

# DB 1차 과제 보고서

12171628 방술찬

연락처 010 7538 3823 bsc981111@naver.com

## 1 구현

각각의 쿼리문들은 `execQuery` 함수와 `execQuerySelect` 함수를 통해서 실행하였습니다. . 두 함수 모두 처음에 `SQLAllocStmt` 함수를 통해 `statement` 핸들을 할당 받고 할당 성공시에 `SQLExecDirect` 함수를 통해 쿼리문을 수행하였습니다. 그 후에 쿼리문 수행이 성공적으로 끝났다면 해당 쿼리문을 출력한 후 `execQuery` 함수는 종료됩니다. 두 함수실행중 에러가 날시 `try catch`문을 통하여 `error trace` 를 출력하도록 하였습니다. `createTables` 함수를 통해서 기본적인 table생성을 하였고 `init` 함수를 통해서 기본 값들을 집어넣었습니다 . 그리고 `main` 문안 코드를 통해서 과제에 주어진 사항들을 수행하였습니다. 모든 쿼리문이 시행된 후에는 table들을 drop한뒤에 `main` 문을 종료하였습니다.

## 2 컴파일 환경 및 언어

JDBC를 이용하여 postgresql에 접근하였고 eclipse java 2019-09에서 java로 컴파일 하여 실행하였습니다.

## 3. 출력결과

1. Create the tables and insert the proper data based on the provided data. You should make the movie, actor, director, and customer tables first and insert data into other related tables.

2. Insert the proper data from the following statements.

2.1. Winona Ryder won the "Best supporting actor" award in 1994

2.2. Andrew Garfield won the "Best supporting actor" award in 2011

```

<terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_231\bin\javaw.exe (2020. 4. 27. 오전 1:59:38)
PostgreSQL JDBC Driver Registered!
org.postgresql.jdbc.PgConnection@2328c243
You made it, take control your database now!

Create table started

Create table ended

init end
2.1 Statement : Winona Ryder won the "Best supporting actor" award in 1994

Translated SQL :insert into award values(1,'Best supporting actor') except select awardID,awardName from award')
Translated SQL :insert into actorObtain values(2, 1,'1994') except select* from actorObtain
Translated SQL :select *from award
-----< award >-----
awardid      awardname
1            Best supporting actor

-----< actorObtain >-----
actorid      awardid      year
2            1            1994

2.2 Statement : Andrew Garfield won the "Best supporting actor" award in 2011

Translated SQL :select actorID from actor where actorName='Andrew Garfield'
Translated SQL :insert into award(awardName) values('Best supporting actor') except select awardName from award
Translated SQL :select awardID from award where awardName='Best supporting actor'
Translated SQL :insert into actorObtain(actorID,awardID,year) values(7,1,2011) except select* from actorObtain
-----< award >-----
awardid      awardname
1            Best supporting actor

-----< actorObtain >-----
actorid      awardid      year
2            1            1994
7            1            2011

```

2.3. Jesse Eisenberg won the "Best main actor" award in 2011

2.4. Johnny Depp won the "Best villain actor" award in 2011

```

Problems Javadoc Console
<terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_231\bin\javaw.exe (2020. 4. 27. 오전 1:59:38)

2.3 Statement : Jesse Eisenberg won the "Best main actor" award in 2011

Translated SQL :select actorID from actor where actorName='Jesse Eisenberg'
Translated SQL :insert into award(awardID,awardName) values(2,'Best main actor') except select* from award
Translated SQL :select awardID from award where awardName='Best supporting actor'
Translated SQL :insert into actorObtain(actorID,awardID,year) values(6,2,2011) except select* from actorObtain
-----< award >-----
awardid      awardname
1            Best supporting actor
2            Best main actor

-----< actorObtain >-----
actorid      awardid      year
2            1            1994
7            1            2011
6            2            2011

2.4 Statement : Johnny Depp won the "Best villain actor" award in 2011

Translated SQL :select actorID from actor where actorName='Johnny Depp'
Translated SQL :insert into award(awardID,awardName) values(3,'Best villain actor') except select* from award
Translated SQL :select awardID from award where awardName='Best villain actor'
Translated SQL :insert into actorObtain(actorID,awardID,year) values(1,3,2011) except select* from actorObtain
-----< award >-----
awardid      awardname
1            Best supporting actor
2            Best main actor
3            Best villain actor

-----< actorObtain >-----
actorid      awardid      year
2            1            1994
7            1            2011
6            2            2011
1            3            2011

```

2.5. Edward Scissorhands won the "Best fantasy movie" award in 1991

2.6. The Dark Knight won the "Best picture" award in 2009

### 2.5. Edward Scissorhands won the "Best fantasy movie" award in 1991

Translated SQL :select movieID from movie where movieName='Edward Scissorhands'

Translated SQL :insert into award(awardID,awardName) values(6,'Best fantasy movie') except select\* from award

Translated SQL :insert into movieObtain(movieID,awardID,year) values(1,6,1991) except select\* from movieObtain

```
-----< award >-----
awardid      awardname
1            Best supporting actor
2            Best main actor
3            Best villain actor
6            Best fantasy movie
```

```
-----< movieObtain >-----
movieid      awardid      year
1            6            1991
```

### 2.6. Alice In Wonderland won the "Best fantasy movie" award in 2011

Translated SQL :select movieID from movie where movieName='Edward Scissorhands'

Translated SQL :insert into award(awardID,awardName) values(6,'Best fantasy movie') except select\* from award

Translated SQL :insert into movieObtain(movieID,awardID,year) values(2,6,2011) except select\* from movieObtain

```
-----< award >-----
awardid      awardname
1            Best supporting actor
2            Best main actor
3            Best villain actor
6            Best fantasy movie
```

```
-----< movieObtain >-----
movieid      awardid      year
1            6            1991
2            6            2011
```

### 2.7. Alice In Wonderland won the "Best fantasy movie" award in 2011

### 2.8. David Fincher won the "Best director" award in 2011

#### 2.7. The Dark Knight won the "Best picture" award in 2009

Translated SQL :select movieID from movie where movieName='Edward Scissorhands'

Translated SQL :insert into award(awardID,awardName) values(7,'Best picture') except select\* from award

Translated SQL :insert into movieObtain(movieID,awardID,year) values(4,7,2009) except select\* from movieObtain

```
-----< award >-----
awardid      awardname
1            Best supporting actor
2            Best main actor
3            Best villain actor
6            Best fantasy movie
7            Best picture
```

```
-----< movieObtain >-----
movieid      awardid      year
1            6            1991
2            6            2011
4            7            2009
```

#### 2.8 David Fincher won the "Best director" award in 2011

Translated SQL :insert into award(awardID,awardName) values(5,'Best director') except select \* from award

Translated SQL :select directorID from director where directorName='David Fincher'

Translated SQL :insert into directorObtain(directorID,awardID,year) values(2,5,2011) except select\* from directorObtain

```
-----< actor >-----
actorid      actorname      dateofbirth      dateofdeath      gender
1            Johnny Depp      1963.6.9         null             Male
2            Winona Ryder      1971.10.29       null             Female
3            Anne Hathaway      1982.11.12       null             Female
4            Christian Bale      1974.1.30        null             Male
5            Heath Ledger      1794.4.4         2008.1.22       Male
6            Jesse Eisenberg      1983.10.5        null             Male
7            Andrew Garfield      1983.8.20        null             Male
```

```
-----< award >-----
awardid      awardname
1            Best supporting actor
2            Best main actor
3            Best villain actor
6            Best fantasy movie
7            Best picture
5            Best director
```

3. Insert data to the proper tables based on the following statements and update avgRate if necessary.

#### 3.1 Bob rates 5 to "The Dark Knight".

### 3.2 Bell rates 5 to the movies whose director is "Tim Burton".

3.1 Bob rates 5 to 'The Dark Knight'

Translated SQL :select customerID from customer where customerName='Bob'

Translated SQL :select movieID from movie where movieName='The Dark Knight'

Translated SQL :insert into customerRate(customerID,movieID,rate) values(1,4,5) except select\* from customerRate

Translated SQL :update movie M set avgRate=(select sum(rate)/count(rate)  
from customerRate  
where M.movieID=customerRate.movieID  
)

-----< movie >-----

movieid	moviename	releaseyear	releasemonth	releasedate	publishername
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents
2	Alice In Wonderland	2010	03	04	Korea Sony Pictures
3	The Social Network	2010	11	18	Korea Sony Pictures
4	The Dark Knight	2008	08	06	Warner Brothers Korea

-----< customerRate >-----

customerid	movieid	rate
1	4	5.0

3.2 Bell rates 5 to the movies whose director is "Tim Burton"

Translated SQL :select directorID from director where directorName='Tim Burton'

Translated SQL :select customerID  
from customer  
where customerName='Bell'

Translated SQL :insert into customerRate  
select 5 as customerID,movieID,5.0 as rate from make where directorID=1 except select\* from customerRate

Translated SQL :update movie M set avgRate=(select sum(rate)/count(rate)  
from customerRate  
where M.movieID=customerRate.movieID  
)

-----< movie >-----

movieid	moviename	releaseyear	releasemonth	releasedate	publishername
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents
2	Alice In Wonderland	2010	03	04	Korea Sony Pictures
3	The Social Network	2010	11	18	Korea Sony Pictures
4	The Dark Knight	2008	08	06	Warner Brothers Korea

-----< customerRate >-----

customerid	movieid	rate
1	4	5.0
5	2	5.0
5	1	5.0

Updates Available x

Updates are available for your software.  
Click to review and install updates.  
[Set up Reminder options](#)

### 3.3 Jill rates 4 to the movies whose main actor is female.

3.3 Jill rates 4 to the movies whose main actor is female.

Translated SQL :select customerID  
from customer  
where customerName='Jill'

Translated SQL :insert into customerRate  
select distinct 4 as customerID, movieID ,4.0 as rate  
from casting natural join actor  
where gender='Female'  
and role='Main actor' except select\* from customerRate

Translated SQL :update movie M set avgRate=(select sum(rate)/count(rate)  
from customerRate  
where M.movieID=customerRate.movieID  
)

-----< movie >-----

movieid	moviename	releaseyear	releasemonth	releasedate	publishername
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents
2	Alice In Wonderland	2010	03	04	Korea Sony Pictures
3	The Social Network	2010	11	18	Korea Sony Pictures
4	The Dark Knight	2008	08	06	Warner Brothers Korea

-----< customerRate >-----

customerid	movieid	rate
1	4	5.0
5	2	5.0
5	1	5.0
4	2	4.0
4	1	4.0

### 3.4 Jack rates 4 to the fantasy movies

Jack rates 4 to the fantasy movies.

Translated SQL :select customerID

from customer

where customerName='Jack'

Translated SQL :insert into customerRate select distinct 3 as customerID,movieID,4.0 as rate from movieGenre where genreName='Fantasy' except sele

Translated SQL :update movie M set avgRate=(select sum(rate)/count(rate)  
from customerRate  
where M.movieID=customerRate.movieID  
)

< movie >							
movieid	moviename	releaseyear	releasemonth	releasedate	publishername		
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents		
2	Alice In Wonderland	2010	03	04	Korea Sonry Pictures		
3	The Social Network	2010	11	18	Korea Sony Pictures		
4	The Dark Knight	2008	08	06	Warner Brothers Korea		

< customerRate >			
customerid	movieid	rate	
1	4	5.0	
5	2	5.0	
5	1	5.0	
4	2	4.0	
4	1	4.0	
3	2	4.0	
3	1	4.0	

3.5 John rates 5 to the movies whose director won the "Best director" award

4. Select the names of the movies whose actor are dead.

5. Select the names of the directors who cast the same actor more than once.

3.5 John rates 5 to the movies whose director won the "Best director" award

Translated SQL :select customerID

from customer

where customerName='John'

Translated SQL :insert into customerRate select distinct 2 as customerID,movieID, 5.0 as rate from award natural join directorObtain natural join

Translated SQL :update movie M set avgRate=(select sum(rate)/count(rate)  
from customerRate  
where M.movieID=customerRate.movieID  
)

< movie >							
movieid	moviename	releaseyear	releasemonth	releasedate	publishername		
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents		
2	Alice In Wonderland	2010	03	04	Korea Sonry Pictures		
3	The Social Network	2010	11	18	Korea Sony Pictures		
4	The Dark Knight	2008	08	06	Warner Brothers Korea		

< customerRate >			
customerid	movieid	rate	
1	4	5.0	
5	2	5.0	
5	1	5.0	
4	2	4.0	
4	1	4.0	
3	2	4.0	
3	1	4.0	
2	3	5.0	

4. Select the names of the movies whose actor are dead.

Translated SQL :select movieName from actor natural join casting natural join movie where dateofdeath is not null

moviename  
The Dark Knight

5. Select the names of the directors who cast the same actor more than once.

Translated SQL :select directorID,actorID,count (actorID) from casting natural join make natural join director group by actorID,directorID having  
directorid actorid count

6. Select the names of the movies and the genres, where movies have the common genre.

7. Delete the movies which did not get any award (including all directors and actors) and delete data from related tables

Translated SQL :delete from movie where movie.movieID not in (select movieID from casting natural join actorObtain) and movie.movieID not in(select

< movie >-----						
movieid	moviename	releaseyear	releasemonth	releasedate	publishername	
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents	
2	Alice In Wonderland	2010	03	04	Korea Sonry Pictures	
3	The Social Network	2010	11	18	Korea Sony Pictures	
-----< casting >-----						
movieid	actorid	role				
1	1	Main actor				
1	2	Main actor				
2	2	Main actor				
2	3	Main actor				
3	6	Main actor				
3	7	Supporting Actor				
-----< make >-----						
movieid	directorid					
1	1					
2	1					
3	2					
-----< movieGenre >-----						
movieid	genrename					
1	Fantasy					
1	Romance					
2	Fantasy					
2	Adventure					
2	Family					
3	Drama					
-----< movieObtain >-----						
movieid	awardid	year				
1	6	1991				
2	6	2011				
-----< customerRate >-----						
customerid	movieid	rate				
5	2	5.0				
5	1	5.0				
4	2	4.0				
4	1	4.0				
3	2	4.0				
3	1	4.0				
2	3	5.0				

## 8. Delete all customers and delete data from related tables.

8. Delete all customers and delete data from related tables.

Translated SQL :delete from customer

< movie >-----						
movieid	moviename	releaseyear	releasemonth	releasedate	publishername	
1	Edward Scissorhands	1991	06	29	20th Century Fox Presents	
2	Alice In Wonderland	2010	03	04	Korea Sonry Pictures	
3	The Social Network	2010	11	18	Korea Sony Pictures	
-----< casting >-----						
movieid	actorid	role				
1	1	Main actor				
1	2	Main actor				
2	2	Main actor				
2	3	Main actor				
3	6	Main actor				
3	7	Supporting Actor				
-----< make >-----						
movieid	directorid					
1	1					
2	1					
3	2					
-----< movieGenre >-----						
movieid	genrename					
1	Fantasy					
1	Romance					
2	Fantasy					
2	Adventure					
2	Family					
3	Drama					
-----< movieObtain >-----						
movieid	awardid	year				
1	6	1991				
2	6	2011				
-----< customerRate >-----						
customerid	movieid	rate				
-----< customer >-----						
customerid	customername	dateofbirth	gender			

## 9. Delete all tables and data.

9. Delete all customers and delete data from related tables.

```
Translated SQL :drop table customerRate
Translated SQL :drop table make
Translated SQL :drop table casting
Translated SQL :drop table directorObtain
Translated SQL :drop table actorObtain
Translated SQL :drop table movieObtain
Translated SQL :drop table movieGenre
Translated SQL :drop table movie
Translated SQL :drop table director
Translated SQL :drop table actor
Translated SQL :drop table customer
Translated SQL :drop table award
Translated SQL :drop table genre
table deleted
```

---