

Assessment

To create the database and then create all three tables (city, customer, country) inside it, here are the exact MySQL commands you can run.

Step 1: Create the Database

```
CREATE DATABASE world_data;
```

Step 2: Use the Database

```
USE world_data;
```

Step 3: Create COUNTRY Table

- SQL – Create Table

```
CREATE TABLE country (
    id INT PRIMARY KEY,
    country_name VARCHAR(100),
    country_name_eng VARCHAR(100),
    country_code VARCHAR(10)
);
```

- SQL – Insert Data

```
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');
```

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

Step 4: Create CITY Table

- SQL – Create Table

```
CREATE TABLE city (
    id INT PRIMARY KEY,
    city_name VARCHAR(100),
    lat DECIMAL(10,6),
    longitude DECIMAL(10,6),
    country_id INT,
    FOREIGN KEY (country_id) REFERENCES country(id)
);
```

- SQL -Insert Data

```
INSERT INTO city (id, city_name, lat, longitude, 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);
```

	id	city_name	lat	longitude	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

Step 5: Create CUSTOMER Table

- SQL – Create Table

```
CREATE TABLE customer (
    id INT PRIMARY KEY,
    customer_name VARCHAR(100),
    city_id INT,
    customer_address VARCHAR(150),
    next_call_date DATE,
    ts_inserted DATETIME,
    FOREIGN KEY (city_id) REFERENCES city(id)
);
```

- SQL – Insert Data

```
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 14:01:20'),
(2, 'Bakery', 1, 'Kurfürstendamm 25', '2020-02-21', '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 09:20:21');
```

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

❖ Task 1: LEFT JOIN (Return all countries + related cities + customers)

Requirement:

- Show all countries, including those without cities (Spain, Russia).
- Show country_name_eng, city_name (if exists), customer_name (if exists).
- LEFT JOIN must be used so missing cities/customers still appear.

➤ SQL Query (Task 1)

```
SELECT
    c.country_name_eng,
    ci.city_name,
    cu.customer_name
FROM country c
LEFT JOIN city ci
    ON c.id = ci.country_id
LEFT JOIN customer cu
    ON ci.id = cu.city_id
ORDER BY c.country_name_eng;
```

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** world_data
- Table:** customer (selected)
- Query Result:**
 - Showing rows 0 - 8 (9 total, Query took 0.0095 seconds.) [country_name_eng: CROATIA... - UNITED STATES OF AMERICA...]
 - SQL Query: SELECT c.country_name_eng, ci.city_name, cu.customer_name FROM country c LEFT JOIN city ci ON c.id = ci.country_id LEFT JOIN customer cu ON ci.id = cu.city_id ORDER BY c.country_name_eng;
 - Buttons: Profiling, Edit inline, Explain SQL, Create PHP code, Refresh.
 - Extra options: Show all, Number of rows: 25, Filter rows, Search this table.
 - Data Table:

country_name_eng	city_name	customer_name
Croatia	Zagreb	Restaurant
Germany	Berlin	Cafe
Germany	Berlin	Bakery
Poland	Warsaw	NULL
Russia	NULL	NULL
Serbia	Belgrade	NULL
Spain	NULL	NULL
United States of America	Los Angeles	NULL
United States of America	New York	Jewelry Store
 - Buttons: Show all, Number of rows: 25, Filter rows, Search this table.

❖ **Task 2: LEFT + INNER JOIN (Return only countries that have cities, and show their customers)**

Requirement:

- Return ONLY countries that have cities (so exclude Spain & Russia).
- For each country-city pair, return all customers, even if a city has zero customers.
- Use INNER JOIN + LEFT JOIN.

➤ **SQL Query (Task 2)**

```

SELECT
    c.country_name_eng,
    ci.city_name,
    cu.customer_name
FROM country c
INNER JOIN city ci
    ON c.id = ci.country_id      -- only countries that have cities
LEFT JOIN customer cu
    ON ci.id = cu.city_id       -- return cities even without customers
ORDER BY c.country_name_eng, ci.city_name;

```

phpMyAdmin

Server: 127.0.0.1 > Database: world_data

Recent Favorites

New employee_information_table information_schema mysql performance_schema phpmyadmin school_db test world_data New city country customer

Show query box

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 6 (7 total, Query took 0.0084 seconds.) [country_name_eng] CROATIA... - UNITED STATES OF AMERICA... [city_name] ZAGREB... - NEW YORK...

SELECT c.country_name_eng, ci.city_name, cu.customer_name FROM country c INNER JOIN city ci ON c.id = ci.country_id LEFT JOIN customer cu ON ci.id = cu.city_id ORDER BY c.country_name_eng, ci.city_name;

Profiling | Edit inline | Explain SQL | Create PHP code | Refresh

Show all Number of rows: 25 Filter rows Search this table

Extra options

country_name_eng	city_name	customer_name
Croatia	Zagreb	Restaurant
Germany	Berlin	Café
Germany	Berlin	Bakery
Poland	Warsaw	NULL
Serbia	Belgrade	NULL
United States of America	Los Angeles	NULL
United States of America	New York	Jewelry Store

Show all Number of rows: 25 Filter rows Search this table