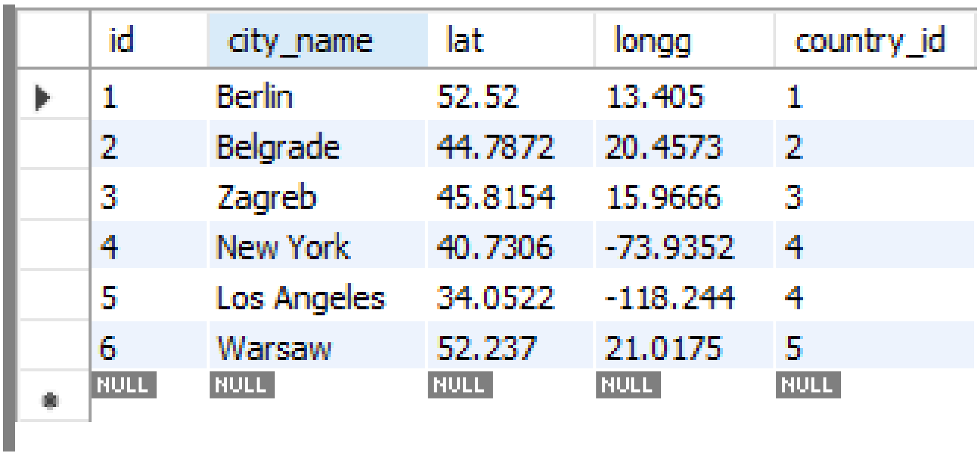
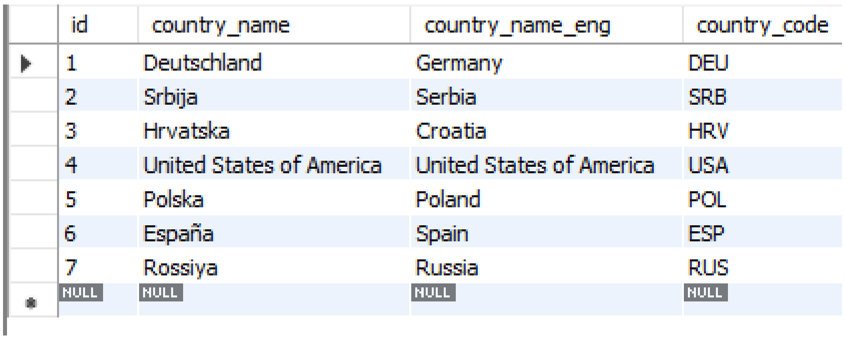
SQL Assessment

```
create table city (
    id int primary key,
    city_name varchar(50),
    lat float,
    long float,
    country_id int
);

insert into city (id, city_name, lat, long, country_id) values
(1, 'Berlin', 52.520008, 13.404954, 1),
(2, 'Belgrade', 44.787197, 20.457273, 2),
(3, 'Zagreb', 45.815399, 15.966568, 3),
(4, 'New York', 40.730610, -73.935242, 4),
(5, 'Los Angeles', 34.052235, -118.243683, 4),
(6, 'Warsaw', 52.237049, 21.017532, 5);

Select * from city;
```



id	city_name	lat	longg	country_id
1	Berlin	52.52	13.405	1
2	Belgrade	44.7872	20.4573	2
3	Zagreb	45.8154	15.9666	3
4	New York	40.7306	-73.9352	4
5	Los Angeles	34.0522	-118.244	4
6	Warsaw	52.237	21.0175	5
NULL	NULL	NULL	NULL	NULL

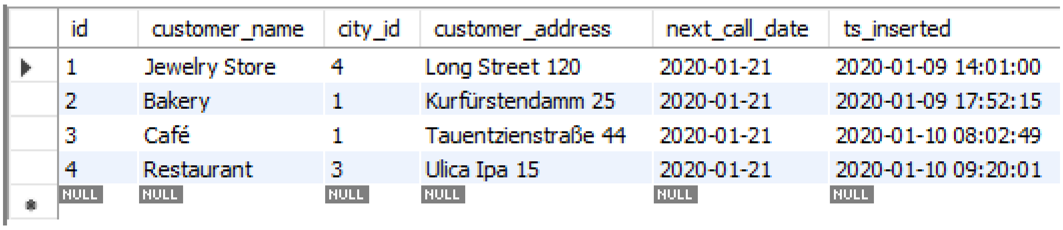
```
create table customer (
    id int primary key,
    customer_name varchar(50),
    city_id int,
    customer_address varchar(100),
    next_call_date date,
    ts_inserted timestamp
);
```



id	country_name	country_name_eng	country_code
1	Deutschland	Germany	DEU
2	Srbija	Serbia	SRB
3	Hrvatska	Croatia	HRV
4	United States of America	United States of America	USA
5	Polska	Poland	POL
6	España	Spain	ESP
7	Rossiya	Russia	RUS
NULL	NULL	NULL	NULL

```
insert into customer (id, customer_name, city_id, customer_address,
next_call_date, ts_inserted) values
(1, 'Jewelry Store', 4, 'Long Street 120', '2020-01-21', '2020-01-09 14:01:00'),
(2, 'Bakery', 1, 'Kurfürstendamm 25', '2020-01-21', '2020-01-09 17:52:15'),
(3, 'Café', 1, 'Tauentzienstraße 44', '2020-01-21', '2020-01-10 08:02:49'),
(4, 'Restaurant', 3, 'Ulica Ipa 15', '2020-01-21', '2020-01-10 09:20:01');
```

```
create table country (
    id int primary key,
    country_name varchar(50),
    country_name_eng varchar(50),
    country_code varchar(10)
);
```



	id	customer_name	city_id	customer_address	next_call_date	ts_inserted
▶	1	Jewelry Store	4	Long Street 120	2020-01-21	2020-01-09 14:01:00
	2	Bakery	1	Kurfürstendamm 25	2020-01-21	2020-01-09 17:52:15
	3	Café	1	Tauentzienstraße 44	2020-01-21	2020-01-10 08:02:49
	4	Restaurant	3	Ulica Ipa 15	2020-01-21	2020-01-10 09:20:01
*	NULL	NULL	NULL	NULL	NULL	NULL

```
insert into country (id, country_name, country_name_eng, country_code) values
(1, 'Deutschland', 'Germany', 'DEU'),
(2, 'Srbija', 'Serbia', 'SRB'),
(3, 'Hrvatska', 'Croatia', 'HRV'),
(4, 'United States of America', 'United States of America', 'USA'),
(5, 'Polska', 'Poland', 'POL'),
(6, 'España', 'Spain', 'ESP'),
(7, 'Rossiya', 'Russia', 'RUS');
```

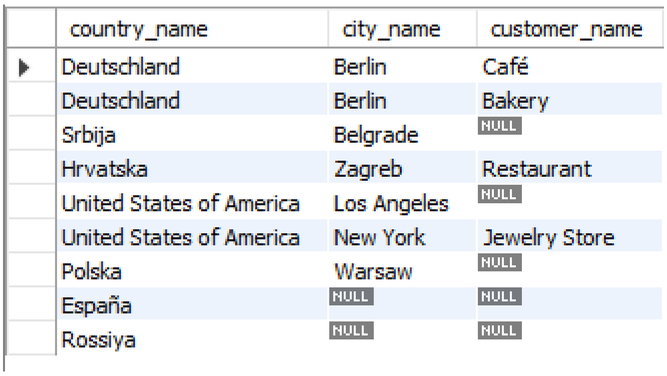
Task 1: (LEFT JOINs)

Objective: List all countries and customers (via cities) related to those countries.
Even if a country has no city or a city has no customers, it should still appear.

Logic:

- country LEFT JOIN city ON country.id = city.country_id
- LEFT JOIN customer ON city.id = customer.city_id

```
select country.country_name, city.city_name, customer.customer_name

from country

left join city on country.id = city.country_id

left join customer on city.id = customer.city_id;
```



country_name	city_name	customer_name
Deutschland	Berlin	Café
Deutschland	Berlin	Bakery
Srbija	Belgrade	NULL
Hrvatska	Zagreb	Restaurant
United States of America	Los Angeles	NULL
United States of America	New York	Jewelry Store
Polska	Warsaw	NULL
España	NULL	NULL
Rossiya	NULL	NULL

The query will:

- Show all countries (even Spain & Russia with no cities),
- Show cities if they exist,
- Show customers if linked to cities.

Task 2: (LEFT JOIN + INNER JOIN)

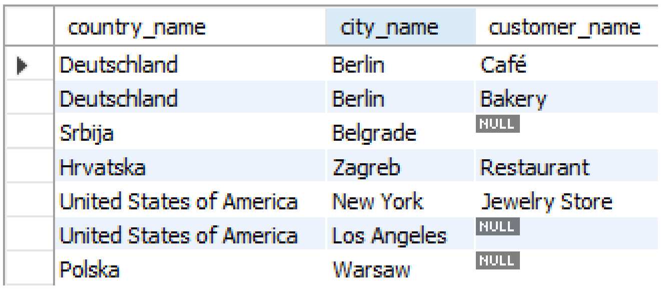
Objective:

- Only include **countries that have at least one city** (exclude Spain & Russia).
- For such countries, list all customers (even if a city has no customers).

Logic:

- Use an **INNER JOIN** between `country` and `city` to only get countries that have cities.
- Use a **LEFT JOIN** from `city` to `customer` to retain cities with or without customers.

```
select country.country_name, city.city_name, customer.customer_name

from city

inner join country on country.id = city.country_id

left join customer on customer.city_id = city.id;
```



country_name	city_name	customer_name
Deutschland	Berlin	Café
Deutschland	Berlin	Bakery
Srbija	Belgrade	NULL
Hrvatska	Zagreb	Restaurant
United States of America	New York	Jewelry Store
United States of America	Los Angeles	NULL
Polska	Warsaw	NULL

This query will:

- Exclude countries not referenced by any city,
- Return all customers under cities of those countries,
- Still include cities with **no customers**.