

Exercise Sheet 5: Query Processing

Schema definition

Given is the following schema of a movie-database:

MOVIE(id, name^{not null}, year, type, remark)
PERSON(id, name^{not null}, sex)
PLAYS(movie, person^{not null}, role^{not null})
CREW(movie, person^{not null}, type^{not null}, creditpos)
REMARK(movie, person^{not null}, remark^{not null})
GENRE(movie, genre^{not null})
COUNTRY(movie, country^{not null})

The primary key of the tables MOVIE and PERSON is *id*. In the other tables the primary key is *movie*. Here are some explanations for the different tables:

- **MOVIE**
 - *type*: cinema, game, TV mini series, TV movie, video
 - *remark*: e.g. unreleased
- **PLAYS**
 - All roles with related PERSON.id as person
- **CREW**
 - 3 different person types participating in a movie: actor, director and writer
 - *creditpos*: position of related person in the credits
- **REMARK**
 - Additional Information to a PERSON/MOVIE combination
- **GENRE**
 - e.g. Action, Fantasy, Sci-Fi
- **COUNTRY**
 - Country of origin for each movie

Exercise 1: Relational Algebra

Please translate the following natural language expressions to relational algebra:

- From which countries are the movies of the year 1893 and what are their names?
- Which PC Games were released after the movie 'Terminator'?
- Which actors have played a 'postman', but never participated in a 'Western'?
- Which roles appeared in 'Star Wars: Episode V', as well as in 'Star Wars: Episode VI'?
- How many action cinema movies come from South Africa?
- In which TV Series from the year 1980 did the director also participate as an actor?

Exercise 2: Queries

Please translate the following relational algebra / SQL expressions to natural language.

- $\pi_{pe.id, pe.name}(\sigma_{like(m.name, 'Star Wars\%') \wedge m.type = 'cinema'}(MOVIE\ m) \bowtie_{m.id=p.movie} \sigma_{p.role='Killer'}(PLAYS\ p) \bowtie_{p.person=pe.id} \sigma_{pe.sex='f'}(PERSON\ pe))$
- $\Gamma_{\emptyset \#count(m.id)}(\sigma_{m.year < 1950}(MOVIE\ m) \bowtie_{m.id=p.movie} \sigma_{p.role='Killer'}(PLAYS\ p))$
- SELECT** min(year)
FROM Crew c, Person p, Movie m
WHERE p.id=c.person **AND** m.id=c.movie **AND** p.name='Spielberg, Steven' **AND** c.type='director'
- SELECT** p.name
FROM Person per **JOIN** Plays pl **ON** (per.id = pl.person)
WHERE NOT EXISTS
 (**SELECT** *
 FROM Plays p
 JOIN Movie m **ON** (p.movie = m.id)
 JOIN Person pe **ON** (pe.id = p.person)
 JOIN Genre g **ON** (g.movie = m.id)
 WHERE pe.id = p.id **AND** g.genre = 'Fantasy')
AND pl.role = 'Killer'
AND p.sex = 'f'
ORDER BY name asc

Exercise 3

Prove the correctness of the following expressions or find a counter-example:

- For all relations R and S which have at least one attribute A holds:
 $\pi_A(R \cap S) = \pi_A(R) \cap \pi_A(S)$
- For all relations R, S and appropriate selections φ which only contains attributes from S holds:
 $\sigma_\varphi(R \times S) = \sigma_\varphi(S)$