

Create SQL queries to the database

Новая база данных Открыть базу данных Записать изменения Отменить изменения Открыть проект Сохранить проект Прикрепить БД Закрыть базу данных		
Структура БД Данные Прогнозы SQL		
Создать таблицу Создать индекс Печать		
Имя	Тип	Схема
Таблицы (5)		
directors		CREATE TABLE directors (movie_id INTEGER NOT NULL, person_id INTEGER NOT NULL, FOREIGN KEY(movie_id) REFERENCES movies(id), FOREIGN KEY(person_id) REFERENCES people(id))
movie_id	INTEGER	"movie_id" INTEGER NOT NULL
person_id	INTEGER	"person_id" INTEGER NOT NULL
movies		CREATE TABLE movies (id INTEGER, title TEXT NOT NULL, year NUMERIC, PRIMARY KEY(id))
id	INTEGER	"id" INTEGER
title	TEXT	"title" TEXT NOT NULL
year	NUMERIC	"year" NUMERIC
people		CREATE TABLE people (id INTEGER, name TEXT NOT NULL, birth NUMERIC, PRIMARY KEY(id))
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
birth	NUMERIC	"birth" NUMERIC
ratings		CREATE TABLE ratings (movie_id INTEGER NOT NULL, rating REAL NOT NULL, votes INTEGER NOT NULL, FOREIGN KEY(movie_id) REFERENCES movies(id))
movie_id	INTEGER	"movie_id" INTEGER NOT NULL
rating	REAL	"rating" REAL NOT NULL
votes	INTEGER	"votes" INTEGER NOT NULL
stars		CREATE TABLE stars (movie_id INTEGER NOT NULL, person_id INTEGER NOT NULL, FOREIGN KEY(movie_id) REFERENCES movies(id), FOREIGN KEY(person_id) REFERENCES people(id))
movie_id	INTEGER	"movie_id" INTEGER NOT NULL
person_id	INTEGER	"person_id" INTEGER NOT NULL

Display the list of titles of all films of 2010

Структура БД | Данные | Прогнозы | SQL

SQL 1 | SQL 2 | SQL 3 | SQL 4 | SQL 5 | SQL 6 | SQL 7 | SQL 8 | SQL 9

```
1 SELECT title
2 FROM movies
3 WHERE year=2010;
```

	title
11697	Wandering Eyes
11698	Green's 284 Blue's 278
11699	Teletubbies in Taipei: Workers Against ...
11700	Swing
11701	Mama
11702	Lady Ninja Kasumi 10
11703	Zangaar
11704	Caccia SpA
11705	Santos Dumont: Pré-Cineasta?
11706	Pangaa Gang
11707	Michael Flatley in Feet of Flames Taiwan
11708	Gammalsvenskby

Execution finished without errors.
Result: 11708 строк возвращено за 272мс
At line 1:
SELECT title
FROM movies
WHERE year=2010;

Display the year of birth of the actress Emma Stone

Структура БД | Данные | Прагмы | SQL

SQL 1 x SQL 2 x SQL 3 x SQL 4 x SQL 5 x SQL 6 x

```
1 SELECT birth
2 FROM people
3 WHERE name="Emma Stone";
```

	birth
1	1988

Execution finished without errors.
Result: 1 строк возвращено за 689мс
At line 1:
SELECT birth
FROM people
WHERE name="Emma Stone";

List all movies released in or after 2018 in alphabetical order

Структура БД | Данные | Прагмы | SQL

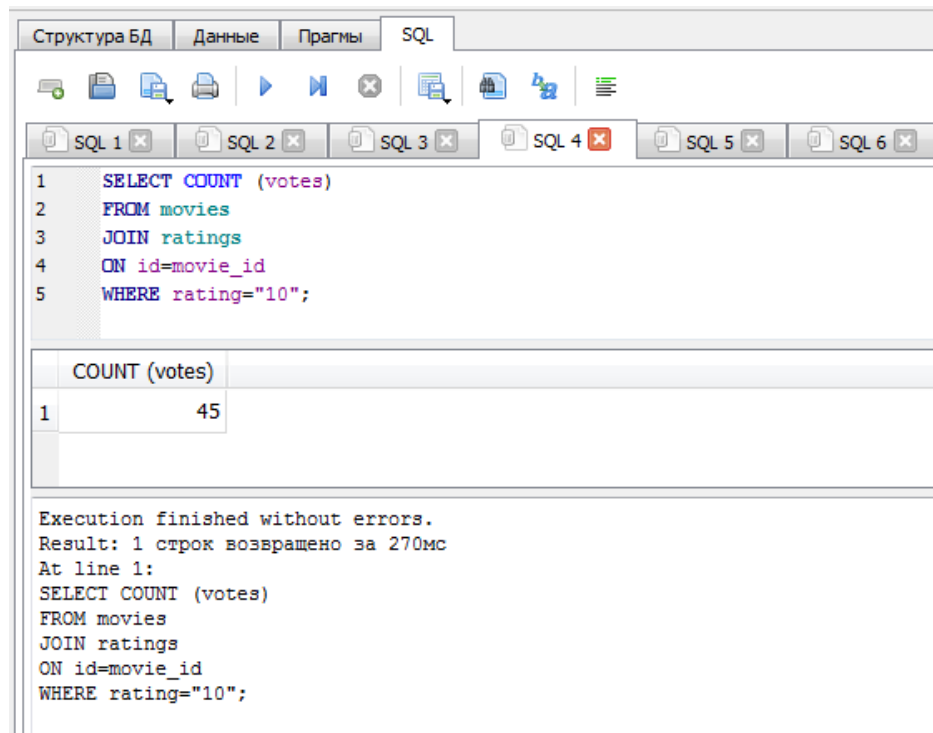
SQL 1 x SQL 2 x SQL 3 x SQL 4 x SQL 5 x SQL 6 x SQL 7 x SQL 8 x

```
1 SELECT title, year
2 FROM movies
3 WHERE year>=2018
4 ORDER BY title ASC;
```

	title	year
1	#1915House	2018
2	#5	2018
3	#AbroHilo	2019
4	#Ahlam	2020
5	#Alive	2020
6	#Anne Frank Parallel Stories	2019
7	#BerhentiKamu	2020
8	#Bicyclediary	2018
9	#BigFertility: It's All About The Money	2018
10	#BlackSkin	2021
11	#Blessed	2020
12	#Bernstein	2019

Execution finished without errors.
Result: 50866 строк возвращено за 651мс
At line 1:
SELECT title, year
FROM movies
WHERE year>=2018
ORDER BY title ASC;

Display the number of films with a rating of 10.0



The screenshot shows a SQL IDE interface with a menu bar (Структура БД, Данные, Прагмы, SQL) and a toolbar. Below the toolbar is a tabbed window with six tabs labeled SQL 1 through SQL 6. The active tab is SQL 4, which contains the following SQL query:

```
1 SELECT COUNT (votes)
2 FROM movies
3 JOIN ratings
4 ON id=movie_id
5 WHERE rating="10";
```

Below the query editor, the results are displayed in a table with two columns: COUNT (votes) and an unnamed column. The table contains one row with the value 45.

	COUNT (votes)	
1	45	

At the bottom of the IDE, the execution status is shown:

```
Execution finished without errors.
Result: 1 строк возвращено за 270мс
At line 1:
SELECT COUNT (votes)
FROM movies
JOIN ratings
ON id=movie_id
WHERE rating="10";
```

Display the names of films with a rating of 10.0

Структура БД | Данные | Прагмы | SQL

SQL 1 | SQL 2 | SQL 3 | SQL 4 | SQL 5 | SQL 6

```
1 SELECT movies.title,ratings.rating
2 FROM movies
3 JOIN ratings
4 ON id=movie_id
5 WHERE rating="10";
```

	title	rating
1	Between the Solitudes	10.0
2	Gatsby in Connecticut: The Untold Story	10.0
3	Identitas - Uguali ma diversi	10.0
4	The Easy Bit	10.0
5	Bure bareta	10.0
6	Eco-Terrorist: The Battle for Our Planet	10.0
7	Rainaldo Graziani da Meridiano Zero al Soggetto ...	10.0
8	Shree ram samartha	10.0
9	Ines Pedretti contro i sensi vietati, le strade del ...	10.0
10	Water get no enemy	10.0
11	La meta è il viaggio nella postmodernità - Murelli...	10.0
12	Sadika: The Multi-Dimensional Artist	10.0
13	Days of Géants	10.0
14	Neofascismo? Rispettatelo! Da giovane ebbe un ...	10.0

Execution finished without errors.
Result: 45 строк возвращено за 106мс
At line 1:
SELECT movies.title,ratings.rating
FROM movies
JOIN ratings
ON id=movie_id
WHERE rating="10";

Display a list of all Harry Potter films in chronological order

The screenshot shows a SQL query execution interface. The top menu bar includes 'Структура БД', 'Данные', 'Прагмы', and 'SQL'. Below the menu is a toolbar with various icons. The main window displays a SQL query in a text editor, followed by a table of results and a status message.

```
1 SELECT title, year
2 FROM movies
3 WHERE title LIKE "%harry potter%"
4 ORDER BY year ASC
```

	title	year
1	Harry Potter and the Sorcerer's Stone	2001
2	Harry Potter and the Chamber of Secrets	2002
3	Harry Potter and the Prisoner of Azkaban	2004
4	Harry Potter and the Goblet of Fire	2005
5	Harry Potter and the Order of the Phoenix	2007
6	Harry Potter and the Half-Blood Prince	2009
7	Harry Potter and the Deathly Hallows: Part 1	2010
8	Creating the World of Harry Potter, Part 4: Sound ...	2010
9	The Seekers Guide to Harry Potter	2010
10	Harry Potter and the Deathly Hallows: Part 2	2011

Execution finished without errors.
Result: 14 строк возвращено за 634мс
At line 1:
SELECT title, year
FROM movies
WHERE title LIKE "%harry potter%"
ORDER BY year ASC

Display the average rating of all films of 2012

The screenshot shows a SQL IDE interface with a menu bar (Структура БД, Данные, Прагмы, SQL) and a toolbar. Below the toolbar is a tabbed window with five tabs labeled SQL 1, SQL 2, SQL 3, SQL 4, and SQL 5. The SQL 1 tab is active and contains the following SQL query:

```
1 SELECT AVG (rating)
2 FROM movies
3 JOIN ratings
4 ON movies.id=ratings.movie_id
5 WHERE year=2012;
```

Below the query editor, the results of the query are displayed in a table with one column, "AVG (rating)". The table contains one row with the value 6.25052023121385.

	AVG (rating)
1	6.25052023121385

At the bottom of the IDE, the execution status is shown:

```
Execution finished without errors.
Result: 1 строка возвращено за 816мс
At line 1:
SELECT AVG (rating)
FROM movies
JOIN ratings
ON movies.id=ratings.movie_id
WHERE year=2012;
```

Display the names of all the actors who played in Toy Story

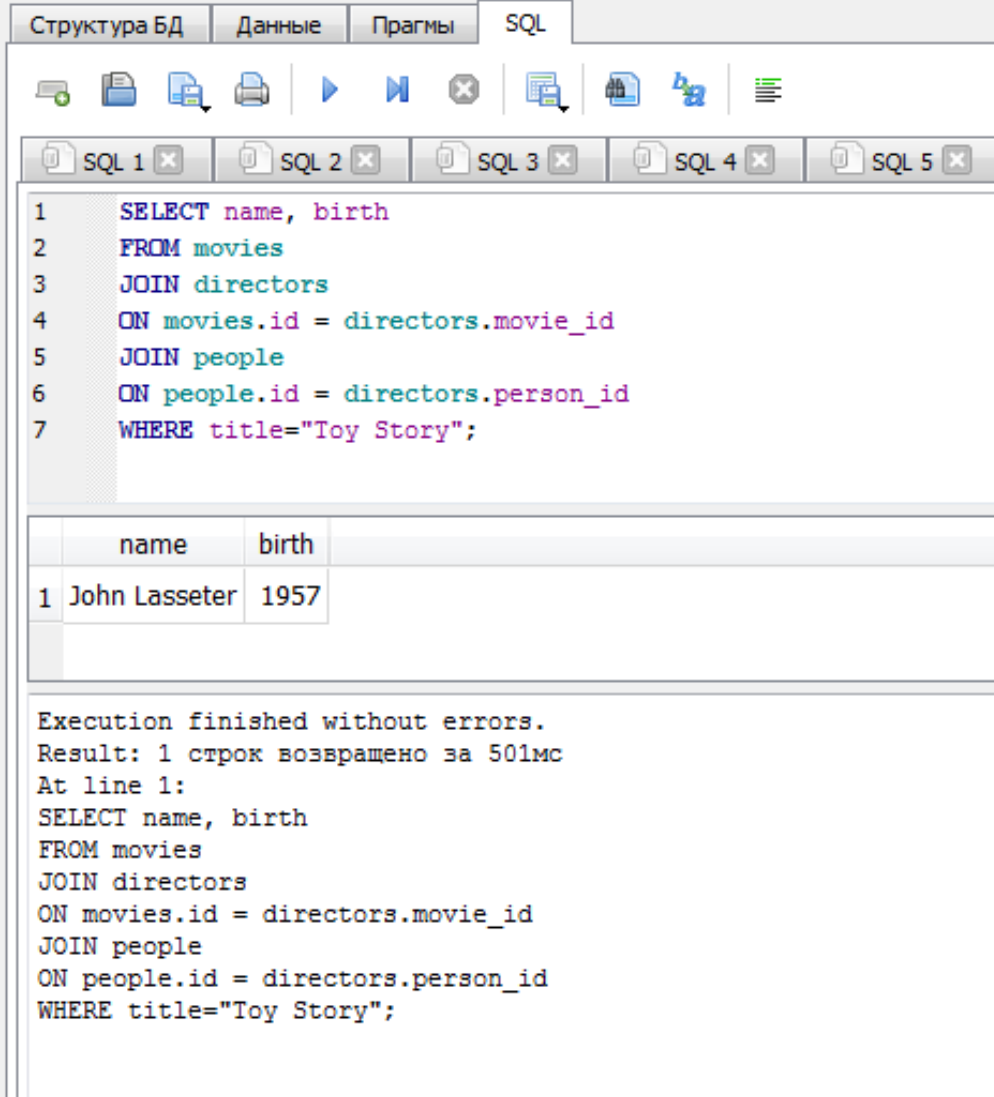
The screenshot shows a database management interface with tabs for 'Структура БД', 'Данные', 'Прагмы', and 'SQL'. The 'SQL' tab is active, displaying a query in a text editor. Below the editor, the results of the query are shown in a table with one column named 'name'. The results list four actors: Tom Hanks, Tim Allen, Don Rickles, and Jim Varney. At the bottom, a status bar indicates that the execution finished without errors and returned 4 rows in 1409ms.

```
1 SELECT name
2 FROM movies
3 JOIN stars
4 ON movies.id = stars.movie_id
5 JOIN people
6 ON people.id = stars.person_id
7 WHERE title="Toy Story";
```

	name
1	Tom Hanks
2	Tim Allen
3	Don Rickles
4	Jim Varney

Execution finished without errors.
Result: 4 строк возвращено за 1409мс
At line 1:
SELECT name
FROM movies
JOIN stars
ON movies.id = stars.movie_id
JOIN people
ON people.id = stars.person_id
WHERE title="Toy Story";

Output the name and year of birth of the director of Toy Story



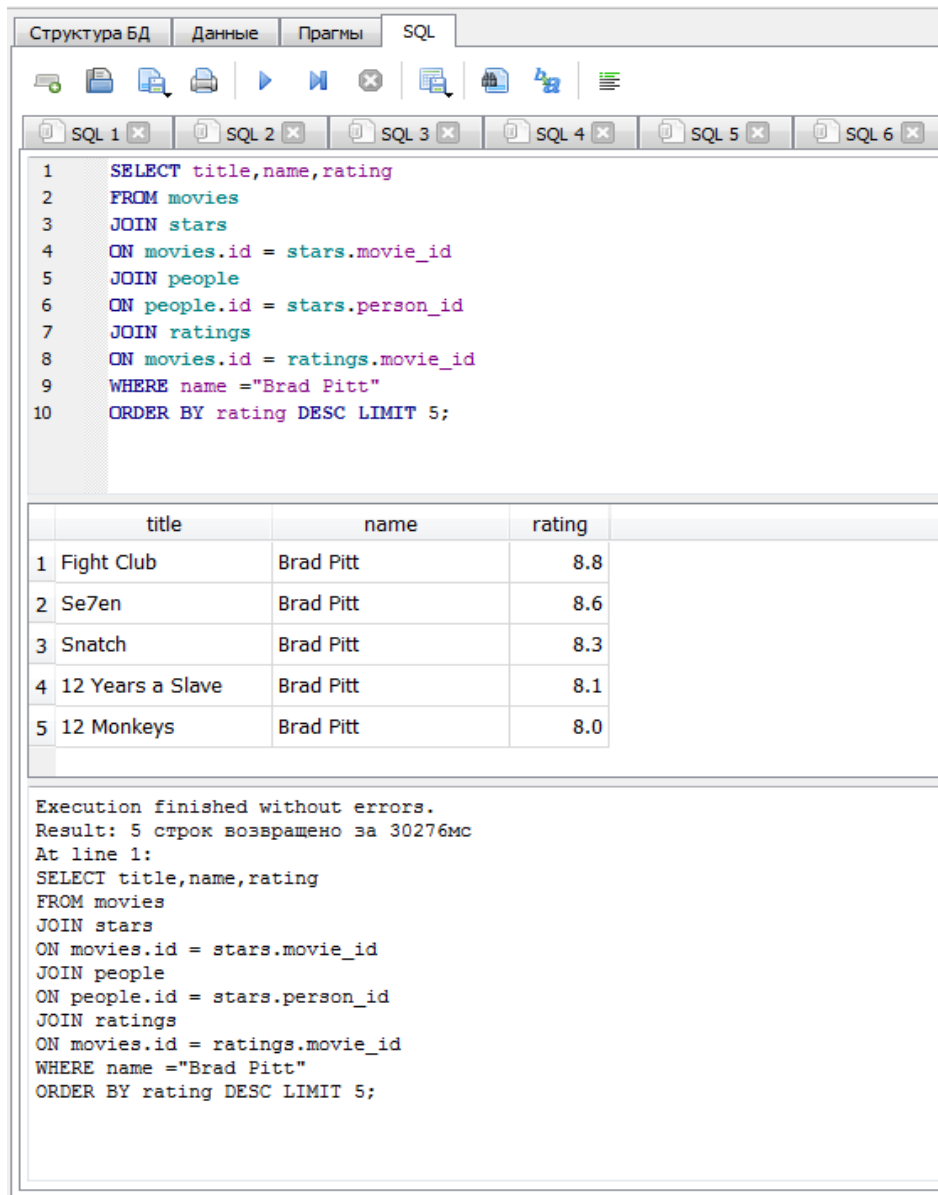
The screenshot shows a database management interface with tabs for 'Структура БД', 'Данные', 'Прагмы', and 'SQL'. The 'SQL' tab is active, displaying a query in a text editor. Below the editor is a table with the results of the query. At the bottom, a status bar indicates that the execution finished without errors and shows the SQL query again.

```
1 SELECT name, birth
2 FROM movies
3 JOIN directors
4 ON movies.id = directors.movie_id
5 JOIN people
6 ON people.id = directors.person_id
7 WHERE title="Toy Story";
```

	name	birth
1	John Lasseter	1957

Execution finished without errors.
Result: 1 строк возвращено за 501мс
At line 1:
SELECT name, birth
FROM movies
JOIN directors
ON movies.id = directors.movie_id
JOIN people
ON people.id = directors.person_id
WHERE title="Toy Story";

Display 5 movies with the highest rating where Brad Pitt played



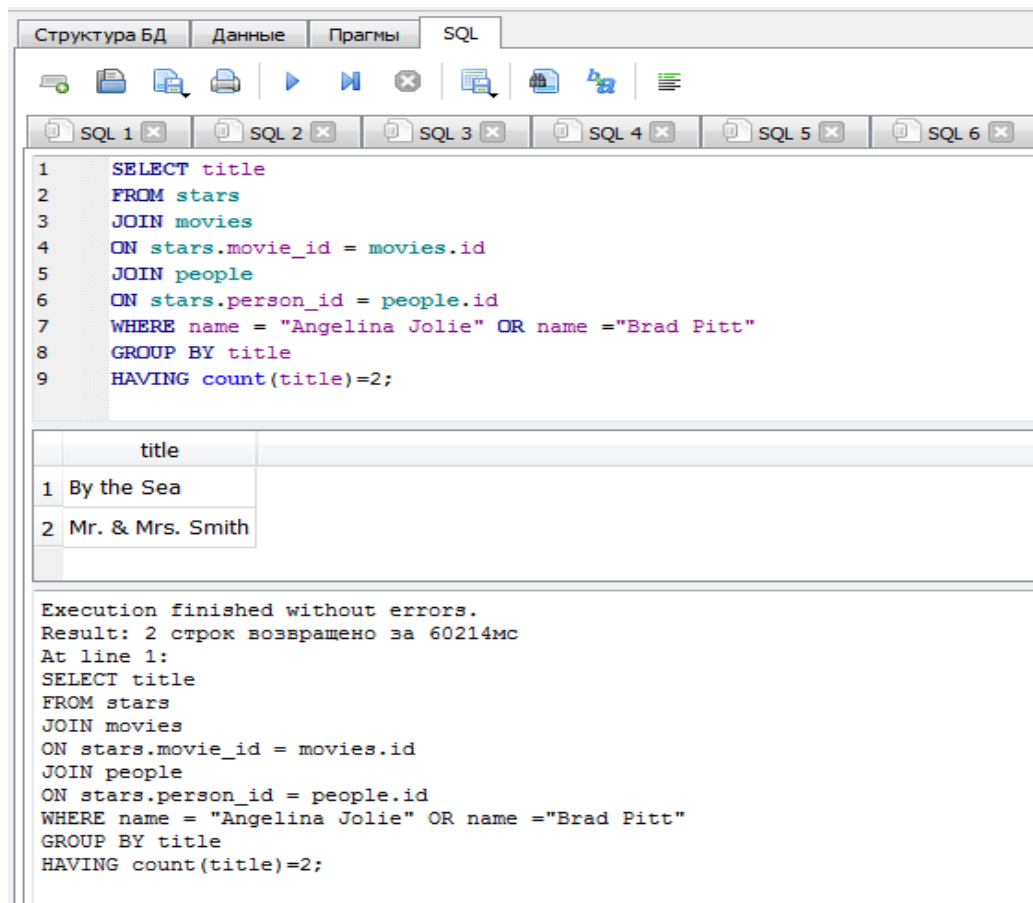
The screenshot shows a database management interface with tabs for 'Структура БД', 'Данные', 'Прагмы', and 'SQL'. The 'SQL' tab is active, displaying a query in a text editor. Below the editor, a table shows the results of the query. At the bottom, a status bar indicates that the execution finished without errors and provides details about the result set.

```
1  SELECT title,name,rating
2  FROM movies
3  JOIN stars
4  ON movies.id = stars.movie_id
5  JOIN people
6  ON people.id = stars.person_id
7  JOIN ratings
8  ON movies.id = ratings.movie_id
9  WHERE name ="Brad Pitt"
10 ORDER BY rating DESC LIMIT 5;
```

	title	name	rating
1	Fight Club	Brad Pitt	8.8
2	Se7en	Brad Pitt	8.6
3	Snatch	Brad Pitt	8.3
4	12 Years a Slave	Brad Pitt	8.1
5	12 Monkeys	Brad Pitt	8.0

Execution finished without errors.
Result: 5 строк возвращено за 30276мс
At line 1:
SELECT title,name,rating
FROM movies
JOIN stars
ON movies.id = stars.movie_id
JOIN people
ON people.id = stars.person_id
JOIN ratings
ON movies.id = ratings.movie_id
WHERE name ="Brad Pitt"
ORDER BY rating DESC LIMIT 5;

Display all movies where Brad Pitt and Angelina Jolie played together



The screenshot shows a SQL IDE interface with a menu bar (Структура БД, Данные, Прагмы, SQL), a toolbar, and a tabbed window for SQL queries. The active query is SQL 1, containing the following SQL code:

```
1 SELECT title
2 FROM stars
3 JOIN movies
4 ON stars.movie_id = movies.id
5 JOIN people
6 ON stars.person_id = people.id
7 WHERE name = "Angelina Jolie" OR name ="Brad Pitt"
8 GROUP BY title
9 HAVING count(title)=2;
```

Below the query editor, the results are displayed in a table with two columns: an index and the movie title.

	title
1	By the Sea
2	Mr. & Mrs. Smith

At the bottom of the IDE, the execution status is shown:

Execution finished without errors.
Result: 2 строк возвращено за 60214мс
At line 1:
SELECT title
FROM stars
JOIN movies
ON stars.movie_id = movies.id
JOIN people
ON stars.person_id = people.id
WHERE name = "Angelina Jolie" OR name ="Brad Pitt"
GROUP BY title
HAVING count(title)=2;