCEPT) that retrieves all movies from the Movie table of task 3 that either
 - have been produced before 1990 with a budget of at least 30 million USD

- or have been produced after 2010

16. Specify an SQL command that retrieves all pairs of movies from the Movies table of task 3 that have been directed by the same director.

Hint 1: For instance, if two movies 'movie1' and 'movie2' were directed by the same director, the result relation should, amongst others, contain the following tuples:

- (movie1, movie2)
- (movie2, movie1)

Hint 2: Use the cross product operator in SQL.

17. Formulate the query of task 16 with nested relational operators.

18. Specify an SQL command that retrieves all pairs of movies (movie1, movie2) from the Movies table of task 3, with movie1's budget exceeding the budget of movie2.

Hint 1: For instance, if a movie 'movie1' has a larger budget than 'movie2', the result relation should contain, amongst others, the following tuple:

- (movie1, movie2)

Hint 2: Use the cross product operator in SQL.

19. Specify an SQL command that calculates the condition join given in task 5.

20. Specify an SQL command that calculates the following nested relational operation involving two condition joins:

$$Festival \bowtie_{fid=festival} FestivalHasConcert \bowtie_{concert=cid} Concert$$

Hint 1: This compound operation can be expressed using basic relational operators as

$$\sigma_{fid=festival}(\sigma_{concert=cid}(Festival \times FestivalHasConcert \times Concert))$$

Hint 2: You can start with and extend the SQL command of task 19.

21. Which of the following statements are true (multiple possible)?

- The result of applying a relational algebra operator to a relation instance is another relation instance.
- A relation in a data model is the equivalent concept of a relationship in an ER-diagram.
- Entities of the ER-diagram can not be described by relations in the data model.

- A relation instance needs to contain at least one tuple.
- Integrity constraints are specified when querying the database.
- Primary keys and foreign keys are types of integrity constraints.
- A foreign key can reference arbitrary attributes of other tables.
- A primary key can be used to look up tuples in a table.
- The relational selection operator always returns a relation instance with fewer tuples.
- The relational projection operator may return a relation instance with fewer tuples.
- The SQL UNION operator can be applied to two relation instances, if they have the same number of attributes.