queries_union_intersect

```sql
SELECT name,
    nationality,
    year_of_birth
FROM actors
WHERE nationality <> 'USA'
UNION
SELECT name,
    nationality,
    year_of_birth
FROM actors
WHERE year_of_birth > 1990;
```

Explanation:

- This SQL query retrieves data from an actors table. It uses a UNION operation to combine the results of two SELECT statements. The first SELECT statement returns all actors who are not from the USA. The second SELECT statement returns all actors born after 1990. The UNION operator combines these results, removing duplicate rows. The final result is a list of actors' names, nationalities, and birth years satisfying either condition.

```sql
SELECT name,
    nationality,
    year_of_birth
FROM actors
WHERE nationality <> 'USA'
INTERSECT
SELECT name,
    nationality,
    year_of_birth
FROM actors
WHERE year_of_birth > 1990;
```

Explanation:

- This SQL query retrieves the name, nationality, and year_of_birth of actors who meet two conditions: they are not from the USA, and they were born after 1990. It uses the INTERSECT operator to find the common rows between the two SELECT statements. The first SELECT finds non-USA actors, and the second finds actors born after 1990. The intersection gives only those actors satisfying both conditions.

```sql
SELECT movie_id, AVG(rating) -- Select the IDs of all movies with average rating higher than 9
FROM renting
GROUP BY movie_id
HAVING AVG(rating) > 9;
```

Explanation:

- This SQL query calculates the average rating for each movie from a table named renting and then filters the results to show only those movies with an average rating greater than 9. It uses GROUP BY to group rows by movie_id and HAVING to filter the grouped results based on the average rating.

```sql
SELECT movie_id
FROM movies
WHERE genre = 'Drama'
INTERSECT  -- Select the IDs of all dramas with average rating higher than 9
SELECT movie_id
FROM renting
GROUP BY movie_id
HAVING AVG(rating)>9;
```

Explanation:

- This SQL query finds the IDs of drama movies that have an average rating higher than 9. It does this by using the INTERSECT operator to combine the results of two separate SELECT statements. The first SELECT statement identifies all movies with the genre 'Drama', while the second SELECT statement finds movies with an average rating greater than 9, using aggregate function AVG() and HAVING clause for filtering. The intersection gives only the movie IDs that satisfy both conditions.

```sql
SELECT *
FROM movies
WHERE movie_id IN -- Select all movies of genre drama with average rating higher than 9
  (SELECT movie_id
   FROM movies
   WHERE genre = 'Drama'
   INTERSECT
   SELECT movie_id
   FROM renting
   GROUP BY movie_id
   HAVING AVG(rating)>9);
```

Explanation:

- This SQL query retrieves all information from the movies table for movies that meet two conditions: they are of the 'Drama' genre, and their average rating (as determined from the renting table) is greater than 9. It uses an INTERSECT operation

to find movie_ids that satisfy both conditions. The subquery using HAVING AVG(rating) > 9 filters for movies with an average rating above 9, while the WHERE genre = 'Drama' subquery selects only drama movies. The intersection combines these results.