12/08/2018

**675-01/775-01 (Friday section)**

Course project phase 2 – SQL statements

Prepared by:

Amelia Rawlings (amelia.rawlings@gmail.com),

Benjamin Mayer (bmayer@mail.sfsu.edu),

Ilya Kopyl (ikopyl@mail.sfsu.edu)

CREATE TABLE FilmProductionStudio (  
 StudioID INTEGER,  
 Name VARCHAR(255) NOT NULL,  
 PRIMARY KEY (StudioID));

CREATE TABLE FilmIndustryWorker (  
 SSN CHAR(11),  
 FirstName VARCHAR(255) NOT NULL,  
 LastName VARCHAR(255) NOT NULL,  
 Gender VARCHAR(10),  
 MemberOfScreenActorsGuild CHAR(1) NOT NULL,  
 PRIMARY KEY (SSN));  
  
CREATE TABLE HasContract (  
 ContractID INTEGER,  
 StudioID INTEGER NOT NULL,  
 SSN CHAR(11) NOT NULL,  
 Compensation DOUBLE NOT NULL,  
 DateSigned DATETIME NOT NULL,  
 PRIMARY KEY (ContractID),  
 FOREIGN KEY (StudioID) REFERENCES FilmProductionStudio,  
 FOREIGN KEY (SSN) REFERENCES FilmIndustryWorker  
 ON DELETE SET NULL);  
  
CREATE TABLE Movie (  
 MovieID INTEGER,  
 StudioID INTEGER NOT NULL,  
 DirectorSSN CHAR(11) NOT NULL,  
 Title VARCHAR(255) NOT NULL,  
 Budget DOUBLE,  
 PRIMARY KEY (MovieID),  
 FOREIGN KEY (StudioID) REFERENCES FilmProductionStudio,  
 FOREIGN KEY (DirectorSSN) REFERENCES FilmIndustryWorker  
 ON DELETE SET NULL);  
  
CREATE TABLE Actor (  
 SSN CHAR(11) NOT NULL,  
 MovieID INTEGER NOT NULL,  
 PRIMARY KEY (SSN, MovieID),  
 FOREIGN KEY (SSN) REFERENCES FilmIndustryWorker,  
 FOREIGN KEY (MovieID) REFERENCES Movie  
 ON DELETE CASCADE);  
  
CREATE TABLE Location (  
 LocationID INTEGER,  
 Name VARCHAR(255) NOT NULL,  
 City VARCHAR(255),  
 State VARCHAR(255),  
 Country VARCHAR(255),  
 PRIMARY KEY (LocationID));

CREATE TABLE ShootingSchedule (  
 ScheduleID INTEGER,  
 MovieID INTEGER NOT NULL,  
 LocationID INTEGER NOT NULL,  
 Date DATETIME,  
 PRIMARY KEY (ScheduleID),  
 FOREIGN KEY (MovieID) REFERENCES Movie,  
 FOREIGN KEY (LocationID) REFERENCES Location  
 ON DELETE CASCADE);

/\*==================================================================================\*/  
/\* 01 - inserting FilmProductionStudio records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO FilmProductionStudio(StudioID, Name) VALUES(10001, 'Columbia Pictures');  
INSERT INTO FilmProductionStudio(StudioID, Name) VALUES(10002, 'Miramax Films');  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10003, "The Walt Disney Company");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10004, "Warner Bros.");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10005, "Paramount Pictures");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10006, "Universal Pictures");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10007, "20th Century Fox");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10008, "Lionsgate");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10009, "DreamWork Studios");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10010, "Metro-Goldwyn-Mayer");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10011, "Sony Pictures");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10012, "LucasFilm");  
INSERT INTO FilmProductionStudio(StudioID, NAME) VALUES(10013, "Marvel Studios");

/\*==================================================================================\*/

Inserted data:

|  |  |
| --- | --- |
| **StudioID** | **Name** |
| 10001 | Columbia Pictures |
| 10002 | Miramax Films |
| 10003 | The Walt Disney Company |
| 10004 | Warner Bros. |
| 10005 | Paramount Pictures |
| 10006 | Universal Pictures |
| 10007 | 20th Century Fox |
| 10008 | Lionsgate |
| 10009 | DreamWork Studios |
| 10010 | Metro-Goldwyn-Mayer |
| 10011 | Sony Pictures |
| 10012 | LucasFilm |
| 10013 | Marvel Studios |

\*==================================================================================\*/  
/\* 02 - inserting FilmIndustryWorker records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1001', 'Stanley', 'Kubrick', 'Male', 0);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1002', 'Quentin', 'Tarantino', 'Male', 1);  
  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1003', 'Peter', 'Sellers', 'Male', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1004', 'Sterling', 'Hayden', 'Male', 1);

INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1005', 'John', 'Travolta', 'Male', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1006', 'Uma', 'Thurman', 'Female', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1007', 'Bruce', 'Willis', 'Male', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1008', 'Samuel L', 'Jackson', 'Male', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1009', 'Amanda', 'Plummer', 'Female', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1010', 'Harvey', 'Keitel', 'Male', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1011', 'Tim', 'Roth', 'Male', 1);  
INSERT INTO FilmIndustryWorker(SSN, FirstName, LastName, Gender, MemberOfScreenActorsGuild) VALUES('123-45-1012', 'Christopher', 'Walken', 'Male', 1);  
  
/\*==================================================================================\*/

Inserted data:



/\*==================================================================================\*/  
/\* 03 - inserting HasContract records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20001, 10001, '123-45-1001', 3000000.00, '1962-07-31');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20002, 10002, '123-45-1002', 1000000.00, '1993-05-20');  
  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20003, 10001, '123-45-1003', 100000.00, '1962-08-15');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20004, 10001, '123-45-1004', 100000.00, '1962-08-15');  
  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20005, 10002, '123-45-1005', 250000.00, '1993-06-15');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20006, 10002, '123-45-1006', 250000.00, '1993-06-15');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20007, 10002, '123-45-1007', 250000.00, '1993-06-15');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20008, 10002, '123-45-1008', 250000.00, '1993-06-15');  
  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20009, 10002, '123-45-1009', 150000.00, '1993-06-17');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20010, 10002, '123-45-1010', 150000.00, '1993-06-17');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20011, 10002, '123-45-1011', 50000.00, '1993-06-17');  
INSERT INTO HasContract(ContractID, StudioID, SSN, Compensation, DateSigned) VALUES(20012, 10002, '123-45-1012', 50000.00, '1993-06-17');  
  
/\*==================================================================================\*/

Inserted data:



/\*==================================================================================\*/  
/\* 04 - inserting Movie records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO Movie(MovieID, StudioID, DirectorSSN, Title, Budget) VALUES(30001, 10001, '123-45-1001', 'Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb', 1800000.00);  
INSERT INTO Movie(MovieID, StudioID, DirectorSSN, Title, Budget) VALUES(30002, 10002, '123-45-1002', 'Pulp Fiction', 8500000.00);  
  
INSERT INTO Movie(MovieID, StudioID, DirectorSSN, Title, Budget) VALUES(30003, 10002, '123-45-1002', 'Reservoir Dogs', 1500000.00);  
INSERT INTO Movie(MovieID, StudioID, DirectorSSN, Title, Budget) VALUES(30004, 10001, '123-45-1001', '2001 - A Space Odyssey', 10500000.00);  
  
/\*==================================================================================\*/

Inserted data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MovieID** | **StudioID** | **DirectorSSN** | **Title** | **Budget** |
| 30001 | 10001 | 123-45-1001 | Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb | 1800000 |
| 30002 | 10002 | 123-45-1002 | Pulp Fiction | 8500000 |
| 30003 | 10002 | 123-45-1002 | Reservoir Dogs | 1500000 |
| 30004 | 10001 | 123-45-1001 | 2001 - A Space Odyssey | 10500000 |

/\*==================================================================================\*/  
/\* 05 - inserting Actor records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1002', 30002);  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1005', 30002);  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1006', 30002);  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1007', 30002);  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1008', 30002);  
  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1003', 30001);  
INSERT INTO Actor(SSN, MovieID) VALUES('123-45-1004', 30001);  
  
/\*==================================================================================\*/

Inserted data:



/\*==================================================================================\*/  
/\* 06 - inserting Location records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO Location(LocationID, Name, City, State, Country) VALUES(60001, 'Shepperton Studios', 'Shepperton', 'Surrey', 'UK');  
INSERT INTO Location(LocationID, Name, City, State, Country) VALUES(60002, 'Culver City Warehouse', 'Culver City', 'California', 'USA');  
INSERT INTO Location(LocationID, Name, City, State, Country) VALUES(60003, 'Hawthorne Grill', 'Hawthorne', 'California', 'USA');  
  
/\*==================================================================================\*/

Inserted data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LocationID** | **Name** | **City** | **State** | **Country** |
| 60001 | Shepperton Studios | Shepperton | Surrey | UK |
| 60002 | Culver City Warehouse | Culver City | California | USA |
| 60003 | Hawthorne Grill | Hawthorne | California | USA |

/\*==================================================================================\*/  
/\* 07 - inserting ShootingSchedule records here \*/  
/\*==================================================================================\*/  
  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70001, 30001, 60001, '1962/10/13');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70002, 30001, 60001, '1962/10/14');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70003, 30001, 60001, '1962/10/15');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70004, 30001, 60001, '1962/10/16');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70005, 30001, 60001, '1962/10/17');  
  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70006, 30002, 60002, '1993/09/20');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70007, 30002, 60002, '1993/09/21');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70008, 30002, 60002, '1993/09/22');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70009, 30002, 60002, '1993/09/23');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70010, 30002, 60002, '1993/09/24');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70011, 30002, 60002, '1993/09/25');  
  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70012, 30002, 60003, '1993/11/01');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70013, 30002, 60003, '1993/11/02');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70014, 30002, 60003, '1993/11/03');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70015, 30002, 60003, '1993/11/04');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70016, 30002, 60003, '1993/11/05');  
INSERT INTO ShootingSchedule(ScheduleID, MovieID, LocationID, Date) VALUES(70017, 30002, 60003, '1993/11/06');  
  
/\*==================================================================================\*/

Inserted data:

|  |  |  |  |
| --- | --- | --- | --- |
| **ScheduleID** | **MovieID** | **LocationID** | **Date** |
| 70001 | 30001 | 60001 | 10/13/62 |
| 70002 | 30001 | 60001 | 10/14/62 |
| 70003 | 30001 | 60001 | 10/15/62 |
| 70004 | 30001 | 60001 | 10/16/62 |
| 70005 | 30001 | 60001 | 10/17/62 |
| 70006 | 30002 | 60002 | 9/20/93 |
| 70007 | 30002 | 60002 | 9/21/93 |
| 70008 | 30002 | 60002 | 9/22/93 |
| 70009 | 30002 | 60002 | 9/23/93 |
| 70010 | 30002 | 60002 | 9/24/93 |
| 70011 | 30002 | 60002 | 9/25/93 |
| 70012 | 30002 | 60003 | 11/1/93 |
| 70013 | 30002 | 60003 | 11/2/93 |
| 70014 | 30002 | 60003 | 11/3/93 |
| 70015 | 30002 | 60003 | 11/4/93 |
| 70016 | 30002 | 60003 | 11/5/93 |
| 70017 | 30002 | 60003 | 11/6/93 |

/\* query01 - Find the names of all movies that were shot in more than one location \*/  
SELECT M.Title  
FROM Movie M  
WHERE M.MovieID IN (  
 SELECT S.MovieID  
 FROM ShootingSchedule S  
 GROUP BY S.MovieID  
 HAVING COUNT(DISTINCT LocationID) >= 2);

Query result:

|  |
| --- |
| **Title** |
| Pulp Fiction |

/\* creating INDEX 1, in Oracle DB format: \*/

CREATE INDEX IndexMovieLocation   
ON ShootingSchedule  
WITH STRUCTURE = BTREE  
INDEX = (MovieID, LocationID);

/\* creating INDEX 1, in SQLite 3 format, equivalent to statement above: \*/  
CREATE INDEX IndexMovieLocation  
ON ShootingSchedule (MovieID, LocationID);  
  
/\* INDEX 1 TEST \*/  
EXPLAIN QUERY PLAN SELECT M.Title  
FROM Movie M  
WHERE M.MovieID IN (  
 SELECT S.MovieID  
 FROM ShootingSchedule S  
 GROUP BY S.MovieID  
 HAVING COUNT(DISTINCT LocationID) >= 2);

Query result:



/\* query02 - Find for how many days each movie was shot at each location \*/  
SELECT M.Title AS 'Movie Title', L.Name AS 'Shooting Location', COUNT(DISTINCT S.Date) AS 'Days Shot'  
FROM Movie M, Location L, ShootingSchedule S  
WHERE M.MovieID = S.MovieID  
 AND L.LocationID = S.LocationID  
GROUP BY M.Title, L.Name;

Query result:

|  |  |  |
| --- | --- | --- |
| **Movie Title** | **Shooting Location** | **Days Shot** |
| Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb | Shepperton Studios | 5 |
| Pulp Fiction | Culver City Warehouse | 6 |
| Pulp Fiction | Hawthorne Grill | 6 |

/\* query03 - Find the names of all Film Production Studios that did not produce any movies (according to the database) \*/  
SELECT FPS.Name AS 'Name of Production Studio'  
FROM FilmProductionStudio FPS  
WHERE FPS.StudioID NOT IN (  
 SELECT M.StudioID  
 FROM Movie M);

Query result:

|  |
| --- |
| **Name of Production Studio** |
| The Walt Disney Company |
| Warner Bros. |
| Paramount Pictures |
| Universal Pictures |
| 20th Century Fox |
| Lionsgate |
| DreamWork Studios |
| Metro-Goldwyn-Mayer |
| Sony Pictures |
| LucasFilm |
| Marvel Studios |

/\* query04 - Find Director's name and name of the movie with budget > $2 Million \*/  
SELECT M.Title, FIW.FirstName, FIW.LastName  
FROM Movie M, FilmIndustryWorker FIW  
WHERE M.DirectorSSN = FIW.SSN  
 AND M.budget > 2000000;

Query result:

|  |  |  |
| --- | --- | --- |
| **Title** | **FirstName** | **LastName** |
| Pulp Fiction | Quentin | Tarantino |
| 2001 - A Space Odyssey | Stanley | Kubrick |

/\* query05 - Find a name of the Film Industry worker with highest compensation \*/  
SELECT FIW.FirstName, FIW.LastName  
FROM FilmIndustryWorker FIW, HasContract HC  
WHERE FIW.SSN = HC.SSN  
 AND HC.Compensation = (SELECT MAX(HC2.Compensation)  
 FROM HasContract HC2);

Query result:

|  |  |
| --- | --- |
| **FirstName** | **LastName** |
| Stanley | Kubrick |

/\* Equivalent statement to the above one \*/  
SELECT FIW.FirstName, FIW.LastName  
FROM FilmIndustryWorker FIW, HasContract HC  
WHERE FIW.SSN = HC.SSN  
 AND HC.Compensation >= ALL (SELECT HC2.Compensation  
 FROM HasContract HC2);

Query result:

|  |  |
| --- | --- |
| **FirstName** | **LastName** |
| Stanley | Kubrick |

/\* query06 - Calculate a compensation of all actors per each movie \*/  
SELECT M.Title, SUM(H.Compensation) AS 'Total Actors Compensation'  
FROM HasContract H, Actor A, Movie M  
WHERE H.SSN = A.SSN  
 AND A.MovieID = M.MovieID  
GROUP BY M.Title;

Query result:

|  |  |
| --- | --- |
| **Title** | **Total Actors Compensation** |
| Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb | 200000 |
| Pulp Fiction | 2000000 |

/\* creating INDEX 2 in Oracle DB format: \*/

CREATE INDEX WorkersCompensation  
On HasContract

WITH STRUCTURE = BTREE  
INDEX = (MovieID, LocationID);

/\* creating INDEX 2 in SQLite 3 format, equivalent to the statement above: \*/  
CREATE INDEX WorkersCompensation  
On HasContract (SSN, Compensation);  
  
/\* INDEX 2 TEST \*/  
EXPLAIN QUERY PLAN SELECT M.Title, SUM(H.Compensation) AS 'Total Actors Compensation'  
FROM HasContract H, Actor A, Movie M  
WHERE H.SSN = A.SSN  
 AND A.MovieID = M.MovieID  
GROUP BY M.Title;

Query result:

|  |  |  |  |
| --- | --- | --- | --- |
| **selectid** | **order** | **from** | **detail** |
| 0 | 0 | 1 | SCAN TABLE Actor AS A |
| 0 | 1 | 2 | SEARCH TABLE Movie AS M USING INTEGER PRIMARY KEY (rowid=?) |
| 0 | 2 | 0 | SEARCH TABLE HasContract AS H USING COVERING INDEX WorkersCompensation (SSN=?) |
| 0 | 0 | 0 | USE TEMP B-TREE FOR GROUP BY |

/\* query07 - Find names of all actors (who are not directors) with the largest compensation \*/  
SELECT FIW.FirstName, FIW.LastName, HC.Compensation  
FROM FilmIndustryWorker FIW, HasContract HC  
WHERE FIW.SSN = HC.SSN  
 AND HC.Compensation = (SELECT MAX(HC2.Compensation)   
 FROM HasContract HC2  
 WHERE HC2.SSN NOT IN (SELECT M.DirectorSSN FROM Movie M)  
 AND HC2.SSN IN (SELECT A.SSN FROM Actor A));

Query result:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Compensation** |
| John | Travolta | 250000 |
| Uma | Thurman | 250000 |
| Bruce | Willis | 250000 |
| Samuel L | Jackson | 250000 |

/\* semantically equivalent to this SQL query: \*/  
SELECT FIW.FirstName, FIW.LastName, HC.Compensation  
FROM FilmIndustryWorker FIW, HasContract HC  
WHERE FIW.SSN = HC.SSN  
 AND HC.Compensation >= ALL (SELECT HC2.Compensation  
 FROM HasContract HC2  
 WHERE HC2.SSN NOT IN (SELECT M.DirectorSSN FROM Movie M)  
 AND HC2.SSN IN (SELECT A.SSN FROM Actor A));

Query result:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Compensation** |
| John | Travolta | 250000 |
| Uma | Thurman | 250000 |
| Bruce | Willis | 250000 |
| Samuel L | Jackson | 250000 |

/\* query08 - Find all directors (who are not actors) with the biggest compensation \*/  
SELECT FIW.FirstName, FIW.LastName, HC.Compensation  
FROM FilmIndustryWorker FIW, HasContract HC  
WHERE FIW.SSN = HC.SSN  
 AND HC.Compensation = (SELECT MAX(HC2.Compensation)   
 FROM HasContract HC2  
 WHERE HC2.SSN IN (SELECT M.DirectorSSN FROM Movie M)  
 AND HC2.SSN NOT IN (SELECT A.SSN FROM Actor A));

Query result:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Compensation** |
| Stanley | Kubrick | 3000000 |

/\* Semantically equivalent SQL query, with ALL: \*/  
SELECT FIW.FirstName, FIW.LastName, HC.Compensation  
FROM FilmIndustryWorker FIW, HasContract HC  
WHERE FIW.SSN = HC.SSN  
 AND HC.Compensation >= ALL (SELECT HC2.Compensation  
 FROM HasContract HC2  
 WHERE HC2.SSN IN (SELECT M.DirectorSSN FROM Movie M)  
 AND HC2.SSN NOT IN (SELECT A.SSN FROM Actor A));

Query result:

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Compensation** |
| Stanley | Kubrick | 3000000 |