a. I used (EXTRACT(YEAR FROM AGE(current_date, actors."date_of_birth"))) inside the AVG() aggregate function on the actors table to find the average age of all actors.

RESULT:

"avg_age"

"33.2000000000000000"

b. I am using the movie title “Movie A” in my query. I first find the location ids for the birth locations of all actors in the movie “Movie A” using an alias result in the WITH statement: MovieAActorBirthLocations. I then count the number of distinct countries for all these birth locations using SELECT COUNT(DISTINCT locations."Country") AS numDiffCountries.

RESULT:

"numdiffcountries"

"1"

c. I simply use the COUNT(actors.eye_color) FROM actors WHERE eye_color = ‘Green’ to find the number of occurrences of the value ‘Green’ in the eye_color column of the actors table

RESULT:

"count"

"1"

d. I am using the name Anna Davis instead of Brad Pitt. I join the actors table with the actor_work table using the actor ids and then count the number of movie ids where the actors first name is Anna and last name is Davis

RESULT:

"count"

"1"

e. I join the movies table and the genres table on the movie_genre table and then group by genre type. I then run the aggregate functions MAX,MIN and AVG on the movie.budget column to get the average, minimum and maximum budgets of all movies for each type (genre)

RESULT:

"genre","min","max","avg"

"Romance",1200000,1200000,"1200000.000000000000"

"Comedy",1000000,1000000,"1000000.000000000000"

"Horror",1400000,1400000,"1400000.000000000000"

"Drama",1500000,1500000,"1500000.000000000000"

"Action",2000000,2000000,"2000000.000000000000"

f. I joing the actors table with actor_work using the actor id. I joing the movies table to the result using the movies id and I joing the location table to the result using the actor's location id. I then avereaged the movie rating across only those tuples in which the actor's eye colour was Blue or the city of their birth was Toronto.

RESULT:

"avgmovierating"

"8.3333333333333333"

g. I joined the movies table with the actor_work table using the movie ids. I joined the result to the actors table using the actor id. I then joined the result to the locations table using the actor's birth location id. I then grouped the final result by movie title and then returned only those movie titles for whom the count of the distinct actor birth location countries was >= 2. In the case of my particular database, there were no such movies.

h. I left join movies with movie_awards on movies.movie_id = movie_awards.movie_id. I then group by movies.title and order by their award count while displaying the number of awards won by each movie.

RESULT:

"Movie Title","Award Count"

"Movie A","1"

"The Imaginary Quest","1"

"Movie C","1"

"Movie D","1"

"Movie E","1"

"Movie B","1"