

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет прикладної математики

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Лабораторна робота № 1

з дисципліни

«Бази даних та засоби управління»

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

Виконала студентка групи:

КВ-11 Михайліченко Софія

Перевірив: Петрашенко А. В.

Оцінка:

Київ – 2023

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 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.

Предметна область:

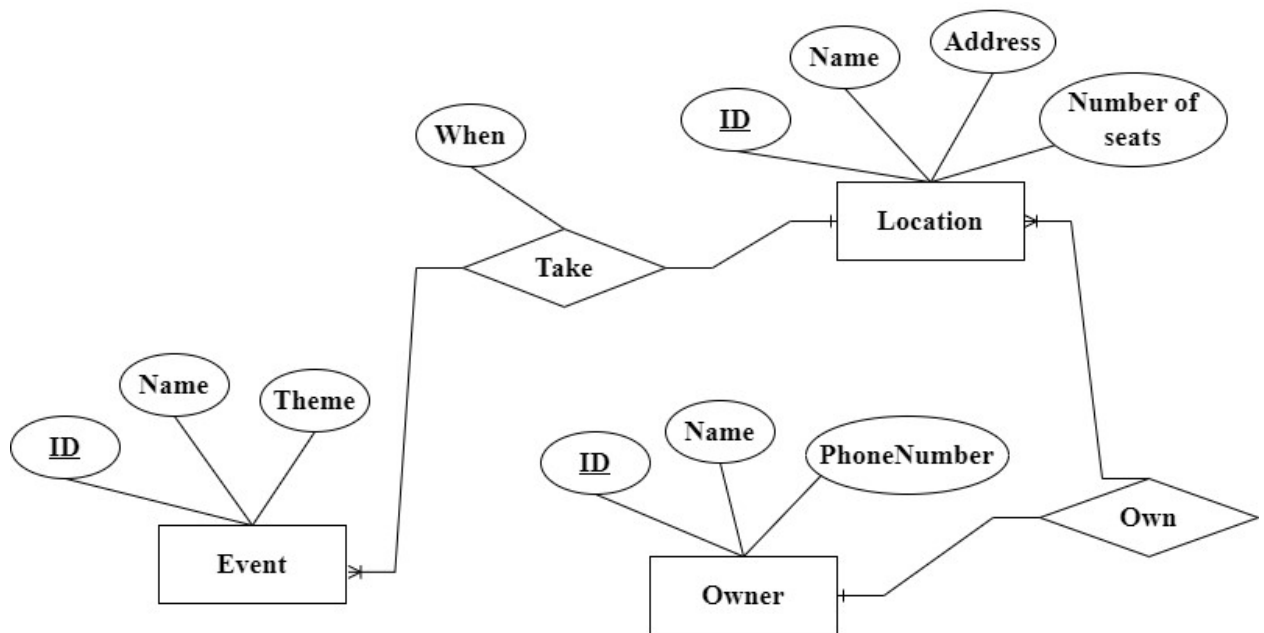
Платформа для бронювання та управління майданчиками для подій.

GitHub репозиторій:

Контакт в Telegram: https://t.me/sone_ttaa

Завдання № 1

Розробка моделі «сутність-зв'язок» предметної галузі для проектування бази даних «A platform for booking and managing venues for events». Предметна галузь – 65 «Платформа для бронювання та управління майданчиками для подій.».



Малюнок 1. ER-діаграма побудована за нотацією «Crow`s foot»

Сутності з описом призначення:

Предметна галузь «A platform for booking and managing venues for events» включає в себе 3 сутності, кожна сутність містить декілька атрибутів:

1. Event (ID, Name, Theme).
2. Location (ID, Name, Address, Number of seats).
3. Owner (ID, Name, PhoneNumber).

Сутність Event описує подію, яка відбудеться на певній локації. Кожна подія має свій ідентифікатор ID, а також має назву та тематику.

Сутність Location описує місце, де певна подія відбуватиметься. Кожна локація має свій ідентифікатор, назву, адресу та кількість можливих відвідувачів за один раз.

Сутність Owner описує власника місця, де відбувається певна подія. Кожен

власник має свій ідентифікатор, ім'я та номер телефону.

Зв'язки між сутностями:

Зв'язок між Event та Location:

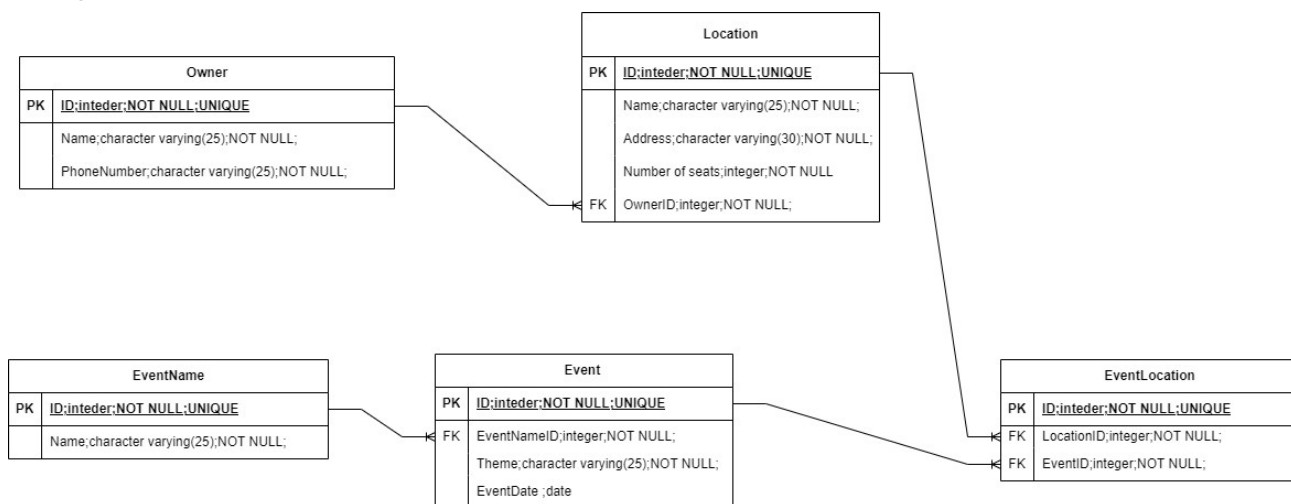
Кожна подія має своє місце проведення. Зв'язок 1:N – на одній локації можуть відбуватися декілька подій (наприклад в ресторані можуть замовити столики під святкування різних подій).

Зв'язок між Location та Owner:

Кожна локація має свого власника. Оскільки один власник може мати декілька локацій, зв'язок 1:N.

Завдання № 2

Перетворення розробленої моделі «сутність-зв'язок» у схему бази даних PostgreSQL



Малюнок 2. Схема бази даних у графічному вигляді

Опис процесу перетворення:

Сутність Event було перетворено на таблицю Event. Первинний ключ (ідентифікатор) ID, атрибути: EventNameID, Theme(тематика), EventDate.

В ER-діаграмі був зазначений атрибут «When». В таблиці БД він перетворився на поле типу date в таблиці Event.

Сутність Location було перетворено на таблицю Location. Первинний ключ

(ідентифікатор) ID, атрибути: Name, Address, Number of seats, OwnerID.

Сутність Owner було перетворено на таблицю Owner. Первинний ключ (ідентифікатор) ID, атрибути: Name, PhoneNumber.

Як бачимо, була створена ще одна таблиця EventName для того, щоб одна назва івенту не повторювалась декілька раз, це вимагало б додаткової пам'яті. Це також реалізовує НФ1, оскільки 1 нормальна форма вимагає відсутні груп полів, які повторюються. В таблиці Event створено зовнішній ключ FKeventname, він посилається на поле ID в таблиці EventName.

Було також створено таблицю EventLocation. Первинний ключ (ідентифікатор) ID, атрибути: LocationID, EventID. Таблиця EventLocation була створена для того, щоб утворювати пару подія-локація. Можна було б додати в таблицю Event поле LocationID, яке теоретично б пов'язувало подію з місцем, де відбувається певна подія. АЛЕ: якщо б 1 і той самий івент відбувався в двох різних локаціях, то довелося би мати 2 майже однакові записи в таблиці Event. Єдиною відмінністю було б поле LocationID. Отже, знову була б виділена пам'ять на всі інші поля (ID, Name, Theme(тематика)). Це вимагало б додаткової пам'яті. А так, завдяки такій таблиці EventLocation, реалізована можливість проведення одного і того ж самого івенту на різних локаціях. Це також реалізовує НФ1, оскільки 1 нормальна форма вимагає відсутні груп полів, які повторюються. В таблиці EventLocation створено два зовнішніх ключа: FKlocation та FKevent. Вони посилаються на поля ID в таблицях Location та Event відповідно.

Кожна локація має свого власника. В таблиці БД це було обумовлено завдяки зовнішньому ключу FKowner, зв'язок «own» (1:N).

Завдання № 3
Функціональні залежності:

1. Event (ID, Name, Theme).

ID→Name

ID→Theme

2. Location (ID, Name, Address, Number of seats).

ID→Name

ID→ Address

ID→ Number of seats

3. Owner (ID, Name, PhoneNumber).

ID→Name

ID→ PhoneNumber

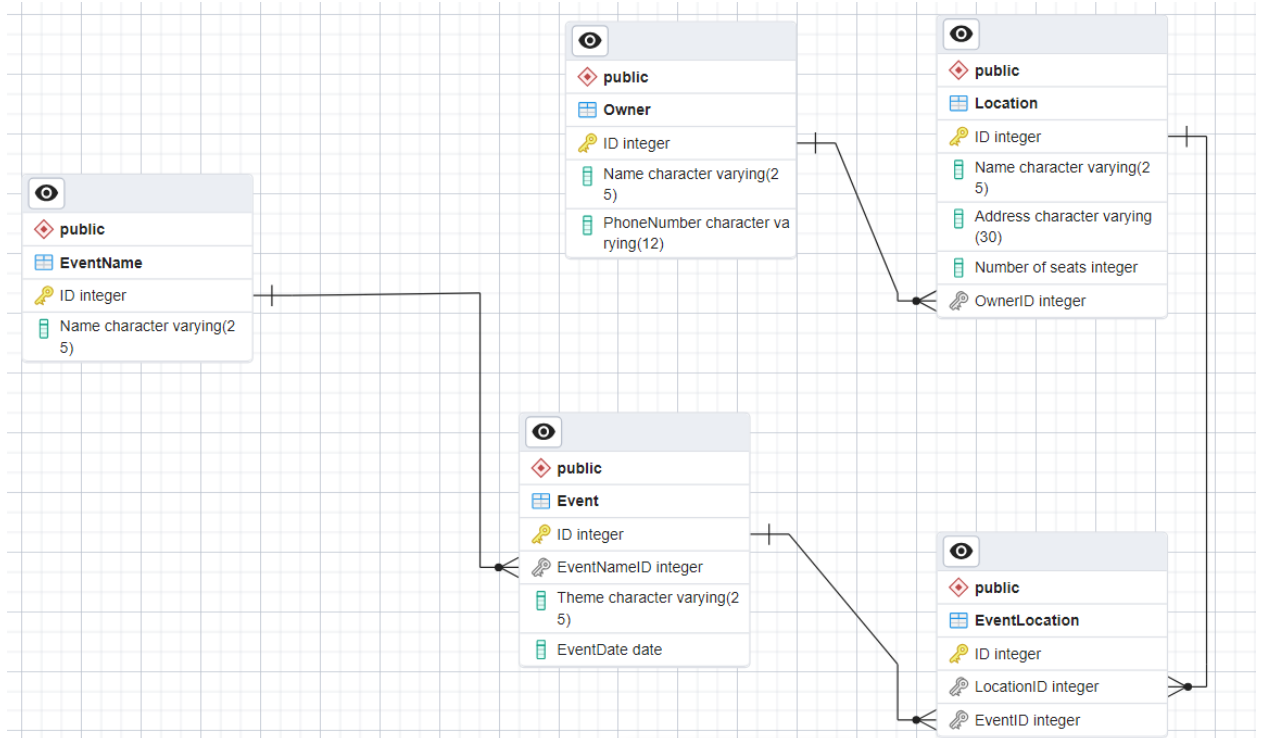
Схема бази даних відповідає 1НФ, тому що значення в кожній комірці таблиці є атомарними, кожне поле таблиці є неподільним, кожен рядок є унікальним, немає повторень рядків.

Схема бази даних відповідає 2НФ, бо вона відповідає 1НФ та кожен неключовий атрибут залежить від первинного повного ключа, отже первинний ключ одразу визначає запис та не є надмірним.

Схема бази даних відповідає 3НФ, тому що вона відповідає 2НФ та кожен неключовий атрибут не є транзитивно залежним від кожного кандидатного ключа. В таблицях нема не ключового поля, яке залежить від значення іншого не ключового поля.

Завдання № 4

Ознайомлення із інструментарієм PostgreSQL та pgAdmin 4 та внесення даних у кожну з таблиць засобами pgAdmin 4.



Малюнок 3. Схема бази даних у pgAdmin4

- ▼ Tables (5)
 - > Event
 - > EventLocation
 - > EventName
 - > Location
 - > Owner

- Tables (5)
 - Event
 - Columns (4)
 - ID
 - EventNameID
 - Theme
 - EventDate
 - Constraints (2)
 - Event_pkey
 - FKeventname
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - EventLocation
 - Columns (3)
 - ID
 - LocationID
 - EventID
 - Constraints (3)
 - EventLocation_pkey
 - FKevent
 - FKlocation
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - EventName
 - Columns (2)
 - ID
 - Name
 - Constraints (1)
 - EventName_pkey
 - Indexes
 - RLS Policies
 - Rules
 - Triggers

- Location
 - Columns (5)
 - ID
 - Name
 - Address
 - Number of seats
 - OwnerID
 - Constraints (2)
 - FKowner
 - Location_pkey
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
- Owner
 - Columns (3)
 - ID
 - Name
 - PhoneNumber
 - Constraints (1)
 - Owner_pkey
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
- Trigger Functions
- Types
- Views

Таблица Event:

Event

General

Columns

Advanced

Constraints

Parameters









Security

SQL

Inherited from table(s)

Select to inherit from...

Columns

| | Name | Data type | Length/Precision | Scale | Not NULL? | Primary key? | Default |
|---|-------------|-------------------|------------------|-------|-------------------------------------|-------------------------------------|---------|
|   | ID | integer | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
|   | EventNameID | integer | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
|   | Theme | character varying | 25 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
|   | EventDate | date | | | <input type="checkbox"/> | <input type="checkbox"/> | |

Query

Query History

1

SELECT * FROM public."Event"

2

ORDER BY "ID" ASC

| | ID [PK] integer | EventNameID integer | Theme character varying (25) | EventDate date |
|---|--------------------|------------------------|---------------------------------|-------------------|
| 1 | 1 | 1 | Halloween | 2025-10-30 |
| 2 | 2 | 2 | Musical | 2025-07-02 |
| 3 | 3 | 1 | Birthday | 2025-09-12 |
| 4 | 4 | 2 | Pop | 2025-01-25 |
| 5 | 5 | 3 | Gothic | 2026-03-05 |
| 6 | 6 | 4 | Gallery | 2026-07-07 |

Таблица EventLocation:

EventLocation

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Inherited from table(s) Select to inherit from...

Columns

| | Name | Data type | Length/Precision | Scale | Not NULL? | Primary key? | Default |
|--|------------|-----------|------------------|-------|-------------------------------------|-------------------------------------|---------|
| | ID | integer | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
| | LocationID | integer | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| | EventID | integer | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |

EventLocation

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Primary KeyForeign KeyCheckUniqueExclude

| | Name | Columns | Referenced Table |
|--|------------|----------------------|------------------|
| | FKevent | (EventID) -> (ID) | public.Event |
| | FKlocation | (LocationID) -> (ID) | public.Location |

QueryQuery History

```
1 SELECT * FROM public."EventLocation"
2 ORDER BY "ID" ASC
```

Data OutputMessagesNotifications

| | ID [PK] integer | LocationID integer | EventID integer |
|---|--------------------|-----------------------|--------------------|
| 1 | 1 | 1 | 2 |
| 2 | 2 | 2 | 1 |
| 3 | 3 | 1 | 3 |
| 4 | 4 | 4 | 4 |
| 5 | 5 | 3 | 4 |

Таблица EventName:

EventName

General

Columns

Advanced

Constraints

Parameters





Security

SQL

Inherited from table(s)

Select to inherit from...

Columns

| | Name | Data type | Length/Precision | Scale | Not NULL? | Primary key? | Default |
|---|------|-------------------|------------------|-------|-------------------------------------|-------------------------------------|---------|
|   | ID | integer | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
|   | Name | character varying | 25 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |

Query

Query History

1

SELECT * FROM public."EventName"

2

ORDER BY "ID" ASC

Data Output

Messages

Notifications



| | ID | Name |
|---|--------------|------------------------|
| | [PK] integer | character varying (25) |
| 1 | 1 | Party |
| 2 | 2 | Concert |
| 3 | 3 | Wedding |
| 4 | 4 | Charity evening |

Таблица Location:

Location

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Inherited from table(s)

Select to inherit from...

Columns

| | Name | Data type | Length/Precision | Scale | Not NULL? | Primary key? | Default |
|--|-----------------|-------------------|------------------|-------|-------------------------------------|-------------------------------------|---------|
| | ID | integer | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
| | Name | character varying | 25 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| | Address | character varying | 30 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| | Number of seats | integer | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| | OwnerID | integer | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |

Location

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Primary KeyForeign KeyCheckUniqueExclude

| | Name | Columns | Referenced Table |
|--|---------|-------------------|------------------|
| | FKowner | (OwnerID) -> (ID) | public.Owner |

QueryQuery History

1SELECT * FROM public."Location"

2ORDER BY "ID" ASC

Data OutputMessagesNotifications

| | ID [PK] integer | Name character varying (25) | Address character varying (30) | Number of seats integer | OwnerID integer |
|---|--------------------|--------------------------------|-----------------------------------|----------------------------|--------------------|
| 1 | 1 | Star Lounge | 123 Galaxy Street | 200 | 1 |
| 2 | 2 | Mystic Gardens | 789 Enchanted Avenue | 150 | 2 |
| 3 | 3 | Neon Plaza | 456 Luminescent Lane | 300 | 3 |
| 4 | 4 | Sapphire Lounge | 567 Celestial Boulevard | 250 | 4 |
| 5 | 5 | Velvet Sky Bar | 890 Stellar Drive | 180 | 2 |

Таблиця Owner:

Owner

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Inherited from table(s)

Select to inherit from...

Columns

| | Name | Data type | Length/Precision | Scale | Not NULL? | Primary key? | Default |
|--|-------------|-------------------|------------------|-------|-------------------------------------|-------------------------------------|---------|
| | ID | integer | | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
| | Name | character varying | 25 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| | PhoneNumber | character varying | 12 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |

QueryQuery History

1

SELECT * FROM public."Owner"

2

ORDER BY "ID" ASC

Data OutputMessagesNotifications

| | ID [PK] integer | Name character varying (25) | PhoneNumber character varying (12) |
|---|--------------------|--------------------------------|---------------------------------------|
| 1 | 1 | Ethan Turner | 876-5432 |
| 2 | 2 | Olivia Bennett | 234-5678 |
| 3 | 3 | Jackson Mitchell | 987-6543 |
| 4 | 4 | Ava Parker | 123-4567 |