

Assignment 6

Joins and Union

Consider the Country table and Persons table that you created earlier.

Query and Result

```
4 • create table country (id int primary key,  
5   country_name varchar(50),  
6   population varchar(50),  
7   area_sqkm varchar(50));  
8 • desc country;  
9 • insert into country(id, country_name, population, area_sqkm)  
10 values(1, 'India', '90000000', '32 lakhs'),  
11        (2, 'China', '90000000', '97 lakhs'),  
12        (3, 'USA', '80000000', '93 lakhs'),  
13        (4, 'UK', '70000000', '2 lakhs'),  
14        (5, 'Canada', '60000000', '99 lakhs'),  
15        (6, 'Australia', '50000000', '50 lakhs'),  
16        (7, 'Netherlands', '40000000', '32 lakhs'),  
17        (8, 'Sweden', '30000000', '4 lakhs'),  
18        (9, 'Austria', '20000000', '3 lakhs'),  
19        (10, 'Switzerland', '10000000', '2 lakhs');  
20 • select*from country;
```

Result Grid

Filter Rows:

Edit:

Export/Import:

| | id | country_name | population | area_sqkm |
|---|------|--------------|------------|-----------|
| ▶ | 1 | India | 90000000 | 32 lakhs |
| | 2 | China | 90000000 | 97 lakhs |
| | 3 | USA | 80000000 | 93 lakhs |
| | 4 | UK | 70000000 | 2 lakhs |
| | 5 | Canada | 60000000 | 99 lakhs |
| | 6 | Australia | 50000000 | 50 lakhs |
| | 7 | Netherlands | 40000000 | 32 lakhs |
| | 8 | Sweden | 30000000 | 4 lakhs |
| | 9 | Austria | 20000000 | 3 lakhs |
| | 10 | Switzerland | 10000000 | 2 lakhs |
| ★ | NULL | NULL | NULL | NULL |




```
22 • create table persons (id int primary key,  
23     first_name varchar(50),  
24     last_name varchar(50),  
25     population varchar(50),  
26     rating decimal(10,2),  
27     country_id int,  
28     country_name varchar(50));  
29 • desc persons;  
30 • insert into persons(id, first_name, last_name, population,rating,country_id,country_name)  
31 values(1, 'Lisha', 'Thomas', '90000000', 1.71, 1, 'India'),  
32 (2, 'Chaang', 'Yaang', '90000000', 2.52, 2, 'China'),  
33 (3, 'Thomas', 'Cook', '80000000', 3.23, 3, 'USA'),  
34 (4, 'Diana', 'Xavier', '90000000', 1.84, 1, 'India'),  
35 (5, 'Freddy', 'Dainz', '60000000', 5.35, 5, 'Canada'),  
36 (6, 'Hari', 'Sharma', '90000000', 1.96, 1, 'India'),  
37 (7, 'william', 'Blake', '40000000', 7.47, 7, 'Netherlands'),  
38 (8, 'Right', 'Thomas', '30000000',8.18, 8, 'Sweden'),  
39 (9, 'Cindrella', 'John', '20000000', 9.69, 9, 'Austria'),  
40 (10, 'Irene', 'Dizooza', '10000000', 10.10, 10, 'Iceland');  
41 • select*from persons;
```

[illegible]

(1) Perform inner join, Left join, and Right join on the tables.




Query and Result for (a) Inner join

```
44 • select c.id, c.country_name, c.population, p.country_name, p.population
45      from country as c inner join persons as p on c.id=p.id;
46
```

| Result Grid  Filter Rows: <input type="text"/> Export:  Wrap Cell Content:  | | | | | |
|--|----|--------------|------------|--------------|------------|
| | id | country_name | population | country_name | population |
| ▶ | 1 | India | 90000000 | India | 90000000 |
| | 2 | China | 90000000 | China | 90000000 |
| | 3 | USA | 80000000 | USA | 80000000 |
| | 4 | UK | 70000000 | India | 90000000 |
| | 5 | Canada | 60000000 | Canada | 60000000 |
| | 6 | Australia | 50000000 | India | 90000000 |
| | 7 | Netherlands | 40000000 | Netherlands | 40000000 |
| | 8 | Sweden | 30000000 | Sweden | 30000000 |
| | 9 | Austria | 20000000 | Austria | 20000000 |
| | 10 | Switzerland | 10000000 | Iceland | 10000000 |

Query and Result for (a) Left join

```
47 • select c.id, c.country_name, c.population, p.country_name, p.population
48      from country as c left join persons as p on c.id=p.id;
49
```

| Result Grid  Filter Rows: <input type="text"/> Export:  Wrap Cell Content:  | | | | | |
|--|----|--------------|------------|--------------|------------|
| | id | country_name | population | country_name | population |
| ▶ | 1 | India | 90000000 | India | 90000000 |
| | 2 | China | 90000000 | China | 90000000 |
| | 3 | USA | 80000000 | USA | 80000000 |
| | 4 | UK | 70000000 | India | 90000000 |
| | 5 | Canada | 60000000 | Canada | 60000000 |
| | 6 | Australia | 50000000 | India | 90000000 |
| | 7 | Netherlands | 40000000 | Netherlands | 40000000 |
| | 8 | Sweden | 30000000 | Sweden | 30000000 |
| | 9 | Austria | 20000000 | Austria | 20000000 |
| | 10 | Switzerland | 10000000 | Iceland | 10000000 |

Query and Result for (a) Right join

```
50 • select c.id, c.country_name, c.population, p.country_name, p.population
51 from country as c right join persons as p on c.id=p.id;
52
```

| Result Grid Filter Rows: <input type="text"/> Export: Wrap Cell Content: | | | | | |
|--|----|--------------|------------|--------------|------------|
| | id | country_name | population | country_name | population |
| ▶ | 1 | India | 90000000 | India | 90000000 |
| | 2 | China | 90000000 | China | 90000000 |
| | 3 | USA | 80000000 | USA | 80000000 |
| | 4 | UK | 70000000 | India | 90000000 |
| | 5 | Canada | 60000000 | Canada | 60000000 |
| | 6 | Australia | 50000000 | India | 90000000 |
| | 7 | Netherlands | 40000000 | Netherlands | 40000000 |
| | 8 | Sweden | 30000000 | Sweden | 30000000 |
| | 9 | Austria | 20000000 | Austria | 20000000 |
| | 10 | Switzerland | 10000000 | Iceland | 10000000 |

(2)List all distinct country names from both the Country and Persons tables.

Query and Result




```
53 #List all distinct country names from both the Country and Persons tables
54 • select distinct country_name from country
55 union
56 select distinct country_name from persons;
57
```

| Result Grid Filter Rows: <input type="text"/> Export: Wrap Cell Content: | |
|--|--------------|
| | country_name |
| ▶ | India |
| | China |
| | USA |
| | UK |
| | Canada |
| | Australia |
| | Netherlands |
| | Sweden |
| | Austria |
| | Switzerland |
| | Iceland |

(3)List all country names from both the Country and Persons tables, including duplicates.

Query and Result

```
58 #List all country names from both the Country and Persons tables, including duplicates
59 • select country_name from country
60 union all
61 select country_name from persons;
```

Result Grid   Filter Rows: Export:  Wrap Cell

| | country_name |
|---|--------------|
| ▶ | India |
| | China |
| | USA |
| | UK |
| | Canada |
| | Australia |
| | Netherlands |
| | Sweden |
| | Austria |
| | Switzerland |
| | India |
| | China |
| | USA |
| | India |
| | Canada |
| | India |
| | Netherlands |
| | Sweden |
| | Austria |
| ▶ | Iceland |

(4)Round the ratings of all persons to the nearest integer in the Persons table.



Query and Result

```


63      #Round the ratings of all persons to the nearest integer in the Persons table
64 •    select rating, round(rating) as round_rating from persons ;
65

```


Result Grid



Filter Rows:

Export:



Wrap Cell Content:



| | rating | round_rating |
|---|--------|--------------|
| ▶ | 1.71 | 2 |
| | 2.52 | 3 |
| | 3.23 | 3 |
| | 1.84 | 2 |
| | 5.35 | 5 |
| | 1.96 | 2 |
| | 7.47 | 7 |
| | 8.18 | 8 |
| | 9.69 | 10 |
| | 10.10 | 10 |