**Buat DB**

postgres=# Create Database movie;

**Masuk databse movie**

postgres-# \c movie

You are now connected to database "movie" as user "postgres".

**Create Table**

movie=# create table actor (

movie(# act\_id int primary key,

movie(# act\_fname varchar,

movie(# act\_lname varchar,

movie(# act\_gender varchar

movie(# );

CREATE TABLE

**Melihat tabel**

movie-# \d

List of relations

Schema | Name | Type | Owner

--------+-----------------+-------+----------

public | actor | table | postgres

public | director | table | postgres

public | genres | table | postgres

public | movie | table | postgres

public | movie\_cast | table | postgres

public | movie\_direction | table | postgres

public | movie\_genres | table | postgres

public | rating | table | postgres

public | reviewer | table | postgres

(9 rows)

**Insert Tabel**

movie=# INSERT INTO actor (act\_id,act\_fname,act\_lname,act\_gender) VALUES

movie-# ('101','James','Stewart','M'),

movie-# ('102','Deborah','Kerr','F'),

movie-# ('103','Peter','OToole','M'),

movie-# ('104','Robert','De Niro','M'),

movie-# ('105','F. Murray','Abraham','M'),

movie-# ('106','Harrison','Ford','M'),

movie-# ('107','Nicole','Kidman','F'),

movie-# ('108','Stephen','Baldwin','M'),

movie-# ('109','Jack','Nicholson','M'),

movie-# ('110','Mark','Wahlberg','M'),

movie-# ('111','Woody','Allen','M'),

movie-# ('112','Claire','Danes','F'),

movie-# ('113','Tim','Robbins','M'),

movie-# ('114','Kevin','Spacey','M'),

movie-# ('115','Kate','Winslet','F'),

movie-# ('116','Robin','Williams','M'),

movie-# ('117','Jon','Voight','M'),

movie-# ('118','Ewan','McGregor','M'),

movie-# ('119','Christian','Bale','M'),

movie-# ('120','Maggie','Gyllenhaal','F'),

movie-# ('121','Dev','Patel','M'),

movie-# ('122','Sigourney','Weaver','F'),

movie-# ('123','David','Aston','M'),

movie-# ('124','Ali','Astin','F');

INSERT 0 24

**Basic SQL Command**

movie=# Select \* From actor;

act\_id | act\_fname | act\_lname | act\_gender

--------+-----------+------------+------------

101 | James | Stewart | M

102 | Deborah | Kerr | F

103 | Peter | OToole | M

104 | Robert | De Niro | M

105 | F. Murray | Abraham | M

106 | Harrison | Ford | M

107 | Nicole | Kidman | F

108 | Stephen | Baldwin | M

109 | Jack | Nicholson | M

110 | Mark | Wahlberg | M

111 | Woody | Allen | M

112 | Claire | Danes | F

113 | Tim | Robbins | M

114 | Kevin | Spacey | M

115 | Kate | Winslet | F

116 | Robin | Williams | M

117 | Jon | Voight | M

118 | Ewan | McGregor | M

119 | Christian | Bale | M

120 | Maggie | Gyllenhaal | F

121 | Dev | Patel | M

122 | Sigourney | Weaver | F

123 | David | Aston | M

124 | Ali | Astin | F

(24 rows)

**Melihat banyaknya film berdasarkan bahasa**

movie=# select mov\_lang, count(mov\_id) from movie group by mov\_lang;

mov\_lang | count

----------+-------

English | 25

Japanese | 3

(2 rows)

**Pemakaian statemen kondisi**

**Memberikan lavel pada kondisi film berdasarkan rating**

movie=# select mov\_id, rev\_stars,

movie-# case

movie-# when rev\_stars between 0 and 5 then 'film jelek'

movie-# when rev\_stars between 5 and 7 then 'film lumayan'

movie-# when rev\_stars between 7 and 10 then 'film bagus'

movie-# else 'belum ada rating'

movie-# end as status\_rating

movie-# from rating;

mov\_id | rev\_stars | status\_rating

--------+-----------+------------------

901 | 8.40 | film bagus

902 | 7.90 | film bagus

903 | 8.30 | film bagus

906 | 8.20 | film bagus

924 | 7.30 | film bagus

908 | 8.60 | film bagus

909 | | belum ada rating

910 | 3.00 | film jelek

911 | 8.10 | film bagus

912 | 8.40 | film bagus

914 | 7.00 | film lumayan

915 | 7.70 | film bagus

916 | 4.00 | film jelek

925 | 7.70 | film bagus

918 | | belum ada rating

920 | 8.10 | film bagus

921 | 8.00 | film bagus

922 | 8.40 | film bagus

923 | 6.70 | film lumayan

(19 rows)

**Latihan**

1. **Find a movie title that has a character named Alice Harford**

movie=# select m.mov\_id, m.mov\_title, mc.role

movie-# from movie m

movie-# inner join movie\_cast mc on m.mov\_id = mc.mov\_id

movie-# where mc.role = 'Alice Harford';

mov\_id | mov\_title | role

--------+----------------+---------------

907 | Eyes Wide Shut | Alice Harford

(1 row)

1. **Find each genre of each movies**

movie=# select m.mov\_id, m.mov\_title, mgenres.gent

movie-# from movie m

movie-# left join (select mg.mov\_id as movid, g.gen\_title as gent from movie\_genres mg, genres g where mg.gen\_id=g.gen\_id) mgenres on m.mov\_id = mgenres.movid;

mov\_id | mov\_title | gent

--------+--------------------------+-----------

922 | Aliens | Action

917 | Deliverance | Adventure

903 | Lawrence of Arabia | Adventure

912 | Princess Mononoke | Animation

911 | Annie Hall | Comedy

908 | The Usual Suspects | Crime

913 | The Shawshank Redemption | Crime

926 | Seven Samurai | Drama

928 | Back to the Future | Drama

918 | Trainspotting | Drama

921 | Slumdog Millionaire | Drama

902 | The Innocents | Horror

923 | Beyond the Sea | Music

907 | Eyes Wide Shut | Mystery

927 | Spirited Away | Mystery

901 | Vertigo | Mystery

914 | American Beauty | Romance

906 | Blade Runner | Thriller

904 | The Deer Hunter | War

910 | Boogie Nights |

919 | The Prestige |

915 | Titanic |

909 | Chinatown |

905 | Amadeus |

924 | Avatar |

920 | Donnie Darko |

916 | Good Will Hunting |

925 | Braveheart |

(28 rows)

1. **Find the most favorite genre (using average rating of each genre)**
2. **Find what year that has most movies (movies of each year)**

movie=# select count(mov\_id) total, mov\_year

movie-# from movie

movie-# group by mov\_year

movie-# order by total desc

movie-# limit 1;

total | mov\_year

-------+----------

4 | 1997

(1 row)

1. **Find actor that has played as Sean Maguire**

movie=# select m.mov\_title, concat(a.act\_fname, ' ',a.act\_lname) actor\_name, c.role

movie-# from movie m join movie\_cast c on m.mov\_id = c.mov\_id

movie-# join actor a on a.act\_id = c.act\_id

movie-# where c.role='Sean Maguire';

mov\_title | actor\_name | role

-------------------+----------------+--------------

Good Will Hunting | Robin Williams | Sean Maguire

(1 row)

1. Find a movie with the lowest rating

movie=# select r.rev\_stars, m.mov\_title

movie-# from movie m join rating r on m.mov\_id = r.mov\_id

movie-# order by r.rev\_stars asc

movie-# limit 1;

rev\_stars | mov\_title

-----------+---------------

3.00 | Boogie Nights

(1 row)

1. Find how many female actor who acted in 21st century
2. Find actor that has played in more than one movie
3. Find female actor whose movies received a highest rating

movie=# select a.act\_id, concat(a.act\_fname, ' ',a.act\_lname) actor\_name, r.rev\_stars

movie-# from actor a

movie-# inner join movie\_cast mc on a.act\_id = mc.act\_id

movie-# inner join rating r on r.mov\_id = mc.mov\_id

movie-# where a.act\_gender = 'F'

movie-# order by r.rev\_stars desc

movie-# limit 1;

act\_id | actor\_name | rev\_stars

--------+--------------+-----------

112 | Claire Danes | 8.40

(1 row)

1. Find reviewer who never give the low rating (below 8)
2. Find the number of director who directed drama movies
3. Count the number of male and female actor

movie=# select act\_gender, count(act\_gender) from actor group by act\_gender;

act\_gender | count

------------+-------

M | 17

F | 7

(2 rows)

**Atau**

movie=# select

movie-# case

movie-# when act\_gender = 'M' then 'Male'

movie-# when act\_gender = 'F' then 'Female'

movie-# else 'no gender'

movie-# end as gender\_label,

movie-# count(act\_gender)

movie-# from actor

movie-# group by act\_gender;

gender\_label | count

--------------+-------

Male | 17

Female | 7

(2 rows)