

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»**

Тема: «***Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL***»

Виконав: студент 3 курсу

ФПМ групи КВ-83

Проценко Владислав

Перевірив: Павловський В.І.

Київ – 2020

***Мета роботи****:* здобуття вмінь проектування бази даних та практичних

навичок створення реляційних баз даних за допомогою PostgreSQL.

***Завдання:***

1. Розробити 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 ввести декілька рядків даних у кожну з таблиць засобами pgAdmin 4.

***Варіант (предметна область):*** база даних для спортивних змагань з футболу.

**Звіт**

У спортивних змаганнях з футболу можуть бути різні турніри. У кожному турнірі грає певна кількість команд за деяким розкладом. Команди можуть грати в декількох турнірах. Гравці можуть грати в одній команді. Також у кожної команди свій власний стадіон. Кожен матч має домашню і виїздну команду, та турнір в рамках якого проводится цей матч.

В концептуальній моделі предметної області "Футбольні змагання"

(Рисунок 1) виділяються наступні сутності та зв'язки між ними:

* Сутність «Player» з атрибутами: id, lastname, firstname, birthday, club\_id;
* Сутність «Club» з атрибутами: id, name;
* Сутність «Stadium» з атрибутами: id, title, capacity, city;
* Сутність «Match» з атрибутами: id, home\_club\_id, away\_club\_id, datetime, result, stadium\_id, ticket\_cost, tournament\_id.

Між сутностями «Club» та «Tournament» зв'язок «participant» типу N:M, тому що клуб може виступати в багатьох турнірах, а в кожному турніру в свою чергу виступає багато команд.

Між сутностями «Club» та «Player» зв'язок «has» типу 1:N , тому що в клубі може бути багато гравців, але футболіст може бути лише в одному клубі.

Між сутностями «Club» та «Stadium» зв'язок «own» типу 1:1 , тому що у кожного клуба свій стадіон.

Між сутностями «Match» та «Club» зв'язок «has» типу M:N , тому що клуб приймає учать в багатьох матчах, а в самому матчі участь приймають 2 клуби.

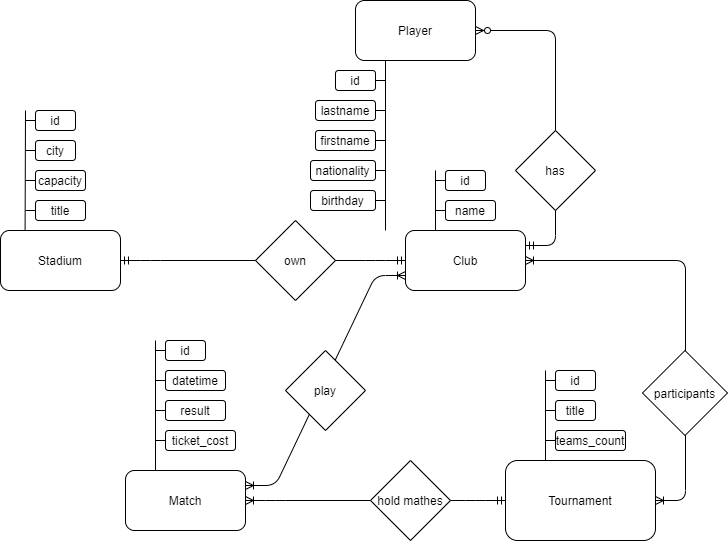


Рисунок. 1 ER-модель предметної області "Футбольні змагання". Використано сервіс Draw.io

1. Схема бази даних PostgreSQL на основі ER-моделі предметної області "Футбольні змагання" (Рисунок 1).

Сутність «Club» було перетворено в таблицю «Club».

Сутність «Player» було перетворено в таблицю «Player».

Сутність «Stadium» було перетворено в таблицю «Staduim».

Зв’язок «own» типу 1:1 призвів до появи «club\_stadium».

Зв'язок «participants» типу N:M призвів до появи таблиці «clubs\_tournaments».

Зв’язок «has» типу 1:M призвів до появи в таблиці «Player» додаткового поля club\_id.

Сутність «Tournament» було перетворено в таблицю «Tournament»

Зв’язок «hold matches» типу 1:M призвів до появи в таблиці «Match» додаткового поля tournament\_id.

Зв'язок «play» типу N:M призвів до появи в таблиці «Match» двох полів: home\_club\_id та away\_club\_id.

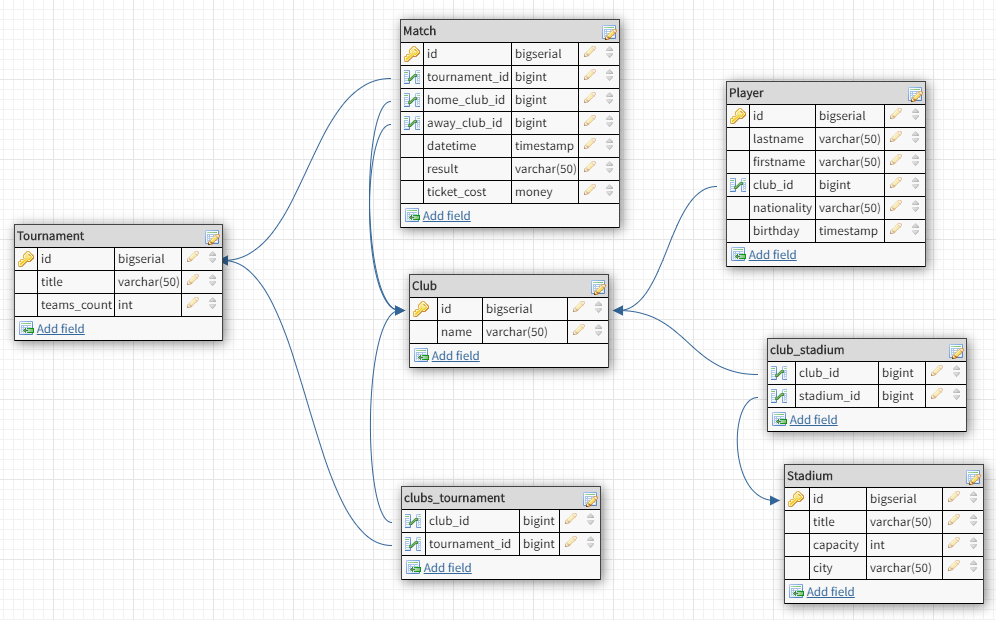
****

Рисунок. 2 Схема бази даних PostgreSQL на основі ER-моделі предметної області "Футбольні змагання".

Таблиця 1 – текстове представлення логічної модели БД.

|  |  |  |
| --- | --- | --- |
| **Таблиця** | **Атрибути** | **Тип** |
| Player – таблиця для сутності Player. Зберігає дані про футболістів | (PK)id – унікальний ключ гравця, не допускає NULL  lastname – прізвище гравця  firstname – ім’я гравця  nationality – рідна країна  (FK)club\_id – клуб, в якому грає футболіст  birthday – дата народження | bigserial  character varying(20)  character varying(20)  character varying(40)  bigint  timestamp |
| Club – таблиця для сутності Club. Зберігає дані про футбольний клуби | (PK)id – унікальний номер клуба  name – назва клуба | bigserial  character varying(20) |
| Tournament – таблиця для сутності Tournament. Зберігає дані про всі турніри | (PK)id – унікальний номер турніру  title – назва турніру  teams\_count – кількість команд учасниць | bigserial  character varying(20)  int |
| Stadium – таблиця для сутності Stadium. Зберігає дані про стадіони | (PK)id – унікальний номер стадіону  title – назва стадіону  capacity – максимальна кількість глядачів  city – місто, в якому находиться стадіон | bigserial  character varying(20)  int  character varying(20) |
| Match – таблиця для сутності Match. Зберігає дані про альбоми | (PK)id – унікальний номер матчу  (FK)tournament\_id – турнір, який організовує матч  (FK)home\_club\_id – домашня команда  (FK)away\_club\_id – виїзна команда  datetime – час проведення матчу  result – результат матчу, null якщо матч ще не відбувся  ticket\_cost – вартість квитка | bigserial  bigint  bigint  bigint  timestamp  character varying(10)  money |
| clubs\_tournament – містить інформація про те, в яких турнірах грають команди. | (PK)id – унікальний номер  (FK)club\_id – номер клубу  (FK)tournament\_id – номер турніру | biserial  bigint  bigint |
| club\_stadium – містить інформацію про те, яким стадіоном володіє каманда. | (FK)club\_id – номер клубу, unique.  (FK)stadium\_id – номер стадіону, unique. | bigint  bigint |

1. Нормалізація БД до третьої нормальної форми

Таблиці бази даних відповідає 1НФ тому, що всі рядки унікальні, всі атрибути не мають нереляційні типи (нереляційні типи даних – масиви, об’єкти і т.д.).

Таблиці бази даних відповідає 2НФ тому, що всі таблиці мають первинні ключі, і всі неключові атрибути повністю залежать від ключа і не мають підмножин.

Схема бази даних відповідає 3НФ тому, що всі не ключові атрибути залежні тільки від первинного ключа і нетранзитивно залежні від нього.

Перевіримо на транзитивність всі таблиці.

«Stadium»:

id → title, capacity, city;

id → title;

id → capacity;

id → city;

title ↮ capacity;

capacity ↮ city;

city ↮ title.

«Player»:

id → lastname, firstname, club\_id, birthday;

id → lastname;

id → firstname;

id → club\_id;

id → birthday.

club\_id ↮ lastname;

lastname ↮ firstname;

firstname ↮ birthday;

birthday ↮ club\_id.

«Club»:

id → name, stadium\_id;

id → name;

id → stadium\_id;

stadium\_id ↮ name;

name ↮ stadium\_id.

«Match»:

id → tournament\_id, home\_club\_id, away\_club\_id, ticket\_cost, datetime, result;

id → tournament\_id;

id → home\_club\_id;

id → away\_club\_id;

id → ticket\_cost;

id → datetime;

id → result;

tournament\_id ↮ home\_club\_id;

home\_club\_id ↮ away\_club\_id;

away\_club\_id ↮ ticket\_cost;

ticket\_cost ↮ ticket\_cost;

ticket\_cost ↮ datetime;

datetime ↮ result;

result ↮ tournament\_id.

«Tournament»:

id → title, teams\_count;

id → title;

id → teams\_count;

title ↮ teams\_count;

teams\_count ↮ title.

«clubs\_tournaments»:

id → club\_id, tournament\_id;

id → club\_id;

id → tournament\_id;

tournament\_id ↮ club\_id;

club\_id ↮ tournament\_id.

«club\_stadium»:

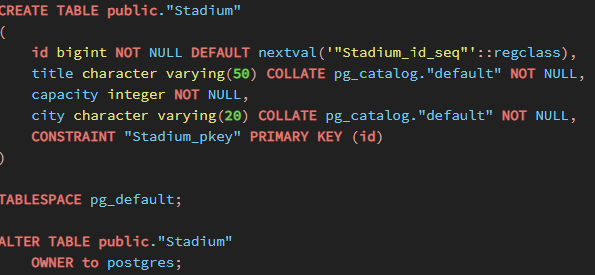
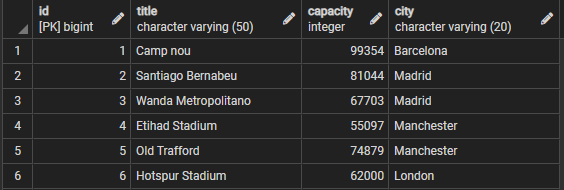
stadium\_id ↮ club\_id;

club\_id ↮ stadium\_id.

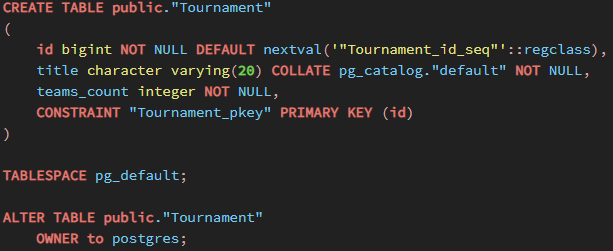
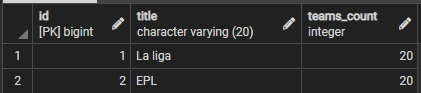
1. Створення БД в PostgreSQL з використанням pgAdmin4.

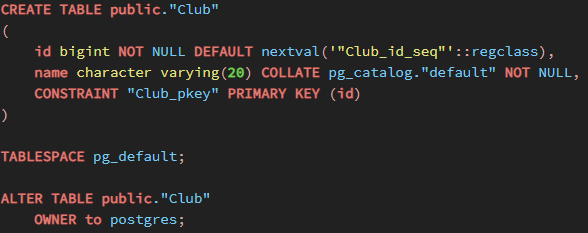
|  |
| --- |
|  |

Таблиця «Стадіон»

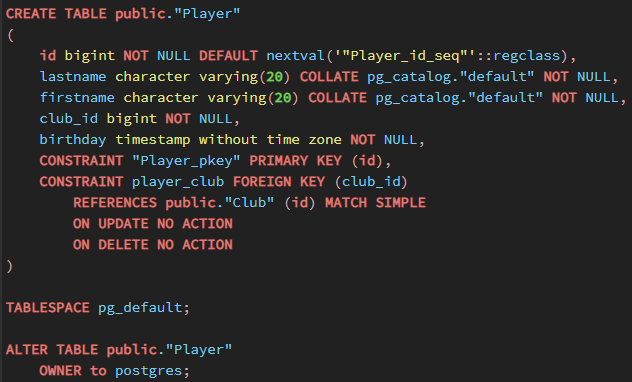
Таблиця «Турнір»

Таблиця «Клуб»  




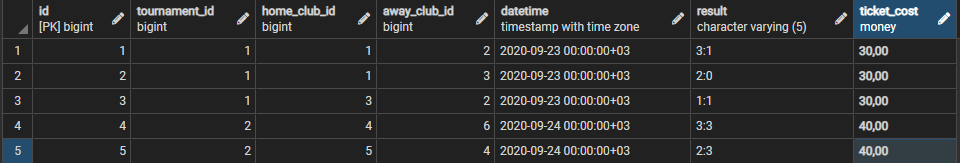
Таблиця «Гравець»





Таблиця «Матч»





Таблиця «Клуби\_Турніри»


Таблиця «Клуб\_Стадіон»

