1. Завантажте дані:

* Створіть схему pandemic у базі даних за допомогою SQL-команди.
* Оберіть її як схему за замовчуванням за допомогою SQL-команди.
* Імпортуйте [дані](https://drive.google.com/file/d/1lHEXJvu2omYRgvSek6mHq-iQ3RmGAQ7e/view?usp=sharing) за допомогою Import wizard так, як ви вже робили це у темі 3.

**CREATE DATABASE IF NOT EXISTS pandemic;**

**USE pandemic;**

2. Нормалізуйте таблицю infectious\_cases до 3ї нормальної форми. Збережіть у цій же схемі дві таблиці з нормалізованими даними:

**USE pandemic;**

**CREATE TABLE entities (**

**entity\_id INT AUTO\_INCREMENT PRIMARY KEY,**

**Entity VARCHAR(255) NOT NULL,**

**Code VARCHAR(255) NOT NULL,**

**UNIQUE (Entity(100), Code(100))**

**);**

**CREATE TABLE infectious\_cases\_normalized (**

**case\_id INT AUTO\_INCREMENT PRIMARY KEY,**

**entity\_id INT NOT NULL,**

**Year INT NOT NULL,**

**Disease VARCHAR(255) NOT NULL,**

**Number\_of\_cases TEXT,**

**FOREIGN KEY (entity\_id) REFERENCES entities(entity\_id)**

**);**

**INSERT INTO entities (Entity, Code)**

**SELECT DISTINCT Entity, Code**

**FROM infectious\_cases;**

**INSERT INTO infectious\_cases\_normalized (entity\_id, Year, Disease, Number\_of\_cases)**

**SELECT e.entity\_id, ic.Year, 'yaws', ic.Number\_yaws**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'polio', ic.polio\_cases**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'guinea\_worm', ic.cases\_guinea\_worm**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'rabies', ic.Number\_rabies**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'malaria', ic.Number\_malaria**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'hiv', ic.Number\_hiv**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'tuberculosis', ic.Number\_tuberculosis**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'smallpox', ic.Number\_smallpox**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code**

**UNION ALL**

**SELECT e.entity\_id, ic.Year, 'cholera', ic.Number\_cholera\_cases**

**FROM infectious\_cases ic**

**JOIN entities e ON ic.Entity = e.Entity AND ic.Code = e.Code;**

3. Проаналізуйте дані (для кожної унікальної комбінації Entity та Code або їх id порахуйте середнє, мінімальне, максимальне значення та суму для атрибута Number\_rabies)

- врахуйте, що атрибут Number\_rabies може містити порожні значення ‘’ — вам попередньо необхідно їх відфільтрувати;

- результат відсортуйте за порахованим середнім значенням у порядку спадання;

- оберіть тільки 10 рядків для виведення на екран:

**USE pandemic;**

**SELECT**

**e.entity\_id,**

**e.Entity,**

**e.Code,**

**AVG(CAST(NULLIF(icn.Number\_of\_cases, '') AS DECIMAL)) AS avg\_rabies,**

**MIN(CAST(NULLIF(icn.Number\_of\_cases, '') AS DECIMAL)) AS min\_rabies,**

**MAX(CAST(NULLIF(icn.Number\_of\_cases, '') AS DECIMAL)) AS max\_rabies,**

**SUM(CAST(NULLIF(icn.Number\_of\_cases, '') AS DECIMAL)) AS sum\_rabies**

**FROM**

**entities e**

**JOIN**

**infectious\_cases\_normalized icn ON e.entity\_id = icn.entity\_id**

**WHERE**

**icn.Disease = 'rabies' AND**

**icn.Number\_of\_cases != ''**

**GROUP BY**

**e.entity\_id, e.Entity, e.Code**

**ORDER BY**

**avg\_rabies DESC**

**LIMIT 10;**

4. Побудуйте колонку різниці в роках.

для оригінальної або нормованої таблиці для колонки Year побудуйте з використанням вбудованих SQL-функцій:

* атрибут, що створює дату першого січня відповідного року (наприклад, якщо атрибут містить значення ’1996’, то значення нового атрибута має бути ‘1996-01-01’);
* атрибут, що дорівнює поточній даті;
* атрибут, що дорівнює різниці в роках двох вищезгаданих колонок;
* перераховувати всі інші атрибути, такі як Number\_malaria, не потрібно:

**USE pandemic;**

**ALTER TABLE infectious\_cases\_normalized**

**ADD COLUMN start\_of\_year DATE,**

**ADD COLUMN current\_date\_col DATE,**

**ADD COLUMN years\_difference INT;**

**SET SQL\_SAFE\_UPDATES = 0;**

**UPDATE infectious\_cases\_normalized**

**SET**

**start\_of\_year = STR\_TO\_DATE(CONCAT(Year, '-01-01'), '%Y-%m-%d'),**

**current\_date\_col = CURDATE(),**

**years\_difference = YEAR(CURDATE()) - Year**

**WHERE**

**case\_id IS NOT NULL;**

**SET SQL\_SAFE\_UPDATES = 1;**

5. Побудуйте власну функцію (створіть і використайте функцію, що будує такий же атрибут, як і в попередньому завданні: функція має приймати на вхід значення року, а повертати різницю в роках між поточною датою та датою, створеною з атрибута року (1996 рік → ‘1996-01-01’):

**use pandemic;**

**DELIMITER //**

**CREATE FUNCTION year\_difference(input\_year INT)**

**RETURNS INT**

**DETERMINISTIC**

**NO SQL**

**BEGIN**

**DECLARE start\_date DATE;**

**DECLARE years\_diff INT;**

**SET start\_date = STR\_TO\_DATE(CONCAT(input\_year, '-01-01'), '%Y-%m-%d');**

**SET years\_diff = YEAR(CURDATE()) - YEAR(start\_date);**

**RETURN years\_diff;**

**END //**

**DELIMITER ;**

**SELECT year\_difference(1996) AS years\_difference;**