DATABASE ASSESMENT

- **Write SQL query to solve the problem given below**

Consider three table named as city, customer and country

The city table is given below :

id	city_name	lat	long	country_id
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

The Customer table :

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 14:01:20.000
2	Bakery	1	Kurfürstendamm 25	2020-02-21	2020-01-09 17:52:15.000
3	Café	1	Tauentzienstraße 44	2020-01-21	2020-01-10 08:02:49.000
4	Restaurant	3	Ulica lipa 15	2020-01-21	2020-01-10 09:20:21.000

The Country table :

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

DATABASE ASSESMENT

- ```
  CREATE TABLE city
  ( id INT PRIMARY KEY AUTO_INCREMENT, city_name VARCHAR(50) NOT NULL,
  lat float NOT NULL, long_ float NOT NULL, country_id INT NOT NULL ),f
  oreign key(country_code)references-country(id);
  ```
- ```
  CREATE TABLE custormer_1(id int PRIMARY key,C_name varchar(30),city_i
  d int,C_address varchar(40),next_call_date date,tl_inserted timestamp
  , FOREIGN key(city_id) REFERENCES city(id));
  ```
- ```
  CREATE TABLE country(id int primary key,country_name varchar(40),c_na
  me_eng varchar(30),c_code varchar(4) );
  ```

TASK 1

Task : 1 (join multiple tables using left join)

List all Countries and customers related to these countries.

For each country displaying its name in English, the name of the city customer is located in as well as the name of the customer.

Return even countries without related cities and customers.

List all Countries and customers related to these countries.

```
SELECT * FROM country left JOIN city ON country.id=city.country_id left joi
n custormer_1 ON city.id=custormer_1.city_id;
```

id	country_name	c_name_eng	c_code	id	city_name	lat	long_	country_id	id	C_name	city_id	C_address	next_call_date	tl_inserted	country_code
4	United states of america	America	USA	4	New york	40.7306	-73.9352	4	1	Jewelry store	4	Long street 120	0000-00-00	2024-01-11 11:30:16	NULL
1	Deutscland	Germany	DEU	1	berlin	52.52	13.405	1	2	Bakery	1	Kufustendamn 25	0000-00-00	2024-01-11 11:30:16	NULL
1	Deutscland	Germany	DEU	1	berlin	52.52	13.405	1	3	Cafe	1	Tauentzienstase 44	0000-00-00	2024-01-11 11:30:16	NULL
3	Hrvatska	Croatia	HRV	3	Zagreb	45.8154	15.9666	3	4	Restaurent	3	Uica lipa 15	0000-00-00	2024-01-11 11:30:16	NULL
2	Sribis	Serbia	SRB	2	Belgrade	44.7872	20.4573	2	NULL	NULL	NULL	NULL	NULL	NULL	NULL
4	United states of america	America	USA	5	Los Angelus	34.0522	-118.244	4	NULL	NULL	NULL	NULL	NULL	NULL	NULL
5	Polska	Poland	POL	6	Warsaw	52.237	21.0176	5	NULL	NULL	NULL	NULL	NULL	NULL	NULL
6	Espania	Spain	ESP	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
```

DATABASE ASSESMENT

Task : 2 (join multiple tables using both left and inner join)

Return the list of all countries that have pairs(exclude countries which are not referenced by any city). For such pairs return all customers.

Return even pairs of not having a single customer

```
SELECT * FROM country left JOIN city ON country.id=city.country_id INNER jo
in custormer_1 ON city.id=custormer_1.city_id;
```

id	country_name	c_name_eng	c_code	id	city_name	lat	long_	country_id	id	C_name	city_id	C_address	next_call_date
4	United states of america	America	USA	4	New york	40.7306	-73.9352	4	1	Jewelry store	4	Long street 120	0000-00-00
1	Deutscland	Germany	DEU	1	berlin	52.52	13.405	1	2	Bakery	1	Kufustendamn 25	0000-00-00
1	Deutscland	Germany	DEU	1	berlin	52.52	13.405	1	3	Cafe	1	Tauentzienstase 44	0000-00-00
3	Hrvatska	Croatia	HRV	3	Zagreb	45.8154	15.9666	3	4	Restaurent	3	Uica lipa 15	0000-00-00