Group 3:

Tejesh Rathi

Palak Agrawal

Sonali Chaudhari

Abstract

All the queries with its result set for Oracle Database

IMDB

Internet Movie Database

-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 1

Query : Fetch Actor and Actress for each Movie not 'IN-THEATRE', also print Director for them AND order them by MOVIE alphabetically\*/

-----------------------------------------------------------------------------------------

SELECT \* FROM (SELECT CFM1.MOVIENAME AS MOVIE,

CFM1.CELEBRITYNAME AS DIRECTOR,

CFM2.CELEBRITYNAME AS ACTORORACTRESS,

SR.Status

FROM (SELECT DISTINCT(CELEBRITYNAME), MOVIENAME, MOVIEID FROM v\_CASTFORMOVIE WHERE TYPEID = 4 ) CFM1

LEFT JOIN (SELECT DISTINCT(CELEBRITYNAME), MOVIENAME, MOVIEID FROM v\_CASTFORMOVIE WHERE TYPEID IN (1,2)) CFM2

ON CFM2.MOVIEID = CFM1.MOVIEID

LEFT JOIN MOVIE M

ON M.MovieID = CFM2.MOVIEID

LEFT JOIN MOVIETVStatusRef SR

ON SR.StatusID = M.StatusID

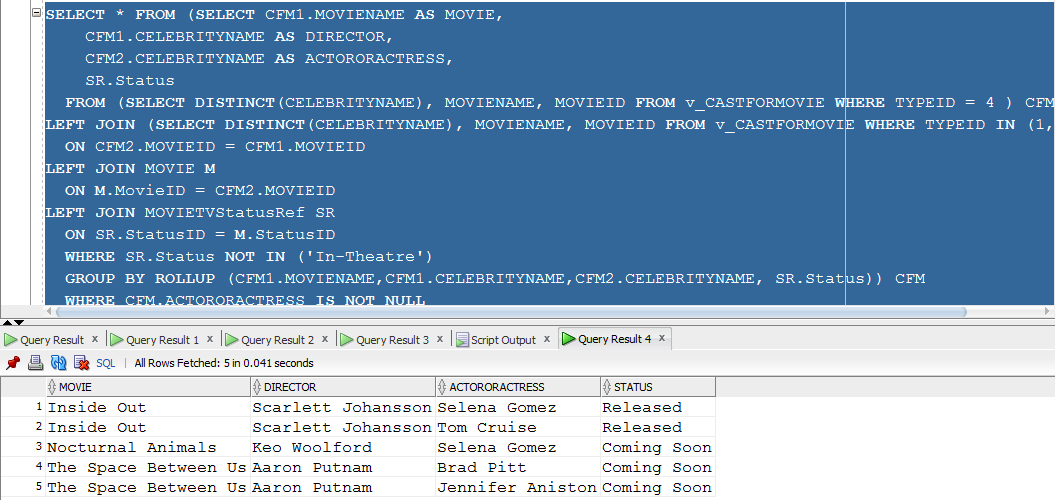
WHERE SR.Status NOT IN ('In-Theatre')

GROUP BY ROLLUP (CFM1.MOVIENAME,CFM1.CELEBRITYNAME,CFM2.CELEBRITYNAME, SR.Status)) CFM

WHERE CFM.ACTORORACTRESS IS NOT NULL

AND CFM.Status IS NOT NULL

ORDER BY CFM.MOVIE;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 2

Query : Show the total collection till today for Movie bookings as per the Showtype for each Day and each Theatre\*/

-----------------------------------------------------------------------------------------

SELECT \*

FROM (SELECT VS.SHOWNAME,

UB.ShowDate,

T.Name AS THEATRENAME,

UB.TotalPrice

FROM v\_SHOWS VS

INNER JOIN MOVIEShowInTheatre MST

ON MST.SHOWTIMEID = VS.ShowTimeID

INNER JOIN USERBOOKING UB

ON UB.ShowTimeID = MST.ShowTimeID

INNER JOIN THEATRE T

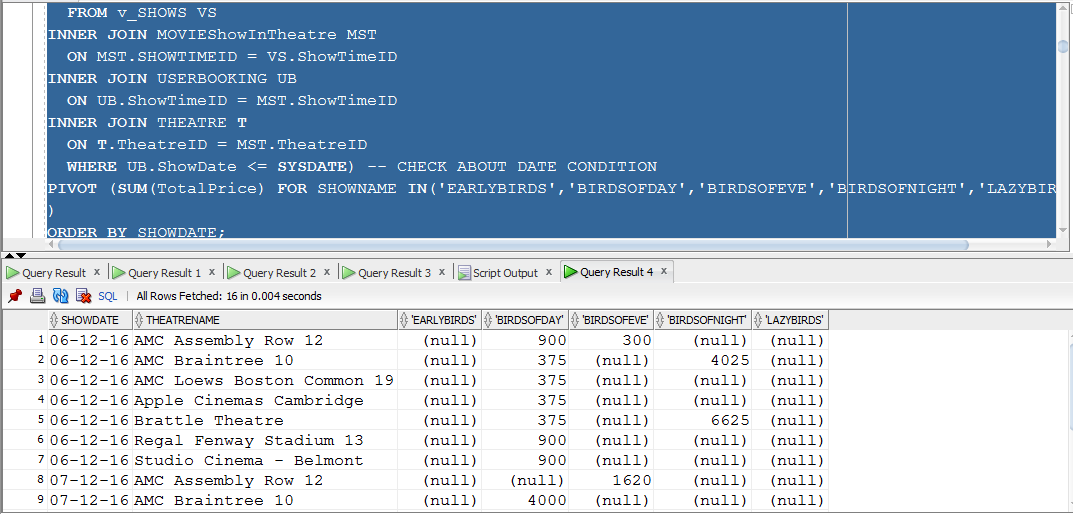
ON T.TheatreID = MST.TheatreID

WHERE UB.ShowDate <= SYSDATE) -- CHECK ABOUT DATE CONDITION

PIVOT (SUM(TotalPrice) FOR SHOWNAME IN('EARLYBIRDS','BIRDSOFDAY','BIRDSOFEVE','BIRDSOFNIGHT','LAZYBIRDS')

)

ORDER BY SHOWDATE;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 3

Query : Retrieve highest rated movie by Genre and If any Genre has two movies with same rating order by movie name alphabetically\*/

-----------------------------------------------------------------------------------------

WITH HIGHESTRATEDGENRE\_CTE(GENRE,MOVIENAME,Rating)

AS

(

SELECT G.Genre,

M.Name AS MOVIENAME,

MAX(M.Rating)

FROM MOVIE M

LEFT JOIN MOVIETVGenre MTG

ON MTG.MovieID = M.MovieID

LEFT JOIN GENRE G

ON G.GenreID = MTG.GenreID

GROUP BY M.Name, Genre

)

SELECT HCTE.GENRE,

HCTE1.MOVIENAME,

HCTE.HIGHESTRATING

FROM (SELECT MAX(RATING) AS HIGHESTRATING, GENRE

FROM HIGHESTRATEDGENRE\_CTE

GROUP BY GENRE) HCTE

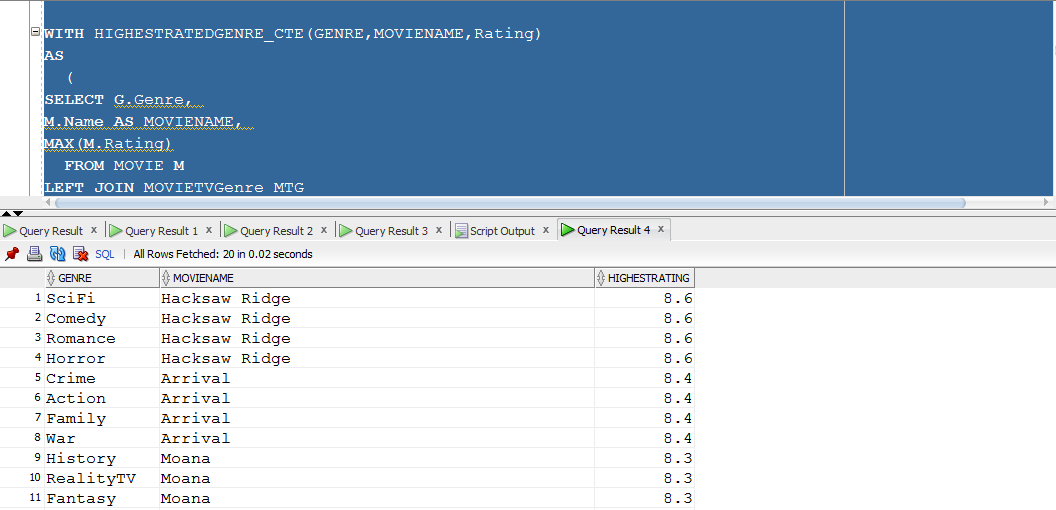
INNER JOIN (SELECT \* FROM HIGHESTRATEDGENRE\_CTE) HCTE1

ON HCTE.GENRE = HCTE1.GENRE

WHERE HCTE.HIGHESTRATING = HCTE1.RATING

AND HCTE.GENRE = HCTE1.GENRE

ORDER BY HCTE.HIGHESTRATING DESC,HCTE1.MOVIENAME;



----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 4

Query : Retrieve Top 10 Movie which has collected highest amount on BoxOffice and Print Number of weeks for which movie was running in descending order\*/

-----------------------------------------------------------------------------------------

SELECT \*

FROM (SELECT C.TOTALBOXOFFICECOLLECTION AS TotalBoxOfficeCollectionInMi,

C.MOVIENAME,

COUNT(DOC.COLLECTIONDAY) AS NumberOfWeeks

FROM (SELECT TO\_CHAR(BOCD.CollectionDate,'DY') AS COLLECTIONDAY,

M.MovieID

FROM MOVIE M

LEFT JOIN BoxOfficeCollectionOfMovie BOCM

ON BOCM.MovieID = M.MovieID

LEFT JOIN BoxOfficeCollectionDATE BOCD

ON BOCD.BoxOfficeCollectionID = BOCM.BoxOfficeCollectionID

GROUP BY BOCD.CollectionDate,M.MovieID

ORDER BY COLLECTIONDAY DESC ) DOC

LEFT JOIN (SELECT M.MovieID,

M.Name AS MOVIENAME,

M.ReleaseDate,

TO\_CHAR(M.ReleaseDate,'DY') AS RELEASEDDAY,

SUM(BOCM.CollectionAmount) AS TOTALBOXOFFICECOLLECTION

FROM MOVIE M

LEFT JOIN BoxOfficeCollectionOfMovie BOCM

ON BOCM.MovieID = M.MovieID

LEFT JOIN BoxOfficeCollectionDATE BOCD

ON BOCD.BoxOfficeCollectionID = BOCM.BoxOfficeCollectionID

GROUP BY M.MovieID,

M.Name,

M.ReleaseDate

ORDER BY TOTALBOXOFFICECOLLECTION DESC) C

ON C.MovieID = DOC.MovieID

WHERE C.RELEASEDDAY = DOC.COLLECTIONDAY

GROUP BY C.TOTALBOXOFFICECOLLECTION,

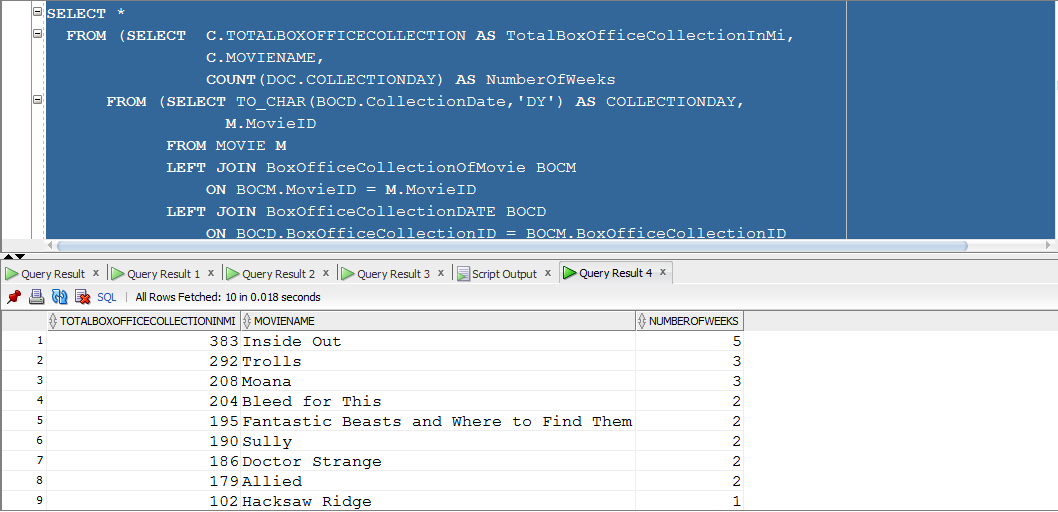
C.MOVIENAME

ORDER BY TotalBoxOfficeCollectionInMi DESC ,

NumberOfWeeks DESC

) Q

WHERE ROWNUM<=10;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 6

Query : NUMBER OF REVIEWS FOR TOP 5 MOVIES BY RATING BY USERS AND CRITICS\*/

-----------------------------------------------------------------------------------------

SELECT COUNT(R.ReviewID) AS "TOTAL\_REVIEWS",

M.MovieID,

M.Name AS MOVIE\_NAME

FROM MOVIE M

INNER JOIN REVIEW R

ON R.MovieID = M.MovieID

WHERE M.MovieID IN (SELECT \* FROM (SELECT M.MovieID

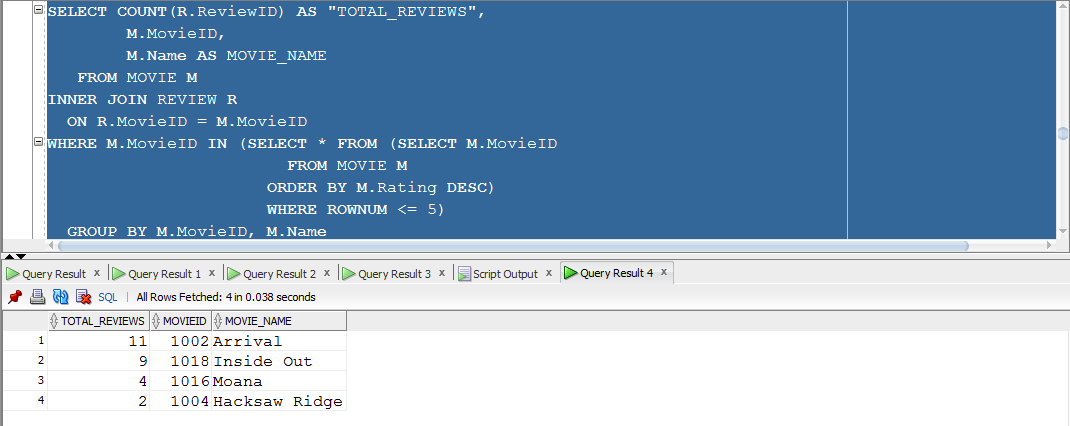
FROM MOVIE M

ORDER BY M.Rating DESC)

WHERE ROWNUM <= 5)

GROUP BY M.MovieID, M.Name

ORDER BY TOTAL\_REVIEWS DESC;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 8

Query : Retrieve number of seasons and episodes of TV Show\*/

-----------------------------------------------------------------------------------------

SELECT SHOW.Name,

SEASON.SeasonNumber, COUNT(EPISODE.EpisodeNumber) AS TOTAL\_EPISODES

FROM TVSHOWs SHOW, TVSeasonShow SEASON, TVSHOWEpisode EPISODE

--CHANNEL CHANNEL, TVSHOWOnChannel SOC, TVSHOWTime ST,

--MovieTVGenre MTG, Genre GENRE

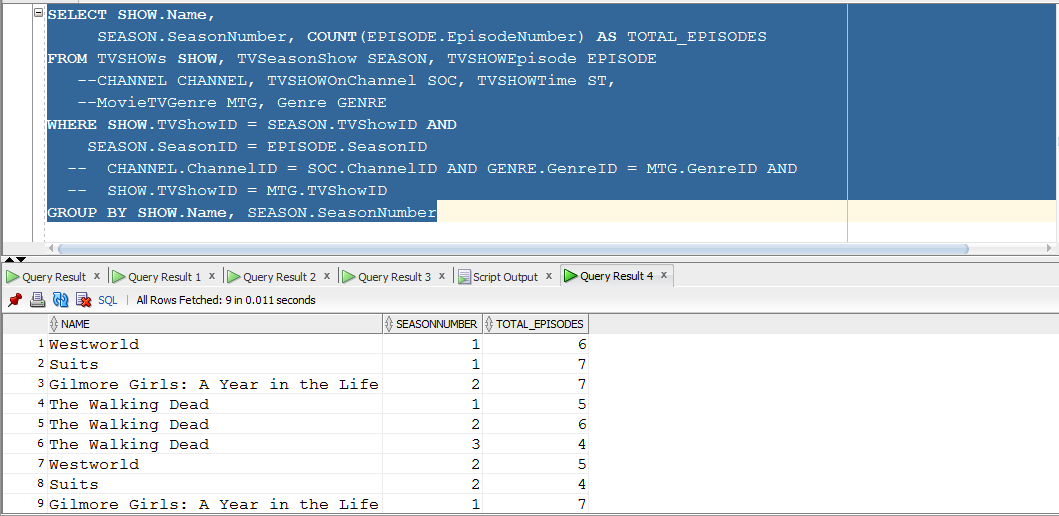
WHERE SHOW.TVShowID = SEASON.TVShowID AND

SEASON.SeasonID = EPISODE.SeasonID

-- CHANNEL.ChannelID = SOC.ChannelID AND GENRE.GenreID = MTG.GenreID AND

-- SHOW.TVShowID = MTG.TVShowID

GROUP BY SHOW.Name, SEASON.SeasonNumber;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 9

Query : Retrieve the details of the TV Shows\*/

-----------------------------------------------------------------------------------------

SELECT SHOW.Name, GENRE.Genre,

SEASON.SeasonNumber, COUNT(EPISODE.EpisodeNumber) AS TOTAL\_EPISODES,

CHANNEL.Channel, ST.ShowTime

FROM TVSHOWs SHOW, TVSeasonShow SEASON, TVSHOWEpisode EPISODE,

CHANNEL CHANNEL, TVSHOWOnChannel SOC, TVSHOWTime ST,

MovieTVGenre MTG, Genre GENRE

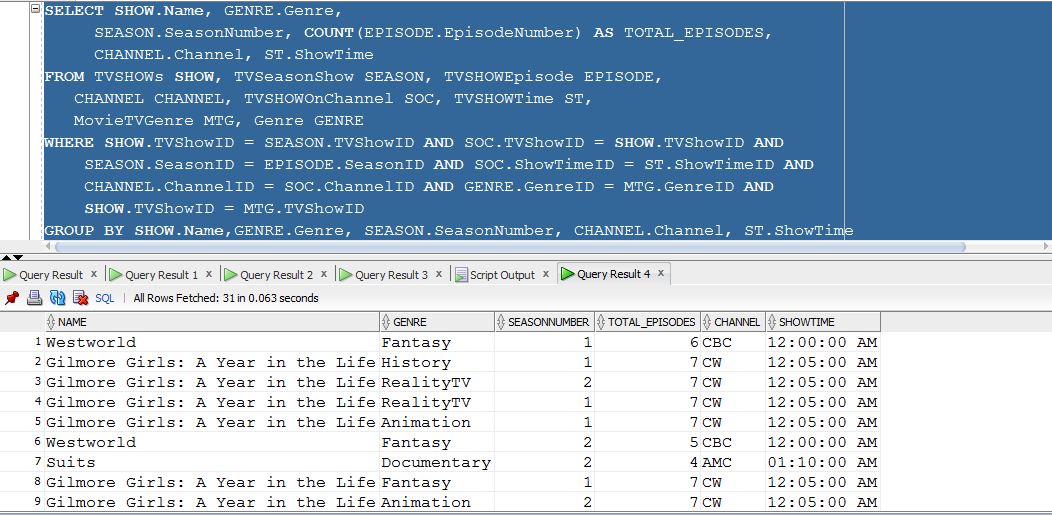
WHERE SHOW.TVShowID = SEASON.TVShowID AND SOC.TVShowID = SHOW.TVShowID AND

SEASON.SeasonID = EPISODE.SeasonID AND SOC.ShowTimeID = ST.ShowTimeID AND

CHANNEL.ChannelID = SOC.ChannelID AND GENRE.GenreID = MTG.GenreID AND

SHOW.TVShowID = MTG.TVShowID

GROUP BY SHOW.Name,GENRE.Genre, SEASON.SeasonNumber, CHANNEL.Channel, ST.ShowTime;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 10

Query : Retrieve the number of Users started particular Discussion in IMDB Community's Messageboard\*/

------------------------------------------------------------------------------------------

SELECT UD.UserID,

P.FirstName ||' '|| P.LastNamE AS "NAME",

MBC.CategoryName,

MBT.TopicName,

MB.DiscussionContent

FROM MESSAGEBOARD MB,

MESSAGEBOARDCategory MBC,

MESSAGEBOARDTopic MBT,

USERDETAILS UD,

PERSON P

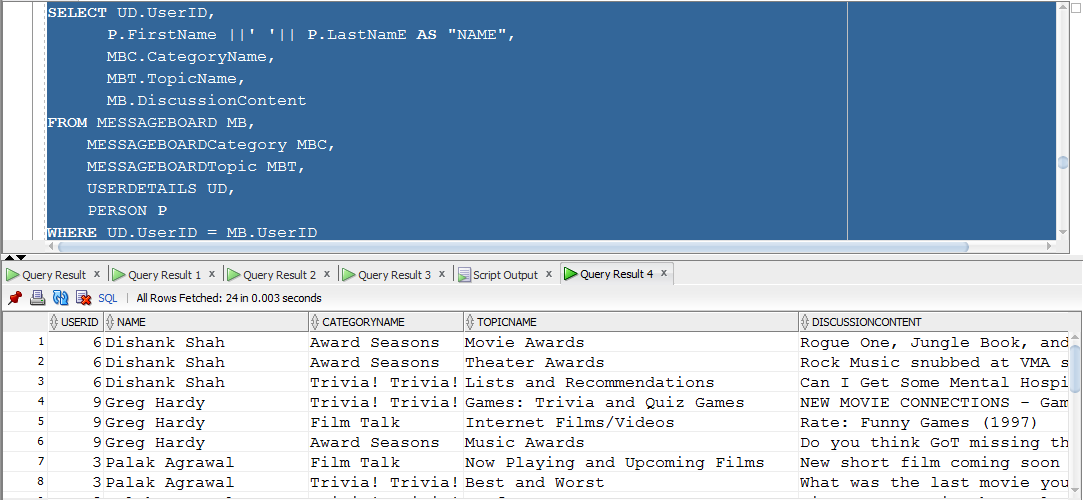
WHERE UD.UserID = MB.UserID

AND MB.TopicID = MBT.TopicID

AND MBC.CategoryID = MBT.CategoryID

AND P.PersonID = UD.UserID

ORDER BY NAME;



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 11

Query : Retrieve the Number of celebrities based on their type\*/

-----------------------------------------------------------------------------------------

SELECT \*

FROM

(SELECT C.NAME,

TC.TYPE

FROM CELEBRITY C,

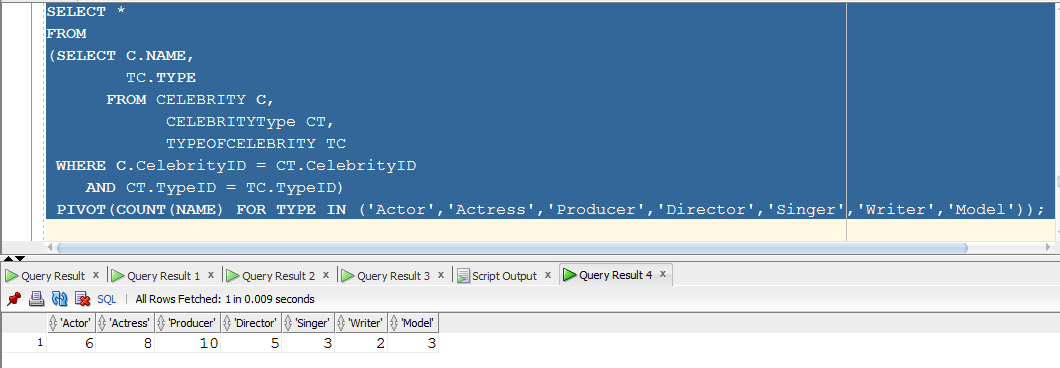
CELEBRITYType CT,

TYPEOFCELEBRITY TC

WHERE C.CelebrityID = CT.CelebrityID

AND CT.TypeID = TC.TypeID)

PIVOT(COUNT(NAME) FOR TYPE IN ('Actor','Actress','Producer','Director','Singer','Writer','Model'));



-----------------------------------------------------------------------------------------

/\*Databases : SQL Server

Author : Group 3

Creation Date : 12-06-2016

Query Number : 12

Query : Retrieve the cheapest ticket for each movie in each theatre for different screens\*/

------------------------------------------------------------------------------------------

SELECT M.Name AS "MOVIENAME",

T.Name AS "THEATRENAME",

S.Name AS "SCREEN",

MIN(MSIT.ShowPrice) AS "MINIMUMPRICE"

FROM MOVIEShowInTheatre MSIT,

MOVIE M,

THEATRE T,

SCREEN S,

SHOWTIME ST,

SHOWDATE SD

WHERE MSIT.MovieID = M.MovieID

AND T.TheatreID = MSIT.TheatreID

AND S.ScreenID = MSIT.ScreenID

AND ST.ShowTimeID = MSIT.ShowTimeID

AND SD.ShowDate = MSIT.ShowDate

GROUP BY M.Name, T.NAME, S.NAME

ORDER BY M.Name DESC, T.Name DESC, S.Name DESC;

