**중간고사 (데이터베이스시스템, 2021년 2학기)**

201710912 컴퓨터과학과

김지섭

다음과 같은 Movie relational database schema를 이용하여 다음의 SQL을 작성하시오. (100점)

|  |
| --- |
| 테이블이(가) 표시된 사진  자동 생성된 설명 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Studio** | | | | | |
| **Name** | | | **Address** | | **Phone** |
| SM Studio | | | Korea | | 111-1234 |
| Disney Studio | | | USA | | 222-1234 |
| Fox Studio | | | USA | | 333-1234 |
| **Movie** | | | | | | | |
| **Title** | **Year** | | **Length** | **Studio\_Name** | | **Producer\_Num** | |
| Star Wars | 1977 | | 95 | Fox Studio | | P1 | |
| Star Wars | 2010 | | 130 | Fox Studio | | P1 | |
| Star Wars | 2017 | | 120 | Fox Studio | | P2 | |
| Moana | 1990 | | 110 | Disney Studio | | P2 | |
| Lion King | 1990 | | 120 | Disney Studio | | P3 | |
| Lion King | 1995 | | 120 | Disney Studio | | P4 | |
| Frozen | 2013 | | 109 | SM Studio | | P4 | |
| Frozen | 2015 | | 109 | SM Studio | | P1 | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Producer** | | | |
| **Num** | **Name** | **Gender** | **Address** |
| P1 | Lucas | M | USA |
| P2 | Spielberg | M | USA |
| P3 | Lucas | F | USA |
| P4 | Jennifer | F | USA |

|  |  |  |
| --- | --- | --- |
| **Star** | | |
| **Name** | **Age** | **Gender** |
| Robert Downey | 48 | M |
| Chris Evans | 60 | M |
| Brad Pitt | 48 | M |
| Angelina Jolie | 48 | F |
| Elsa | null | F |
| Anna | null | F |
| Jennifer | 45 | F |

|  |  |  |
| --- | --- | --- |
| **StarsIn** | | |
| **Movie\_Title** | **Movie\_Year** | **Star\_Name** |
| Star Wars | 1977 | Robert Downey |
| Star Wars | 1977 | Chris Evans |
| Star Wars | 2010 | Robert Downey |
| Star Wars | 2010 | Chris Evans |
| Star Wars | 2010 | Angelina Jolie |
| Star Wars | 2017 | Elsa |
| Frozen | 2013 | Elsa |
| Frozen | 2015 | Anna |

Movie, StarsIn, Star, Studio, Producer 테이블을 생성하는 SQL문을 작성하시오. 단, primary key, foreign key를 명확히 정의하시오. 각 속성의 도메인은 임의로 설정하시오. 또한, 각각의 테이블에 위의 그림처럼 튜플을 추가하기 위한 SQL문을 모두 작성하시오. (30점)

CREATE DATABASE db\_midterm;

USE db\_midterm;

CREATE TABLE Star (

Name varchar(30),

Age int,

Gender varchar(1),

primary key(Name));

CREATE TABLE Studio (

Name varchar(30),

Address varchar(20),

Phone varchar(13),

primary key(Name));

CREATE TABLE Producer (

Num varchar(5),

Name varchar(30),

Gender varchar(1),

Address varchar(20),

primary key(Num));

CREATE TABLE Movie (

Title varchar(20),

Year int,

Length int,

Studio\_Name varchar(30),

Producer\_Num varchar(5),

primary key(Title,Year),

foreign key(Studio\_Name) references Studio(Name),

foreign key(Producer\_Num) references Producer(Num));

CREATE TABLE Starsln (

Movie\_Title varchar(20),

Movie\_Year int,

Star\_Name varchar(30),

primary key(Movie\_Title,Movie\_Year, Star\_Name),

foreign key(Movie\_Title,Movie\_Year) references Movie(Title,Year),

foreign key(Star\_Name) references Star(Name));

insert into star values('Robert Downey', 48, "M");

insert into star values('Chris Evans', 60, "M");

insert into star values('Brad Pitt', 48, "M");

insert into star values('Angenlina Jolie', 48, "F");

insert into star values('Elsa', null, "F");

insert into star values('Anna', null, "F");

insert into star values('Jennifer', 45, "F");

insert into Studio values('SM Studio', 'Korea', '111-1234');

insert into Studio values('Disney Studio', 'USA', '222-1234');

insert into Studio values('Fox Studio', 'USA', '333-1234');

insert into Producer values('P1', 'Lucas', 'M', 'USA');

insert into Producer values('P2', 'Spielberg', 'M', 'USA');

insert into Producer values('P3', 'Lucas', 'F', 'USA');

insert into Producer values('P4', 'Jennifer', 'F', 'USA');

insert into Movie values('Star Wars', 1977, 95, 'Fox Studio', 'P1');

insert into Movie values('Star Wars', 2010, 130, 'Fox Studio', 'P1');

insert into Movie values('Star Wars', 2017, 120, 'Fox Studio', 'P2');

insert into Movie values('Moana', 1990, 110, 'Disney Studio', 'P2');

insert into Movie values('Lion King', 1990, 120, 'Disney Studio', 'P3');

insert into Movie values('Lion King', 1995, 120, 'Disney Studio', 'P4');

insert into Movie values('Frozen', 2013, 109, 'SM Studio', 'P4');

insert into Movie values('Frozen', 2015, 109, 'SM Studio', 'P2');

insert into Starsln values('Star Wars', 1977, 'Robert Downey');

insert into Starsln values('Star Wars', 1977, 'Chris Evans');

insert into Starsln values('Star Wars', 2010, 'Robert Downey');

insert into Starsln values('Star Wars', 2010, 'Chris Evans');

insert into Starsln values('Star Wars', 2010, 'Angenlina Jolie');

insert into Starsln values('Star Wars', 2017, 'Elsa');

insert into Starsln values('Frozen', 2013, 'Elsa');

insert into Starsln values('Frozen', 2015, 'Anna');

Find the title of all movies of which length is greater than 115 minutes (10 점)

Select Title

from Movie

where length > 115;

Find the names and addresses of the producers of the movies that were produced by “Disney Studio” in 1990 (10점)

select Name, Address

from Producer join Movie on Producer.Num = Movie.Producer\_Num

where Studio\_Name = 'Disney studio' and Year = 1990;

Find the names of all movie stars whose age is unknown (10점)

Select Name

from star

where age is null;

Find the (distinct) names of all movie stars that appeared in a movie with “Star Wars”. Note that the results should be ordered by age (oldest first) and among stars of same age, the alphabetic order of names (10점)

Select Name

from (Select Distinct Name, age

from star join starsln on star.Name = Starsln.Star\_Name

where Movie\_Title = 'Star Wars'

order by age desc, Name) subtable;

For each studio, retrieve the studio name and the total number of movie produced by the studio (반드시 Group By 사용) (10점)

select Name, count(\*)

from Studio join Movie on Studio.Name = Movie.Studio\_Name

group by Name;

For each studio, retrieve the studio name, studio address, studio phone number, and the total number of movie produced by the studio

(반드시 Correlated Scalar Subquery를 사용) (10점)

select Name, Address, Phone, (select count(\*)

from Movie

where Studio.Name = Movie.Studio\_Name)

as counter

from Studio;

For all producers who made at least two movies, display their Name and the total length of movies which they produced (반드시 Having을 사용, Derived Relation 사용 금지) (10점)

select Name, sum(length)

from Studio join Movie on Movie.Studio\_Name = Studio.Name

group by Name

having count(\*) >= 2;

For all producers who made at least two movies, display their Name and the total length of movies which they produced (반드시 Derived Relation을 사용, Having 사용 금지) (10점)

select Name, sum\_length

from (select Name, sum(length) as sum\_length, count(\*) as counter

from Studio join Movie on Movie.Studio\_Name = Studio.Name

group by Name) count\_sum

where counter >= 2;

전체 내용 입니다.

CREATE DATABASE db\_midterm;

USE db\_midterm;

CREATE TABLE Star (

Name varchar(30),

Age int,

Gender varchar(1),

primary key(Name));

CREATE TABLE Studio (

Name varchar(30),

Address varchar(20),

Phone varchar(13),

primary key(Name));

CREATE TABLE Producer (

Num varchar(5),

Name varchar(30),

Gender varchar(1),

Address varchar(20),

primary key(Num));

CREATE TABLE Movie (

Title varchar(20),

Year int,

Length int,

Studio\_Name varchar(30),

Producer\_Num varchar(5),

primary key(Title,Year),

foreign key(Studio\_Name) references Studio(Name),

foreign key(Producer\_Num) references Producer(Num));

CREATE TABLE Starsln (

Movie\_Title varchar(20),

Movie\_Year int,

Star\_Name varchar(30),

primary key(Movie\_Title,Movie\_Year, Star\_Name),

foreign key(Movie\_Title,Movie\_Year) references Movie(Title,Year),

foreign key(Star\_Name) references Star(Name));

insert into star values('Robert Downey', 48, "M");

insert into star values('Chris Evans', 60, "M");

insert into star values('Brad Pitt', 48, "M");

insert into star values('Angenlina Jolie', 48, "F");

insert into star values('Elsa', null, "F");

insert into star values('Anna', null, "F");

insert into star values('Jennifer', 45, "F");

insert into Studio values('SM Studio', 'Korea', '111-1234');

insert into Studio values('Disney Studio', 'USA', '222-1234');

insert into Studio values('Fox Studio', 'USA', '333-1234');

insert into Producer values('P1', 'Lucas', 'M', 'USA');

insert into Producer values('P2', 'Spielberg', 'M', 'USA');

insert into Producer values('P3', 'Lucas', 'F', 'USA');

insert into Producer values('P4', 'Jennifer', 'F', 'USA');

insert into Movie values('Star Wars', 1977, 95, 'Fox Studio', 'P1');

insert into Movie values('Star Wars', 2010, 130, 'Fox Studio', 'P1');

insert into Movie values('Star Wars', 2017, 120, 'Fox Studio', 'P2');

insert into Movie values('Moana', 1990, 110, 'Disney Studio', 'P2');

insert into Movie values('Lion King', 1990, 120, 'Disney Studio', 'P3');

insert into Movie values('Lion King', 1995, 120, 'Disney Studio', 'P4');

insert into Movie values('Frozen', 2013, 109, 'SM Studio', 'P4');

insert into Movie values('Frozen', 2015, 109, 'SM Studio', 'P2');

insert into Starsln values('Star Wars', 1977, 'Robert Downey');

insert into Starsln values('Star Wars', 1977, 'Chris Evans');

insert into Starsln values('Star Wars', 2010, 'Robert Downey');

insert into Starsln values('Star Wars', 2010, 'Chris Evans');

insert into Starsln values('Star Wars', 2010, 'Angenlina Jolie');

insert into Starsln values('Star Wars', 2017, 'Elsa');

insert into Starsln values('Frozen', 2013, 'Elsa');

insert into Starsln values('Frozen', 2015, 'Anna');

Select Title from Movie where length > 115;

select Name, Address

from Producer join Movie on Producer.Num = Movie.Producer\_Num

where Studio\_Name = 'Disney studio' and Year = 1990;

Select Name

from star

where age is null;

Select Name

from (Select Distinct Name, age

from star join starsln on star.Name = Starsln.Star\_Name

where Movie\_Title = 'Star Wars'

order by age desc, Name) subtable;

select Name, count(\*)

from Studio join Movie on Studio.Name = Movie.Studio\_Name

group by Name;

select Name, Address, Phone, (select count(\*)

from Movie

where Studio.Name = Movie.Studio\_Name)

as counter

from Studio;

select Name, sum(length)

from Studio join Movie on Movie.Studio\_Name = Studio.Name

group by Name

having count(\*) >= 2;

select Name, sum\_length

from (select Name, sum(length) as sum\_length, count(\*) as counter

from Studio join Movie on Movie.Studio\_Name = Studio.Name

group by Name) count\_sum

where counter >= 2;