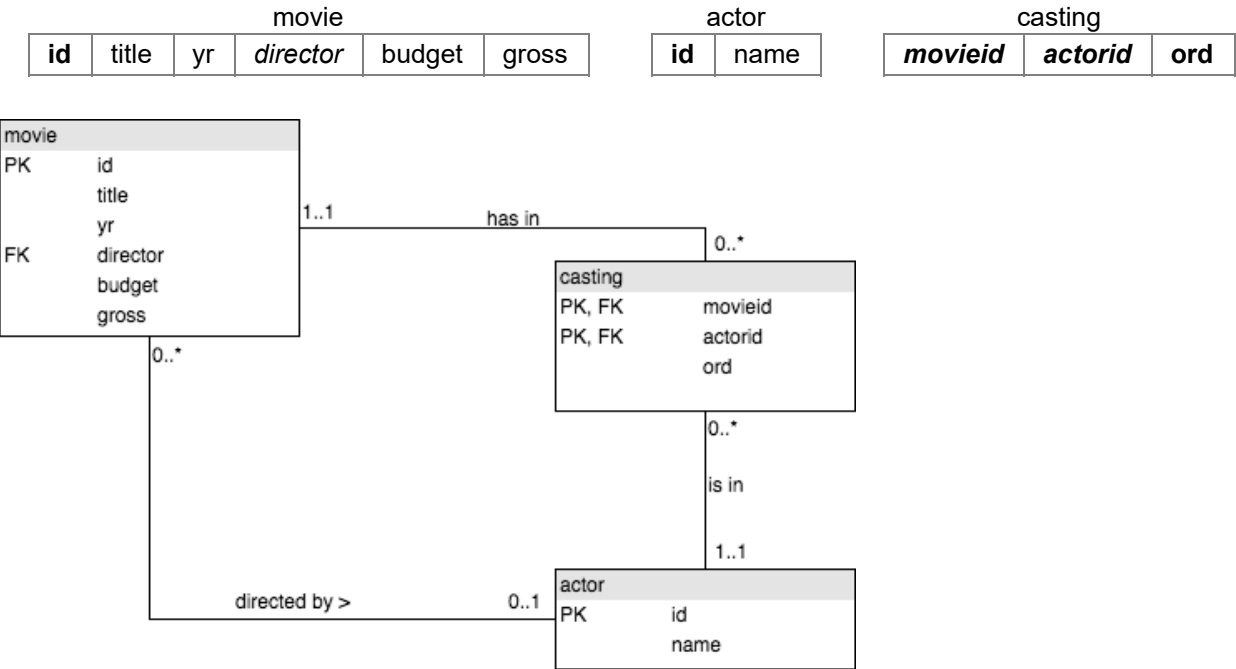


# More JOIN operations

Language: English • 日本語 • 中文

This tutorial introduces the notion of a join. The database consists of three tables movie , actor and casting .



[More details about the database.](#)

## Summary

## Contents

- 1962 movies
- When was Citizen Kane released?
- Star Trek movies
- id for actor Glenn Close
- id for Casablanca
- Cast list for Casablanca
- Alien cast list
- Harrison Ford movies
- Harrison Ford as a supporting actor
- Lead actors in 1962 movies
- Busy years for John Travolta
- Lead actor in Julie Andrews movies
- Actors with 30 leading roles

## 1962 movies

### 1.

List the films where the yr is 1962 [Show id, title]

```
SELECT id, title  
FROM movie  
WHERE yr=1962
```

Submit SQLRestore default

result

## When was Citizen Kane released?

---

# 2.

Give year of 'Citizen Kane'.

```
select yr  
from movie  
where title = 'Citizen Kane'
```

Submit SQLRestore default

result

## Star Trek movies

---

### 3.

List all of the Star Trek movies, include the **id**, **title** and **yr** (all of these movies include the words Star Trek in the title). Order results by year.

```
select id, title, yr
from movie
where title LIKE '%Star Trek%'
```

Submit SQLRestore default

result

## id for actor Glenn Close

---

### 4.

What **id** number does the actor 'Glenn Close' have?

```
select id
from actor
where name = 'Glenn Close'
```

Submit SQLRestore default

result

## id for Casablanca

---

# 5.

What is the **id** of the film 'Casablanca'

```
select id
from movie
where title = 'Casablanca'
```

Submit SQLRestore default

result

[Get to the point](#)

## Cast list for Casablanca

---

# 6.

Obtain the cast list for 'Casablanca'.

*what is a cast list?*

Use **movieid=11768**, (or whatever value you got from the previous question)

```
SELECT name
FROM casting JOIN actor ON casting.actorid=actor.id
WHERE movieid = 11768
```

Submit SQL

Restore default

result

## Alien cast list

---

# 7.

Obtain the cast list for the film 'Alien'

```
SELECT name
FROM casting
      JOIN actor ON casting.actorid=actor.id
      JOIN movie ON casting.movieid=movie.id
WHERE movie.title='Alien'
```

Submit SQL

Restore default

result

## Harrison Ford movies

---

# 8.

List the films in which 'Harrison Ford' has appeared

```
SELECT movie.title
FROM casting
      JOIN actor ON casting.actorid=actor.id
      JOIN movie ON casting.movieid=movie.id
WHERE actor.name='Harrison Ford'
```

Submit SQL

Restore default

result

## Harrison Ford as a supporting actor

---

# 9.

List the films where 'Harrison Ford' has appeared - but not in the starring role. [Note: the **ord** field of casting gives the position of the actor. If ord=1 then this actor is in the starring role]

```
SELECT movie.title
FROM casting
      JOIN actor ON casting.actorid=actor.id
      JOIN movie ON casting.movieid=movie.id
WHERE actor.name='Harrison Ford' AND casting.ord > 1
```

Submit SQLRestore default

result

## Lead actors in 1962 movies

---

# 10.

List the films together with the leading star for all 1962 films.

```
SELECT movie.title, actor.name
FROM casting
      JOIN actor ON casting.actorid=actor.id
      JOIN movie ON casting.movieid=movie.id
WHERE movie.yr = 1962 AND casting.ord = 1
```

Submit SQLRestore default

result

---

Harder Questions

---

## Busy years for John Travolta

---

# 11.

Which were the busiest years for 'John Travolta', show the year and the number of movies he made each year for any year in which he made more than 2 movies.



```
SELECT yr,COUNT(title)
FROM movie
      JOIN casting ON movie.id=movieid
      JOIN actor   ON actorid=actor.id
WHERE name='John Travolta'
GROUP BY yr
HAVING COUNT(title)=(SELECT MAX(c) FROM
(SELECT yr,COUNT(title) AS c FROM
      movie JOIN casting ON movie.id=movieid
      JOIN actor   ON actorid=actor.id
      WHERE name='John Travolta'
      GROUP BY yr) AS t
)
```

Submit SQLRestore default

result

## Lead actor in Julie Andrews movies

---

# 12.

List the film title and the leading actor for all of the films 'Julie Andrews' played in.

*Did you get "Little Miss Marker twice"?*

```
SELECT movie.title, actor.name
FROM casting
      JOIN movie ON movie.id=movieid
      JOIN actor ON actorid=actor.id
WHERE casting.movieid IN (SELECT casting.movieid
                        FROM casting JOIN actor ON casting.actorid=actor.id
                        WHERE actor.id IN (SELECT id
                                          FROM actor
                                          WHERE name='Julie Andrews' ) )
AND casting.ord = 1
```

Submit SQLRestore default

mj12



result

## Actors with 30 leading roles

---

# 13.

Obtain a list, in alphabetical order, of actors who've had at least 30 starring roles.

```
SELECT actor.name
FROM casting
      JOIN actor ON casting.actorid=actor.id
WHERE casting.ord = 1
GROUP BY actor.name
HAVING COUNT(casting.ord) >= 30
```

[Submit SQL](#)[Restore default](#)

result

---

## 14.

List the films released in the year 1978 ordered by the number of actors in the cast, then by title.

```
SELECT title, COUNT(actorid)
FROM movie
      JOIN casting ON movie.id=movieid
WHERE yr = 1978
GROUP BY title, yr
ORDER BY COUNT(actorid) DESC, title
```

[Submit SQL](#)[Restore default](#)

result

# 15.

List all the people who have worked with 'Art Garfunkel'.

```
SELECT DISTINCT name
FROM casting
      JOIN actor ON casting.actorid=actor.id
WHERE movieid = ANY (SELECT DISTINCT movieid
                    FROM casting
                      JOIN actor ON casting.actorid=actor.id
                      WHERE name = 'Art Garfunkel') AND name <> 'Art Garfunkel'
```

Submit SQL

Restore default

result

Clear your results

## JOIN Quiz 2

That is definitely enough. Students should, under no circumstances look at the next tutorial, concerning outer joins. ([http://sqlzoo.net/w/index.php/Using\\_Null](http://sqlzoo.net/w/index.php/Using_Null))

Retrieved from "[https://sqlzoo.net/w/index.php?title=More\\_JOIN\\_operations&oldid=39514](https://sqlzoo.net/w/index.php?title=More_JOIN_operations&oldid=39514)"

This page was last edited on 29 May 2018, at 03:29.

