Write SQL query to solve the problem given below.

- **Create database:-**

```
CREATE  DATABASE  mydatabase;
```

- **Create country table:-**

```
CREATE TABLE contry(id int PRIMARY KEY AUTO_INCREMENT,
        country_name varchar(20),
        country_name_eng varchar(20),
        country_code varchar(20));
```

- **Inserte data in country table:-**

```
INSERT INTO contry VALUES(1,"Deutschland","Germany","DEU"),
        (2,"Srbija","Aerbia","SRB"),
        (3,"Hrvatska","Croatia","HRV"),
        (4,"United States of America","United state of america","USA"),
        (5,"Polska","Poland","POL"),
        (6,"Esparia","Spain","ESP"),
        (7,"Rossiya","Russia","RUS");
```

- **Create city table:-**

```
CREATE TABLE  city(id int PRIMARY KEY,
                   city_name varchar(20),
                   lat decimal(10,6),
                   lon decimal(10,6),
                   country_id int,
                   FOREIGN KEY(country_id) REFERENCES country(id) );
```

- **Insert data in city table:-**

```
INSERT INTO city VALUES (1,"berlin",52.520008,13.404954,1),
(2,"berlin",44.787197,20.457273,2),
(3,"zageb",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);
```

- **Create customer table:-**

```
CREATE TABLE customer(id int  PRIMARY KEY,
        customer_name varchar(20),
        city_id int,
        FOREIGN KEY(city_id) REFERENCES city(id),
        customer_address varchar(20),
        next_call_date date,
        ts_inserted datetime);
```

- **Inserte data in customer table:-**

```
INSERT INTO customer VALUES (1,"jewelry store",4,"long street 120",'2020-01-21','2020-01-09 14:01:20.000'),
(2,"bakery",1,"kufurstendamm 25",'2020-02-21','2020-01-09 17:52:15.000'),
(3,"cafe",1,"tauentzienstrabe 44",'2020-01-21','2020-01-10 08:02:49.000'),
(4,"resturant",3,"uice lipa 15",'2020-01-21','2020-01-10 09:20:21.000');
```

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.

```
SELECT country.country_name_eng, city.city_name, customer.customer_name
FROM country
LEFT JOIN city ON city.country_id = country.id
LEFT JOIN customer ON customer.city_id = city.id;
```

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 country.country_name_eng, city.city_name, customer.customer_name
FROM country
INNER JOIN city ON city.country_id = country.id
LEFT JOIN customer ON customer.city_id = city.id;
```