



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря
Сікорського»
ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ
**Кафедра системного програмування та спеціалізованих
комп'ютерних систем**

Лабораторна робота №1
з дисципліни
«Бази даних і засоби управління»

Виконав: студент III курсу
ФПМ групи КВ-83
Трофімцов Дмитро Сергійович
Перевірив: Павловський В.І.

Київ – 2020

Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL

Метою роботи є здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL.

Завдання роботи полягає у наступному:

1. Розробити концептуальну модель - модель «сутність-зв'язок» предметної галузі, обраної студентом самостійно, відповідно до пункту «Вимоги до ER-моделі»;
2. Перетворити розроблену модель у схему бази даних (таблиці) PostgreSQL;
3. Виконати нормалізацію схеми бази даних до третьої нормальної форми (3НФ);
4. Ознайомитись із інструментарієм PostgreSQL та pgAdmin 4 та внести декілька рядків даних у кожен з таблиць засобами pgAdmin 4.

Концептуальна модель предметної області “Тестування вакцини”

В концептуальній моделі предметної області “Тестування вакцини” (Рисунок 1) виділяються наступні сутності та зв'язки між ними:

1. Сутність “Компанії” з атрибутами: ID, назва, власник, країна;
2. Сутність “Вакцини” з атрибутами: ID, назва, компанія;
3. Сутність “Пацієнти” з атрибутами: ID, повне ім'я, вік;
4. Сутність “Результати” з атрибутами: ID, пацієнт, вакцина, статус, побічні ефекти;
5. Сутність “Типи захворювання” з атрибутами: ID, назва, рік відкриття, місце ураження.

Одна компанія може створити декілька варіантів вакцини, тому між сутностями “Компанії” та “Вакцини” зв'язок R(1:N).

Один пацієнт може приймати тільки одну вакцину із запропонованих типів, тому між сутностями “Пацієнти” та “Вакцини” зв'язок R(1:1).

Кожен пацієнт може бути заражений декількома штампами коронавірусу, та навпаки, одним штампом коронавірусу може бути

заражено декілька пацієнтів, тому між сутностями “Пацієнти” та “Типи захворювання” зв’язок R(N:M).

У кожного пацієнта після лікування конкретним типом вакцини передбачається єдиний результат лікування, тому між “Пацієнт” та “Результати” зв’язок R(1:1).

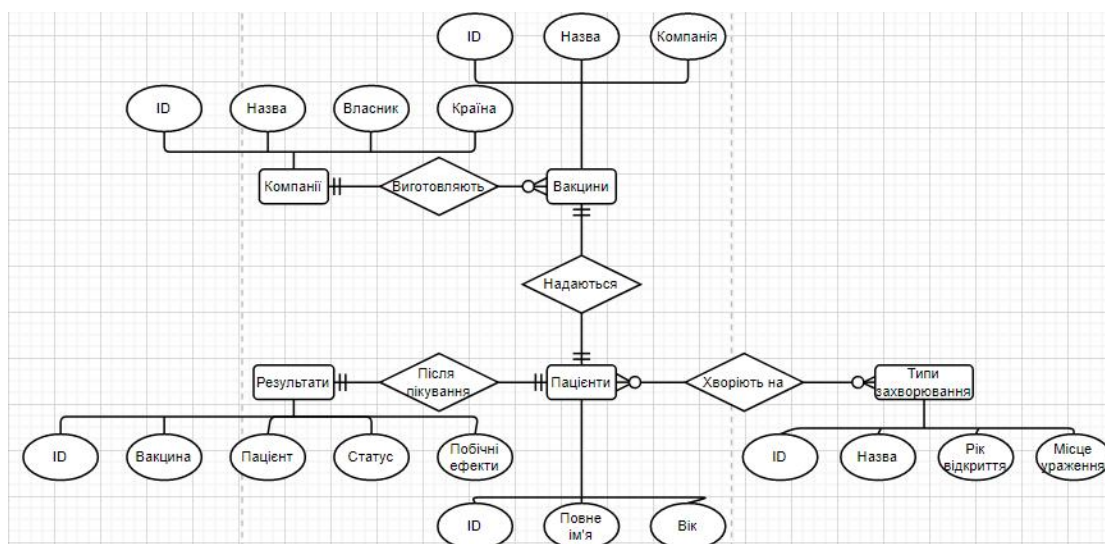


Рисунок 1 - Концептуальна модель предметної області “Тестування вакцини”.

Логічна модель (схема) БД “Тестування вакцини”

В логічній моделі (Рисунок 2):

1. Сутність “Компанії” перетворена в таблицю “Companies”;
2. Сутність “Вакцини” перетворена в таблицю “Vaccines”;
3. Сутність “Пацієнти” перетворена в таблицю “Patients”;
4. Сутність “Типи захворювання” перетворена в таблицю “CoronaType”;
5. Сутність “Результати” перетворена в таблицю “Results”.

Оскільки відношення між пацієнтами та типами захворювання R(N:M), потрібно створити ще одну додаткову таблицю з пацієнтами та захворюваннями під назвою “Patient/Corona”.

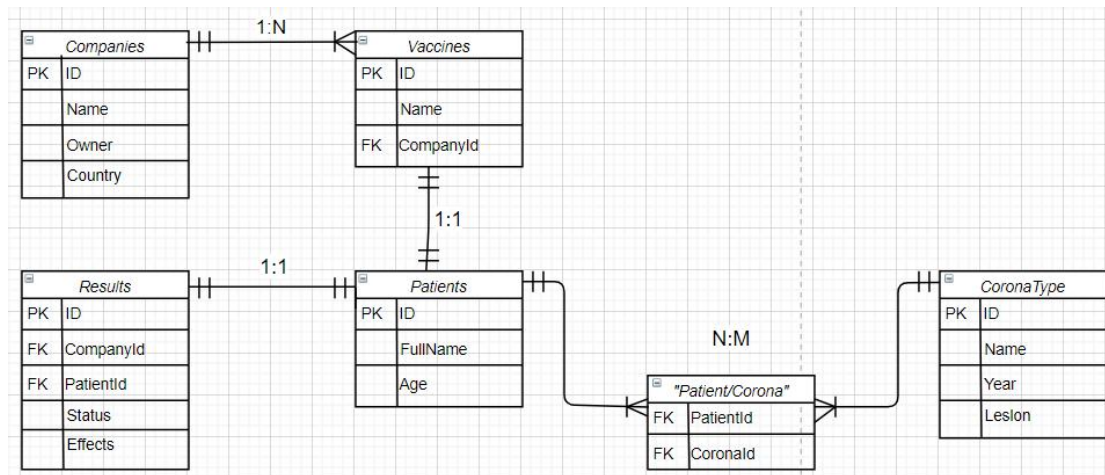


Рисунок 2 - Логічна модель предметної області “Тестування вакцини”.

Опис структури БД

Сутність	Атрибут	Тип
Companies (інформація про компанії)	Id - унікальний ID компанії Name – назва компанії Owner – повне ім'я власника компанії Country – країна, де знаходиться компанія	Числовий Текстовий Текстовий Текстовий
Vaccines (інформація про вакцини, що виготовляють компанії)	Id - унікальний ID вакцини Name – назва вакцини CompanyId – ID компанії, що виготовила вакцину	Числовий Текстовий Числовий
Patients (інформація про пацієнтів, які хворі на штампі коронавірусу)	Id - унікальний ID пацієнта FullName – повне ім'я пацієнта Age – вік пацієнта	Числовий Текстовий Числовий
CoronaType (інформація про штампі коронавірусу)	Id – унікальний ID хвороби Name – назва хвороби Year – рік відкриття Lesion – місце ураження	Числовий Текстовий Числовий Текстовий
Patient/Corona (інформація про пацієнтів та хвороби)	PatientId - ID пацієнта CoronaId - ID хвороби	Числовий Числовий
Results (інформація про	Id – унікальний ID звіту PatientId – ID пацієнта	Числовий Числовий

стан пацієнтів після лікування)	CoronaId – ID хвороби Status – статус здоров'я пацієнта Effects – побічні ефекти	Числовий Перечислювальний Перечислювальний
---------------------------------	--	--

Структура БД “Тестування вакцини”

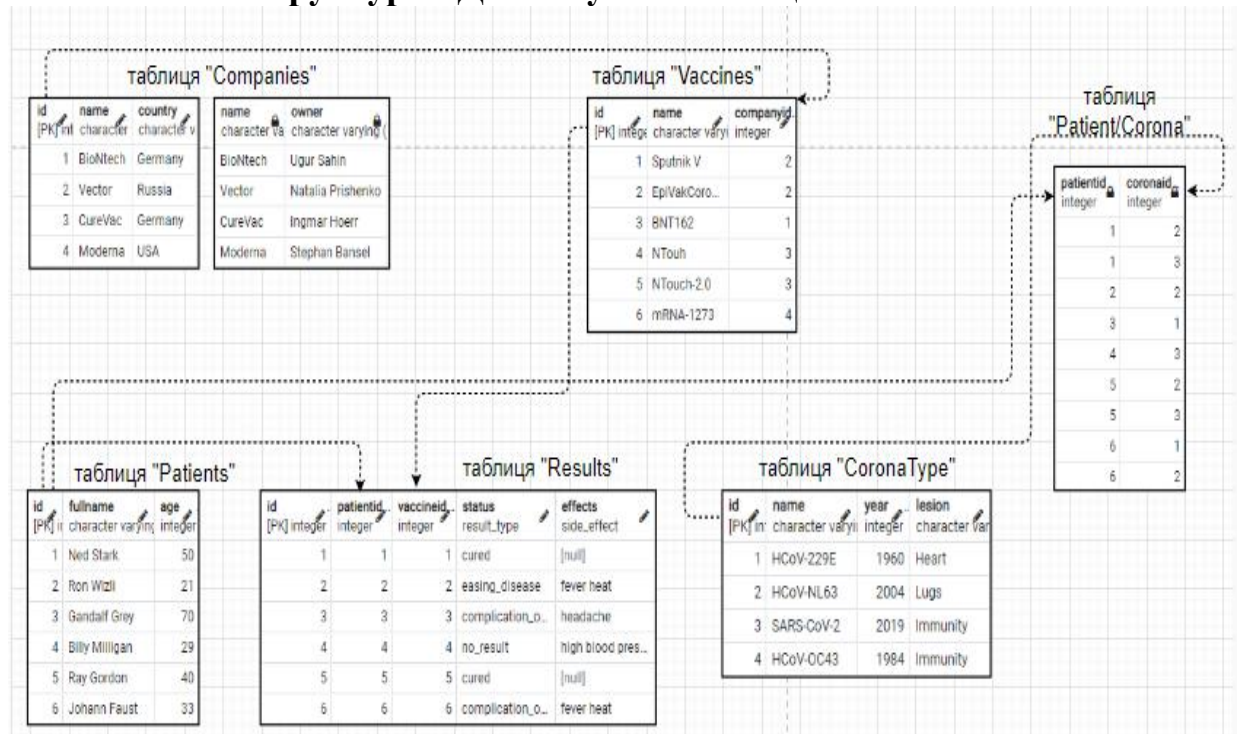


Рисунок 3 - Структурна модель предметної області “Тестування вакцини”

Схема відповідає 1НФ тому, що кожний стовбець таблиці є атомарним і кожний запис є унікальним.

Схема відповідає 2НФ тому, що схема відповідає 1НФ і всі таблиці мають первинні ключі, і всі атрибути описують первинний ключ повністю (не має ключів які складаються більше ніж з одного атрибута).

Схема відповідає 3НФ тому, що схема відповідає 2НФ і всі атрибути залежні тільки від первинного ключа (в ній немає транзитивних залежностей між неключовими атрибутами).

Структура БД “Тестування вакцини” в pgAdmin 4



Рисунок 4 - Структура предметної області “Тестування вакцини” в pgAdmin 4

Опис таблиць БД “Тестування вакцини” в pgAdmin 4

```
create table Companies(  
    Id serial not null,  
    Name varchar(40) not null,  
    Owner varchar(40) not null,  
    Country varchar(40) not null,  
    constraint company_primary_key primary key (Id)  
);  
  
insert into Companies (Name, Owner, Country) values  
( 'BioNtech', 'Ugur Sahin', 'Germany'),  
( 'Vector', 'Natalia Prishenko', 'Russia'),  
( 'CureVac', 'Ingmar Hoerr', 'Germany'),  
( 'Moderna', 'Stephan Bansel', 'USA');  
  
create table Vaccines(  
    Id serial not null,  
    Name varchar(40) not null,  
    CompanyId integer not null,  
    constraint vaccine_primary_key primary key (Id),  
    constraint company_fk foreign key (CompanyId)  
        references Companies (Id)  
        on update cascade  
        on delete cascade  
);  
  
insert into Vaccines (Name, CompanyId) values  
( 'Sputnik V', (select Id from Companies where Companies.Name = 'Vector')),  
( 'EpiVakCorona', (select Id from Companies where Companies.Name = 'Vector')),  
( 'BNT162', (select Id from Companies where Companies.Name = 'BioNtech')),  
( 'NTouh', (select Id from Companies where Companies.Name = 'CureVac')),  
( 'NTouch-2.0', (select Id from Companies where Companies.Name = 'CureVac')),  
( 'mRNA-1273', (select Id from Companies where Companies.Name = 'Moderna'));  
  
create table Patients(  
    Id serial not null,  
    FullName varchar(40) not null,  
    Age integer not null,  
    constraint patient_primary_key primary key (Id)  
);  
  
insert into Patients (FullName, Age) values  
( 'Ned Stark', 50),  
( 'Ron Wizli', 21),  
( 'Gandalf Grey', 70),
```



```
('Billy Milligan', 29),
('Ray Gordon', 40),
('Johann Faust', 33);
```

```
create type result_type as enum ('cured', 'easing_disease', 'complication_of_disease', 'no_result');
create type side_effect as enum ('fever heat', 'headache', 'high blood pressure', 'low pressure');
```

```
create table Results(
    Id serial not null,
    PatientId integer not null,
    VaccineId integer not null,
    Status result_type not null,
    Effects side_effect,
    constraint resault_primary_key primary key (Id),
    constraint patient_fk foreign key (PatientId)
        references Patients(Id)
        on update cascade
        on delete cascade,
    constraint vaccine_fk foreign key (VaccineId)
        references Vaccines(Id)
        on update cascade
        on delete cascade
);
```

```
insert into Results (PatientId, VaccineId, Status, Effects) values
(
    (select Id from Patients where Patients.FullName = 'Ned Stark'),
    (select Id from Vaccines where Vaccines.Name = 'Sputnik V'),
    'cured',
    null
),
(
    (select Id from Patients where Patients.FullName = 'Ron Wizli'),
    (select Id from Vaccines where Vaccines.Name = 'EpiVakCorona'),
    'easing_disease',
    'fever heat'
),
(
    (select Id from Patients where Patients.FullName = 'Gandalf Grey'),
    (select Id from Vaccines where Vaccines.Name = 'BNT162'),
    'complication_of_disease',
    'headache'
),
(
```



```

(
    (select Id from Patients where Patients.FullName = 'Billy Milligan'),
    (select Id from Vaccines where Vaccines.Name = 'NTouh'),
    'no_result',
    'high blood pressure'
),
(
    (select Id from Patients where Patients.FullName = 'Ray Gordon'),
    (select Id from Vaccines where Vaccines.Name = 'NTouch-2.0'),
    'cured',
    null
),
(
    (select Id from Patients where Patients.FullName = 'Johann Faust'),
    (select Id from Vaccines where Vaccines.Name = 'mRNA-1273'),
    'complication_of_disease',
    'fever heat'
);

create table CoronaType(
    Id serial not null,
    Name varchar(40) not null,
    Year integer not null,
    Lesion varchar(40) not null,
    constraint corona_type_primary_key primary key (Id)
);

insert into CoronaType (Name, Year, Lesion) values
('HCoV-229E', 1960, 'Heart'),
('HCoV-NL63', 2004, 'Lugs'),
('SARS-CoV-2', 2019, 'Immunity'),
('HCoV-OC43', 1984, 'Immunity');

create table "Patient/Corona"(
    PatientId integer not null,
    CoronaId integer not null,
    constraint patient_fk foreign key (PatientId)
        references Patients (Id)
        on update cascade
        on delete cascade,
    constraint corona_fk foreign key (CoronaId)
        references CoronaType (Id)
        on update cascade
        on delete cascade
);

insert into "Patient/Corona" (PatientId, CoronaId) values
(1, 2), (1, 3), (2, 2), (3, 1), (4, 3), (5, 2), (5, 3), (6, 1), (6, 2);

```