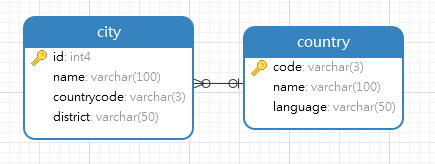
# Tugas Tema : Pengetahuan dasar SQL

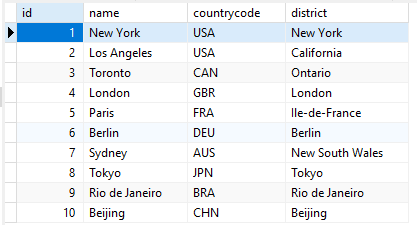
Petunjuk Pengerjaan dan Peraturan

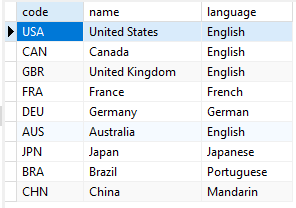
# Terdapat 5 buah soal yang harus dikerjakan

* Pengerjaan jawaban dapat dikerjakan di Word ataupun secara langsung pada local laptop

*Schema*



*Tabel City*

*Tabel Country*

# Lakukan instalasi rdbms pada local laptop baik postgresql atau mysql dan buat database baru dengan nama ‘world'

# Login ke my sql

# mysql -u root -p

# buat database world

# CREATE DATABASE world;

# USE world; // use world di gunakan untuk pindah ke database world

# 

* 1. a. Buatlah table ‘country’ dengan kriteria berikut :
     + Membuat kolom "Code" dengan tipe data VARCHAR(3), yang berarti kode negara akan memiliki maksimal panjang 3 karakter. PRIMARY KEY menandakan bahwa kolom ini adalah kunci utama (primary key) tabel, yang berarti setiap nilai di kolom ini harus unik dan tidak boleh null (kosong).
     + Membuat kolom "Name" dengan tipe data VARCHAR(100), yang berarti nama negara dapat memiliki maksimal panjang 100 karakter.
     + Membuat kolom "Language" dengan tipe data VARCHAR(50), yang berarti bahasa negara dapat memiliki maksimal panjang 50 karakter.

Buat table country

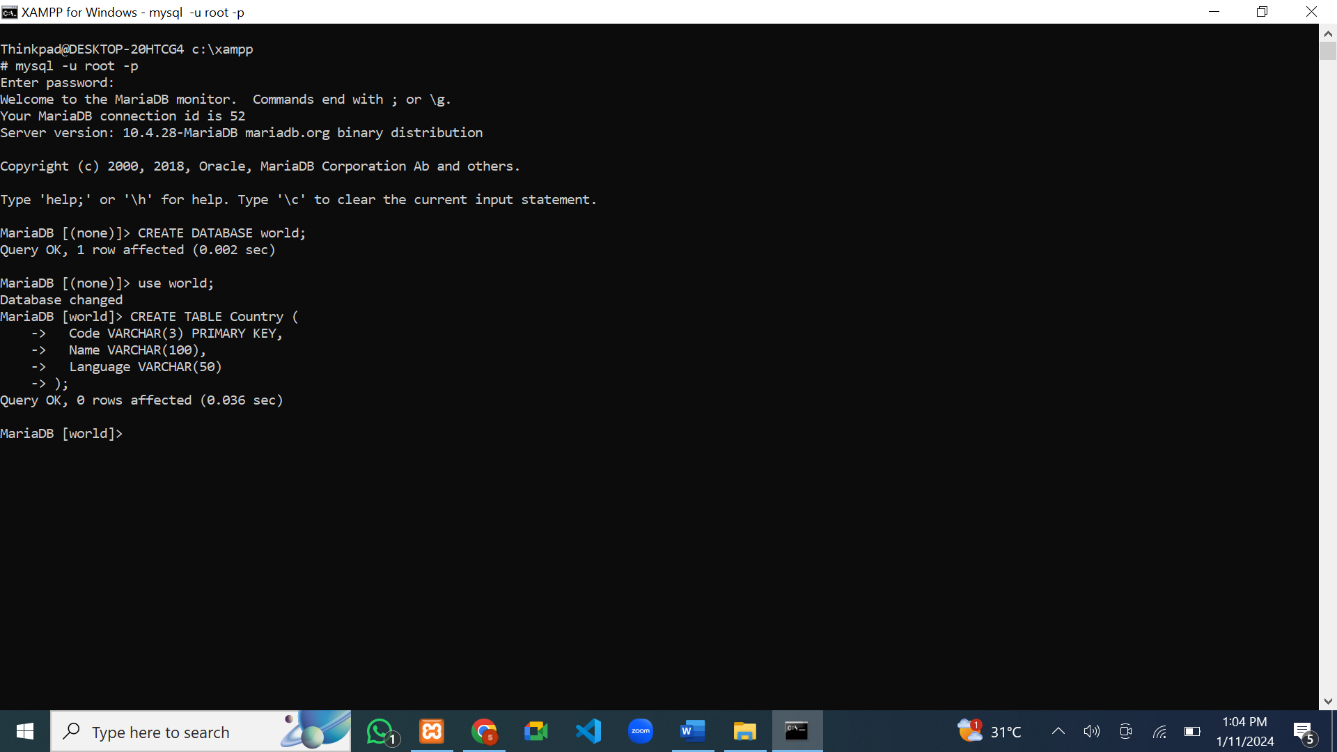
CREATE TABLE Country (

Code VARCHAR(3) PRIMARY KEY,

Name VARCHAR(100),

Language VARCHAR(50)

);



# b. Buatlah table ‘city’ dengan kriteria berikut :

* + - Membuat kolom "ID" dengan tipe data SERIAL, yang akan secara otomatis diisi dengan nilai yang unik dan berturut-turut. PRIMARY KEY menandakan bahwa kolom ini adalah kunci utama tabel.
    - Membuat kolom "Name" dengan tipe data VARCHAR(100), yang berarti nama kota dapat memiliki maksimal panjang 100 karakter.
    - Membuat kolom "CountryCode" dengan tipe data VARCHAR(3), yang merujuk pada kolom "Code" dari tabel "Country". Ini adalah contoh penggunaan FOREIGN KEY (REFERENCES) yang menghubungkan dua tabel. Hal ini memastikan bahwa setiap nilai di kolom "CountryCode" harus sesuai dengan nilai yang ada di kolom "Code" dari tabel "Country".
    - Membuat kolom "District" dengan tipe data VARCHAR(50), yang berarti distrik kota dapat memiliki maksimal panjang 50 karakter.

Buat table city

CREATE TABLE City (

ID INT AUTO\_INCREMENT PRIMARY KEY,

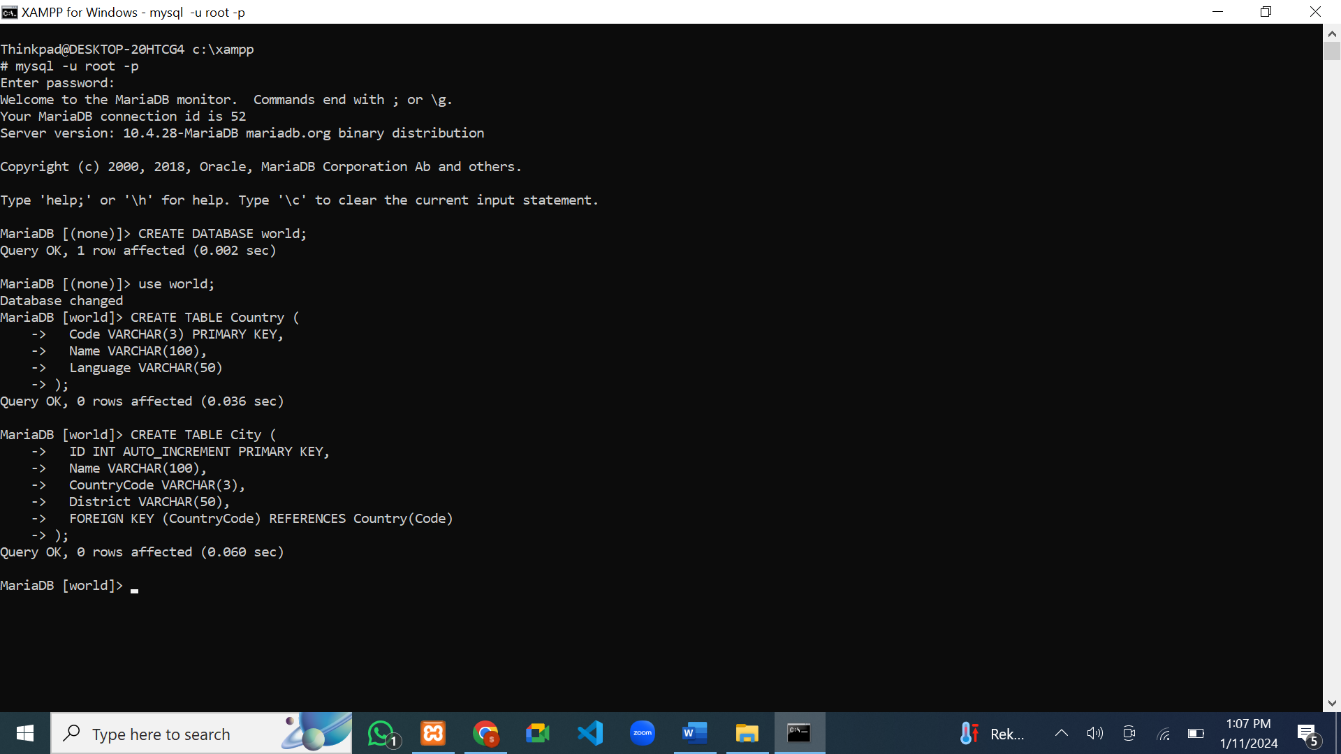
Name VARCHAR(100),

CountryCode VARCHAR(3),

District VARCHAR(50),

FOREIGN KEY (CountryCode) REFERENCES Country(Code)

);



# Mengisi table yang telah dibuat sebelumnya dengan data berikut :

-- Tabel Country

INSERT INTO Country (Code, Name, Language) VALUES ('USA', 'United States', 'English'),

('CAN', 'Canada', 'English'),

('GBR', 'United Kingdom', 'English'),

('FRA', 'France', 'French'),

('DEU', 'Germany', 'German'),

('AUS', 'Australia', 'English'),

('JPN', 'Japan', 'Japanese'),

('BRA', 'Brazil', 'Portuguese'),

('CHN', 'China', 'Mandarin');

// menambah data di atas ke dalam table country

INSERT INTO country (Code, Name, Language) VALUES

('USA', 'United States', 'English'),

('CAN', 'Canada', 'English'),

('GBR', 'United Kingdom', 'English'),

('FRA', 'France', 'French'),

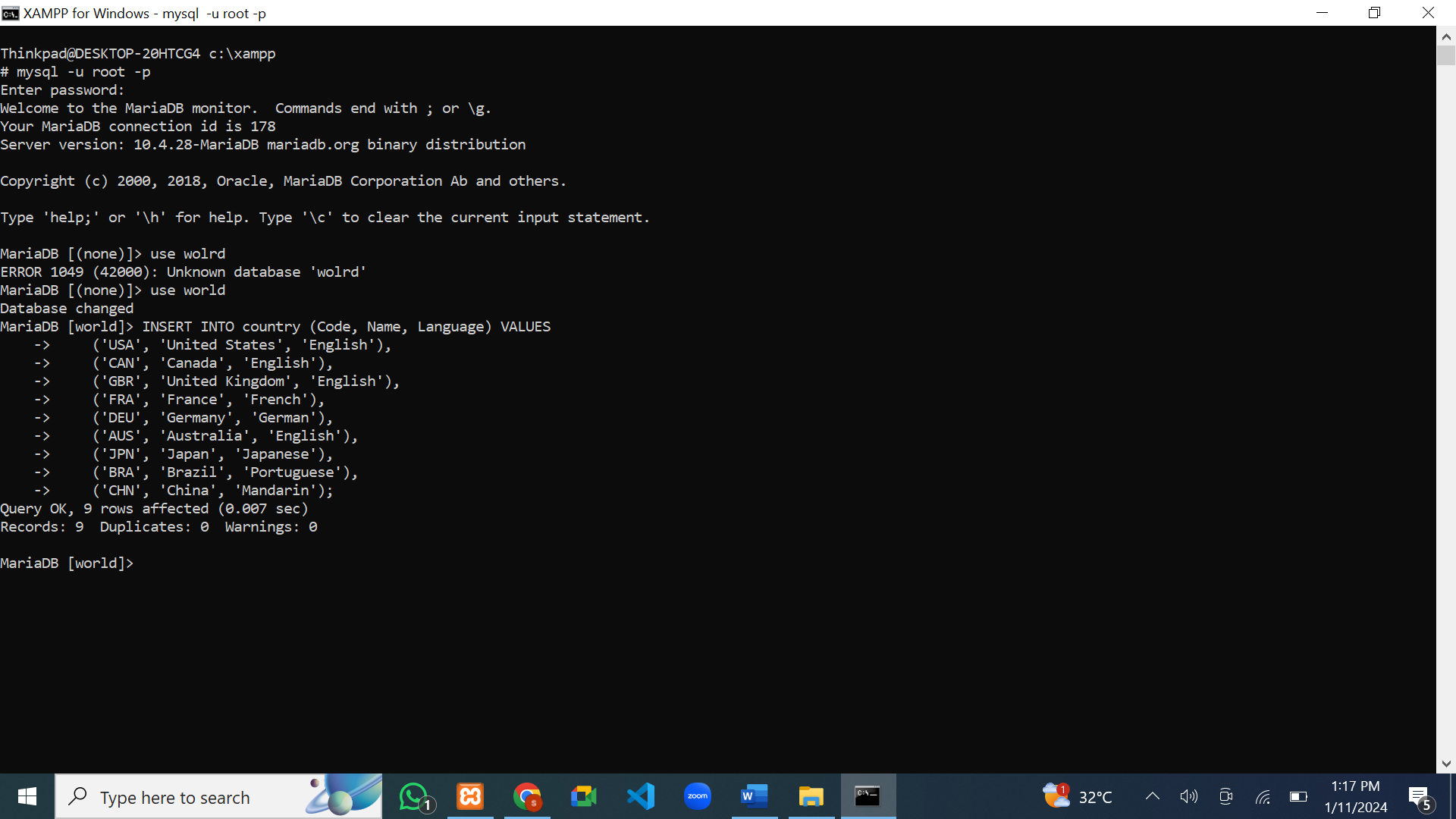
('DEU', 'Germany', 'German'),

('AUS', 'Australia', 'English'),

('JPN', 'Japan', 'Japanese'),

('BRA', 'Brazil', 'Portuguese'),

('CHN', 'China', 'Mandarin');



-- Tabel City

INSERT INTO City (Name, CountryCode, District) VALUES ('New York', 'USA', 'New York'),

('Los Angeles', 'USA', 'California'),

('Toronto', 'CAN', 'Ontario'),

('London', 'GBR', 'London'),

('Paris', 'FRA', 'Ile-de-France'),

('Berlin', 'DEU', 'Berlin'),

('Sydney', 'AUS', 'New South Wales'),

('Tokyo', 'JPN', 'Tokyo'),

('Rio de Janeiro', 'BRA', 'Rio de Janeiro'), ('Beijing', 'CHN', 'Beijing');

-- Tabel City

INSERT INTO city (Name, CountryCode, District) VALUES

('New York', 'USA', 'New York'),

('Los Angeles', 'USA', 'California'),

('Toronto', 'CAN', 'Ontario'),

('London', 'GBR', 'London'),

('Paris', 'FRA', 'Ile-de-France'),

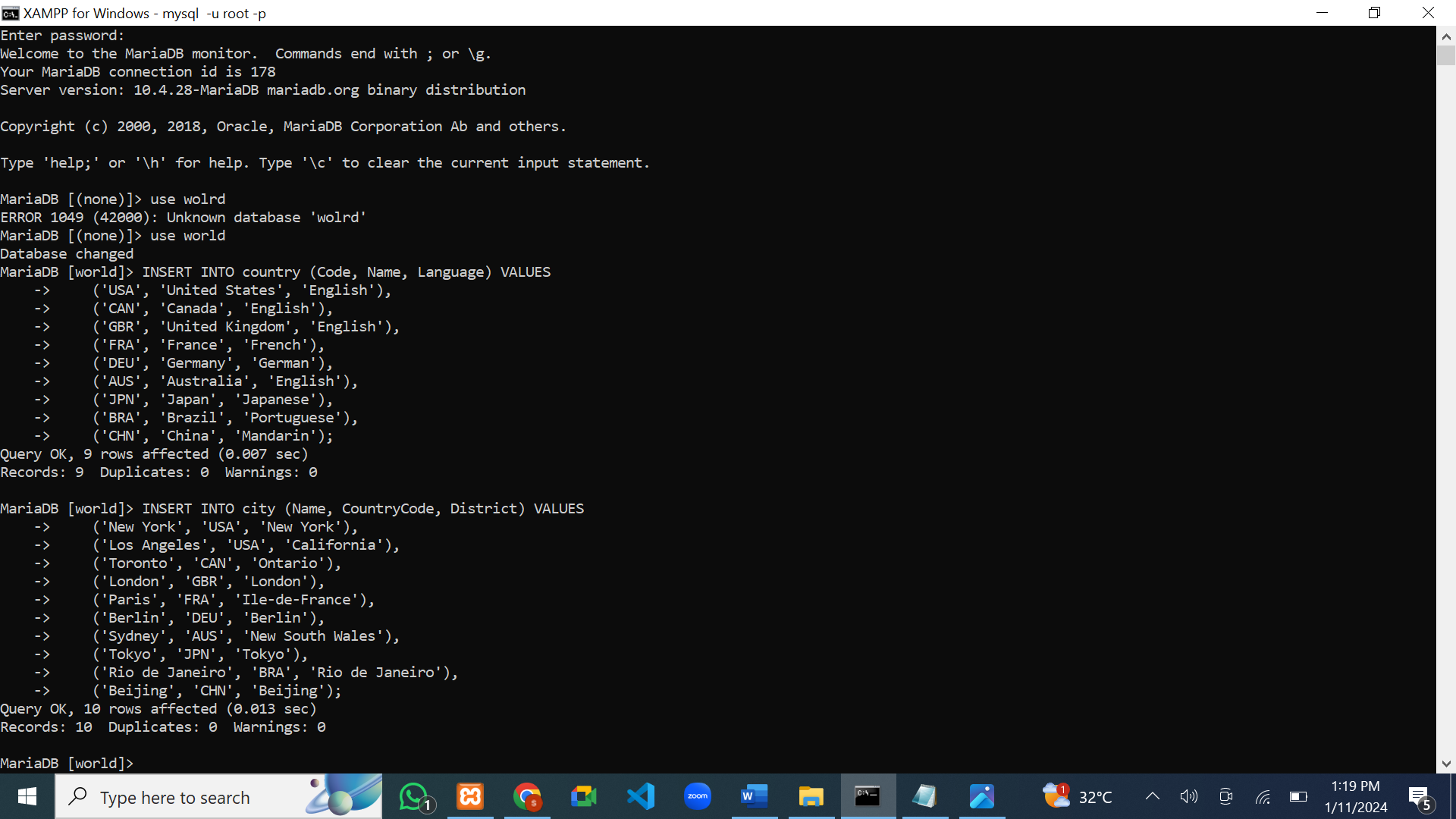
('Berlin', 'DEU', 'Berlin'),

('Sydney', 'AUS', 'New South Wales'),

('Tokyo', 'JPN', 'Tokyo'),

('Rio de Janeiro', 'BRA', 'Rio de Janeiro'),

('Beijing', 'CHN', 'Beijing');



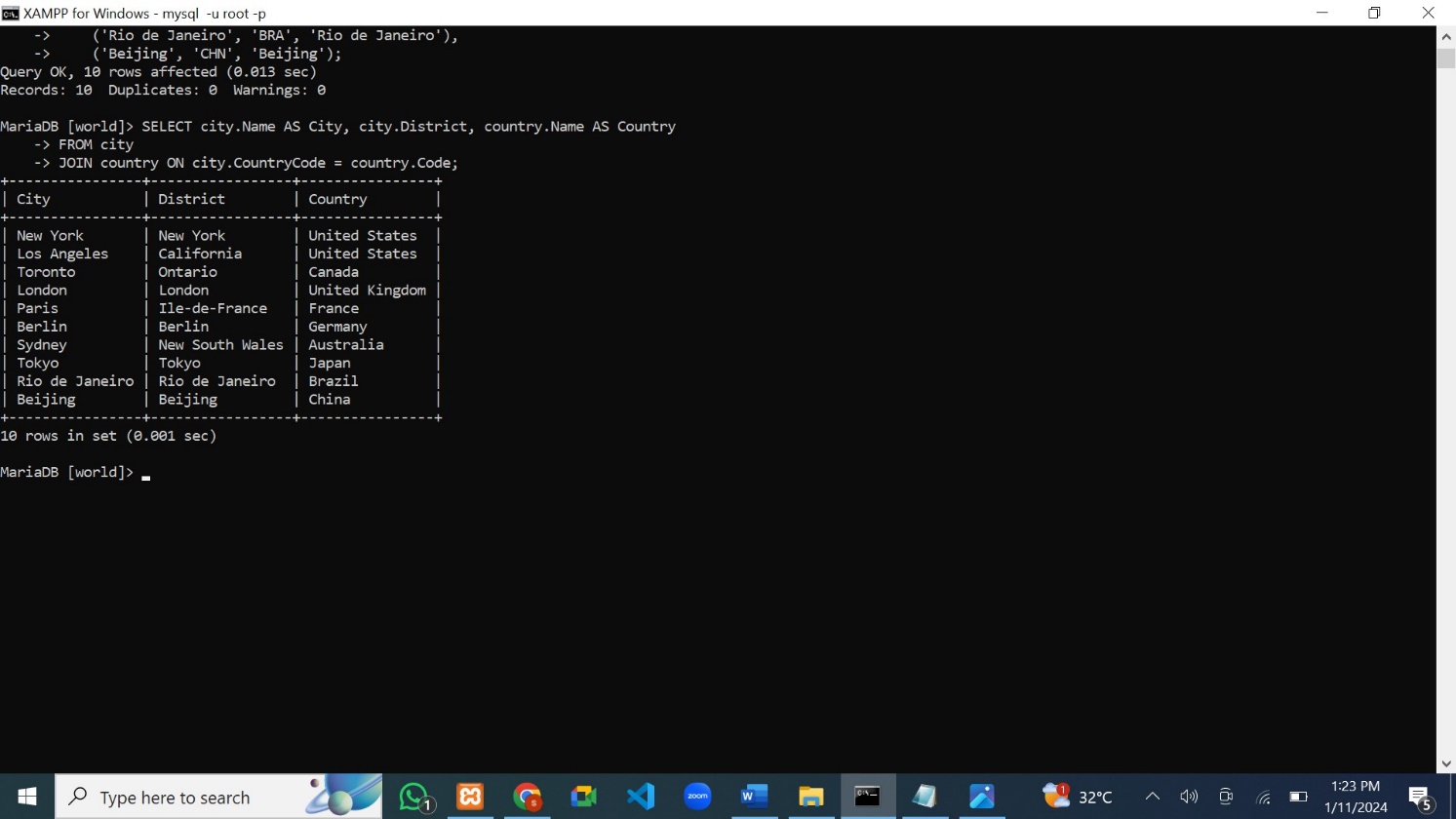
# Tulislah Query (SQL) untuk kasus dibawah dengan mengacu skema tabel di atas:

* 1. Tampilkan nama kota (City), distrik (District), dan negara asalnya (Country).

SELECT city.Name AS City, city.District, country.Name AS Country

FROM city

JOIN country ON city.CountryCode = country.Code;

`

# Tampilkan nama kota dan distrik dari semua kota di negara yang bahasanya English.

# SELECT city.Name AS City, city.District

# FROM city

# JOIN country ON city.CountryCode = country.Code

# WHERE country.Language = 'English';

# 

* 1. Jelaskan Pengertian Serta Perbedaan SQL Dengan NoSql

SQL

- (Relasional): Database SQL menggunakan skema tabel untuk menyimpan data yang memiliki relasi antar tabel. Data disimpan dalam bentuk tabel dengan baris dan kolom, dan setiap tabel memiliki skema yang didefinisikan dengan jelas.

- Cocok untuk aplikasi dengan struktur data yang stabil dan query yang kompleks

NoSQL

- (Non-relasional): Database NoSQL dapat menggunakan beberapa format penyimpanan, termasuk dokumen, grafik, key-value, dan lainnya. NoSQL memungkinkan fleksibilitas dalam penyimpanan data tanpa memerlukan skema yang ketat.

- Lebih cocok untuk aplikasi dengan kebutuhan perubahan skema yang cepat, dan penanganan beban kerja tinggi dengan data yang tidak selalu relasional.

# Berikan Contoh DB Nosql Beserta fungsi atau kegunaannya

# MongoDB: Database dokumen yang fleksibel. Cocok untuk menyimpan dan mengelola data dokumen.

* 1. Jika Saya Memiliki suatu kumpulan Data Yang Memiliki besar data -+2 TB, dapatkah anda Merekomendasikan Dbms yang Lebih efisien Untuk Mengelola data” Tsb? Berikan Penjelasan..

MongoDB dirancang untuk menangani data berskala besar dan dapat dikonfigurasi untuk mempertahankan performa dengan ukuran data yang besar.

# Jelaskan pengertian Relationship Database beserta contohnya..

# Database yang menggunakan relasi antar tabel. Contohnya adalah hubungan antara tabel 'City' dan 'Country' melalui kolom 'CountryCode'.

* 1. Coba Jelaskan Perbedaan Dbms Mysql Dan Postgresql secara Sederhana..

MySQL:

Dimiliki oleh Oracle.

Dikenal karena performa tinggi.

Model lisensi sederhana.

PostgreSQL:

Dikenal karena ketaatan standar SQL dan ekstensibilitas.

Mendukung query dan transaksi kompleks.

Open-source dan menggunakan lisensi yang lebih longgar (PostgreSQL License).

# 