

## Joins Part 1

### 1. Films directed by John Woo?

```
select m.*
from movies m
      join credits c
        on c.movieid = m.movieid
      join people p
        on p.peopleid = c.peopleid
where c.credited_as = 'D'
      and p.first_name = 'John'
      and p.surname = 'Woo';
```

### 2. List the cast of the American "Treasure Island" 1950 film (first name, surname, and display "Director" instead of "D" and "Actor" instead of "A")

```
select p.first_name,
       p.surname,
       case c.credited_as
         when 'A' then 'Actor'
         when 'D' then 'Director' end as credited_as
from movies m
      join credits c
        on c.movieid = m.movieid
      join people p
        on p.peopleid = c.peopleid
where m.title = 'Treasure Island' -- Beware, for strict comparison SQLite
      and m.country = 'us'         -- IS case sensitive
      and m.year_released = 1950;
```

### 3. Marilyn Monroe appeared in how many 1952 films?

```
select count(*)
from (select m.movieid
      from people p
        join credits c
          on c.peopleid = p.peopleid
        join movies m
          on m.movieid = c.movieid
      where c.credited_as = 'A'
            and p.first_name = 'Marilyn'
            and p.surname = 'Monroe'
            and m.year_released = 1952) a ;
```

#### 4. Directors who have directed films in more than one country?

```
select p.first_name, p.surname
from (select c.peopleid
      from credits c
           join movies m
              on m.movieid = c.movieid
      where c.credited_as = 'D'
      group by c.peopleid
      having count(distinct m.country) > 1) x
      join people p
         on p.peopleid = x.peopleid;
```

#### 5. Display the title, year and country code for films from 2010 or later for which the name of the director is missing from the database.

```
select m.title, m.year_released, m.country
from movies m
      left join credits c
         on c.movieid = m.movieid
           and c.credited_as = 'D'
where m.year_released >= 2010
      and c.movieid is null; -- Any column from credits works
```

#### 6. What are the title, country and year of release of remakes in the database ? (films for which an earlier film with the same title exists)

```
select distinct r.title, r.country, r.year_released
from movies m
      join movies r -- remakes
         on r.title = m.title and r.year_released > m.year_released;
```

**distinct** is required when there is more than one remake (otherwise the third film would appear as remake of the first one and as remake of the second one).

#### 7. Longest film directed by a woman?

```
select distinct m.title,
               m.country,
               m.year_released,
               m.runtime
from movies m
      join credits c on c.movieid = m.movieid
```

```

        join people p on p.peopleid = c.peopleid
where p.gender = 'F' and c.credited_as = 'D'
    and m.runtime =
        (select max(m.runtime) -- NULLs will be ignored
         from movies m
          join credits c
            on c.movieid = m.movieid
          join people p
            on p.peopleid = c.peopleid
         where p.gender = 'F'
           and c.credited_as = 'D');
-- or
with dfdw as -- Detail of Films Directed by Women
    (select m.title, m.country, m.year_released, m.runtime
     from (select distinct c.movieid
           from credits c
            join people p
              on p.peopleid = c.peopleid
             where p.gender = 'F' and c.credited_as = 'D') fdw
          join movies m on m.movieid = fdw.movieid
     where coalesce(runtime, 0) > 0 -- Ignore anything for which we have
    )
select dfdw.*
from dfdw
where dfdw.runtime =
    -- no data or zero
    (select max(runtime) from dfdw);

```

## Set operators

1. List all year and "Events" (films released time, people births time, people deaths time) that occurred between 1930 and 1935

```

SELECT m.year_released AS year,
       m.title || ' (' || c.country_name || ') was released' AS event
FROM movies m
JOIN
    countries c ON c.country_code = m.country
WHERE m.year_released BETWEEN 1930 AND 1935
UNION ALL
SELECT born,
       trim(coalesce(first_name, '') || ' ' || surname || ' was born')
FROM people
WHERE born BETWEEN 1930 AND 1935

```

```

UNION ALL
SELECT died,
       trim(coalesce(first_name, '') || ' ' || surname || ' died')
FROM people
WHERE died BETWEEN 1930 AND 1935
ORDER BY year;

```

2. Same as question1, pushed into a subquery to add a sort key

```

SELECT year, event
FROM (SELECT m.year_released AS year,
             m.title || ' (' || c.country_name || ') was released' AS event,
             m.title AS sort_key
FROM movies m
JOIN
countries c ON c.country_code = m.country
WHERE m.year_released BETWEEN 1930 AND 1935
UNION ALL
SELECT born,
       trim(coalesce(first_name, '') || ' ' || surname || ' was born'),
       surname AS sort_key
FROM people
WHERE born BETWEEN 1930 AND 1935
UNION ALL
SELECT died,
       trim(coalesce(first_name, '') || ' ' || surname || ' died'),
       surname AS sort_key
FROM people
WHERE died BETWEEN 1930 AND 1935) x
ORDER BY year, sort_key;

```

3. Films where Qi Shu played without Ge You. Illustrates that "except" isn't really necessary

```

SELECT m.title,
       m.country,
       m.year_released
FROM (
    SELECT c.movieid
    FROM credits c
    JOIN
    people p ON p.peopleid = c.peopleid
    WHERE p.first_name = 'Qi' AND
          p.surname = 'Shu' AND
          c.credited_as = 'A'

```

```

        EXCEPT
        SELECT c.movieid
            FROM credits c
            JOIN
            people p ON p.peopleid = c.peopleid
        WHERE p.first_name = 'You' AND
            p.surname = 'Ge' AND
            c.credited_as = 'A'
    )
    x
    JOIN
    movies m ON m.movieid = x.movieid
ORDER BY m.year_released;

-- or

SELECT m.title, m.country, m.year_released
    FROM (
        SELECT c.movieid
            FROM credits c
            JOIN
            people p ON p.peopleid = c.peopleid
        WHERE p.first_name = 'Qi' AND
            p.surname = 'Shu' AND
            c.credited_as = 'A' AND
            c.movieid NOT IN (
                SELECT c.movieid
                    FROM credits c
                    JOIN
                    people p ON p.peopleid = c.peopleid
                WHERE p.first_name = 'You' AND
                    p.surname = 'Ge' AND
                    c.credited_as = 'A'
            )
    )
    x
    JOIN
    movies m ON m.movieid = x.movieid
ORDER BY m.year_released;

-- or

SELECT m.title, m.country, m.year_released
    FROM (
        SELECT c.movieid
            FROM credits c

```

```

        JOIN
        people p ON p.peopleid = c.peopleid
WHERE p.first_name = 'Qi' AND
      p.surname = 'Shu' AND
      c.credited_as = 'A' AND
      NOT EXISTS (
        SELECT NULL
        FROM credits c2
        JOIN
        people p2 ON p2.peopleid = c2.peopleid
        WHERE p2.first_name = 'You' AND
              p2.surname = 'Ge' AND
              c2.credited_as = 'A' AND
              p2.peopleid = c2.peopleid AND
              c2.movieid = c.movieid
      )
    )x
  JOIN
  movies m ON m.movieid = x.movieid
ORDER BY m.year_released;

-- or

SELECT m.title, m.country, m.year_released
FROM (
  SELECT c.movieid
  FROM credits c
  JOIN
  people p ON p.peopleid = c.peopleid
  LEFT OUTER JOIN
  (
    SELECT c2.movieid
    FROM credits c2
    JOIN
    people p2 ON p2.peopleid = c2.peopleid
    WHERE p2.first_name = 'You' AND
          p2.surname = 'Ge' AND
          c2.credited_as = 'A' AND
          p2.peopleid = c2.peopleid
  )
  y ON y.movieid = c.movieid
WHERE p.first_name = 'Qi' AND
      p.surname = 'Shu' AND
      c.credited_as = 'A' AND

```

```

        y.movieid IS NULL
    )
    x
  JOIN
    movies m ON m.movieid = x.movieid
  ORDER BY m.year_released;
-- or

SELECT m.title, m.country, m.year_released
  FROM (
    SELECT c.movieid
      FROM credits c
      JOIN
        people p ON p.peopleid = c.peopleid
     WHERE (p.first_name = 'Shu' AND
            p.surname = 'Qi') OR
            (p.first_name = 'You' AND
            p.surname = 'Ge') AND
            c.credited_as = 'A'
    GROUP BY c.movieid
    HAVING count( * ) = 1 AND
           min(surname) = 'Qi'
  )x
-- If the min is Qi, there is no Ge
  JOIN
    movies m ON m.movieid = x.movieid
  ORDER BY m.year_released;

```