------ CineInsight: Analyzing IMDb's Movie_Database Through _SQL ----------- Project by Subhadeep Das ------

-- create schema

create schema imdb;

-- use database

use imdb;

-- create table director

create table director(name varchar(100) not null, id int not null, gender varchar(10) not null, uid int not null, department varchar(100) not null, primary key(uid));

-- change 0 1 2 to respective gender

UPDATE director
SET gender = CASE
WHEN gender = 0 THEN 'other'
WHEN gender = 1 THEN 'female'
WHEN gender = 2 THEN 'male'
END;

	A	D	C	U	E	
1	name	id	gender	uid	department	
44	Robert Zemeckis	4804	2	24	Directing	
45	Lilly Wachowski	4805	1	9339	Directing	
46	Pete Docter	4806	2	12890	Directing	
47	Conrad Vernon	4807	2	12080	Directing	
48	Jon Favreau	4808	2	15277	Directing	
49	Martin Scorsese	4809	2	1032	Directing	
50	Roh Cohen	4810	2	18878	Directing	

previously

-- show table director

select * from director;

Re	esult Grid 🔢	Filter Ro	ws:		Edit: [
	name	id	gender	uid	department		
•	George Lucas	4901	male	1	Directing		
	Andrew Stanton	4766	male	7	Directing		
	Lee Unkrich	4792	male	8	Directing		
	Robert Zemeckis	4804	male	24	Directing		
	Sam Mendes	4764	male	39	Directing		
	Luc Besson	4949	male	59	Directing		
	Peter Jackson	4777	male	108	Directing		
	Quentin Tarantino	4927	male	138	Directing		
	Clint Eastwood	5087	male	190	Directing		
	David Silverman	5050	male	197	Directing		
	Michel Gondry	4911	male	201	Directing		
	Alejandro GonzÃi	4874	male	223	Directing		
	David Cronenberg	5229	male	224	Directing		
	Stanley Kubrick	5089	male	240	Directing		
dir	Terry Cilliam ector 18 ×	5014	mala	280	Direction		

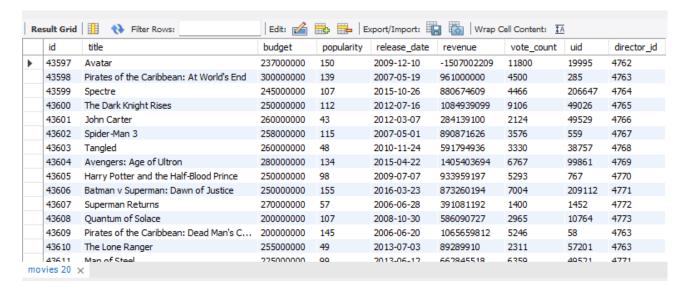
now

-- create table movies

create table movies(id int not null, title varchar(100) not null, budget bigint not null, popularity int not null, release_date date not null, revenue bigint, vote_count int not null, uid int not null, director_id int not null, primary key(id));

-- show table movies

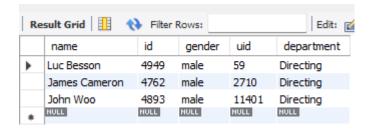
select * from movies;



------ BASIC ANALYSIS ------

-- Q1. Find these 3 directors: James Cameron, Luc Besson, John Woo

select * from director where name in('James Cameron', 'Luc Besson', 'John Woo');



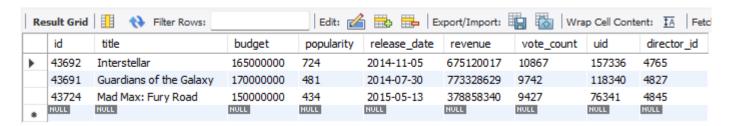
-- Q2. Count the female directors

select count(*) from director where gender = 'female';



-- Q3. What are the 3 most popular movies

select * from movies order by popularity desc limit 3;



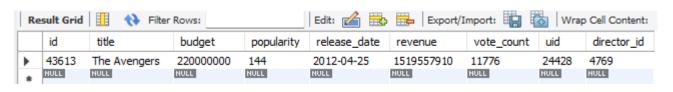
-- Q4. What are 3 most bankable movies

select * from movies order by budget desc limit 3;

Result Grid III 🛟 Filter Rows: Edit: 🕍 🐯 🖶 Export/Import: 📳 🔯 Wrap Cell Content: 🔼 Fetch rows:										
		id	title	budget	popularity	release_date	revenue	vote_count	uid	director_id
	•	43614	Pirates of the Caribbean: On Stranger Tides	380000000	135	2011-05-14	1045713802	4948	1865	4775
		43598	Pirates of the Caribbean: At World's End	300000000	139	2007-05-19	961000000	4500	285	4763
		43604	Avengers: Age of Ultron	280000000	134	2015-04-22	1405403694	6767	99861	4769
		NULL	NULL	NULL	NULL	NULL	NULL	NULL	HULL	NULL

-- Q5. Which is the most profitable movie since Jan 1st, 2010

select * from movies where release_date > '2010-01-01' order by revenue desc limit 1;



------JOINS ------

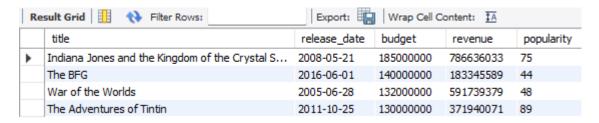
-- Q6. Which movie(s) were directed by Brenda Chapman

select title from movies join director on movies.director_id = director.id where director.name = 'Brenda Chapman';



-- Q7. Give the title, release date, budget, revenue, populartity movies made by Steven Spielberg

select title, release_date, budget, revenue, popularity from movies join director on movies.director_id = director.id where director.name = 'Steven Spielberg';



------ REVENUE ANALYSIS ------

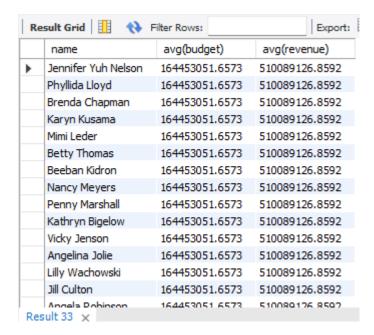
-- Q8. Find the top 10 highest revenue making movies

select title from movies order by revenue limit 10;



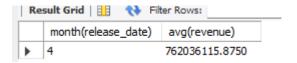
-- Q9. Find the average budget and revenue for movies directed by female directors.

select name, avg(budget), avg(revenue) from movies join director where director.gender = 'female' group by director.name;



-- Q10. Calculate the month with the highest average revenue across all years

select month(release_date), avg(revenue) from movies group by month(release_date) order by avg(revenue) desc limit 1;



------ THE END ------