

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**

*на тему: “Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL”*

Виконав:

студент ІII курсу

групи КВ-01

Степаненко Павло

Перевірив: Павловський В.І.

Київ – 2022

*Метою роботи є* здобуття 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 кожну з таблиць засобами pgAdmin 4.

*Зміст звіту*

1. Опис проблемного середовища;
2. Концептуальна модель предметної області;
3. Логічна модель (схема) даних БД;
4. Склад СУБД PostgreSQL;
5. Список обмежень цілісності в термінах СУБД PostgreSQL;
6. Фізична модель (схема) даних БД в pgAdmin III;
7. Приклад вмісту БД.

**Опис предметної галузі «Готель»**

Концептуальна модель обраної предметної області містить інформацію про кімнати готелю, робітників, гостей готелю, бронювання та про сам готель. Один готель має багато кімнат і так само багато працівників. Один працівник може взаємодіяти з певною кількістю гостей. В свою чергу один гість може забронювати декілька номерів.

**Опис сутностей предметної області**

Для побудови бази даних предметної області було виділено декілька сутностей , зображених на рисунку 1:

1)Кімнати(Rooms) , з атрибутами: тип номеру, номер(№) кімнати, кількість кімнат в номері, код готелю.

2) Готель(Hotel) , з атрибутами: код готелю, розташування, назва.

3)Працівник(Worker) , з атрибутами: код працівника, ім’я , прізвище,

код готелю.

4)Відвідувач(Costumer) , з атрибутами: код відвідувача, ім’я , прізвище, електронна адреса, код працівника.

5)Бронь(Booking) , з атрибутами: код бронювання , кількість , дата, код відвідувача.

**Опис зв’язків між сутностями предметної області**

Сутність «Готель» має зв’язок 1:N як до сутності «Кімнати» (адже в одному готелі знаходиться певна кількість номерів), так і до сутності «Працівник» (адже в одному готелі працює декілька людей різних посад).

Сутність «Працівник» має зв’язок 1:N по відношенню до сутності «Відвідувач», так як один працівник може обслуговувати певну кількість відвідувачів.

Сутність «Відвідувач» має зв’язок 1:N по відношенню до сутності «Бронь», тому що один відвідувач може забронювати декілька номерів різного типу на різні дати.

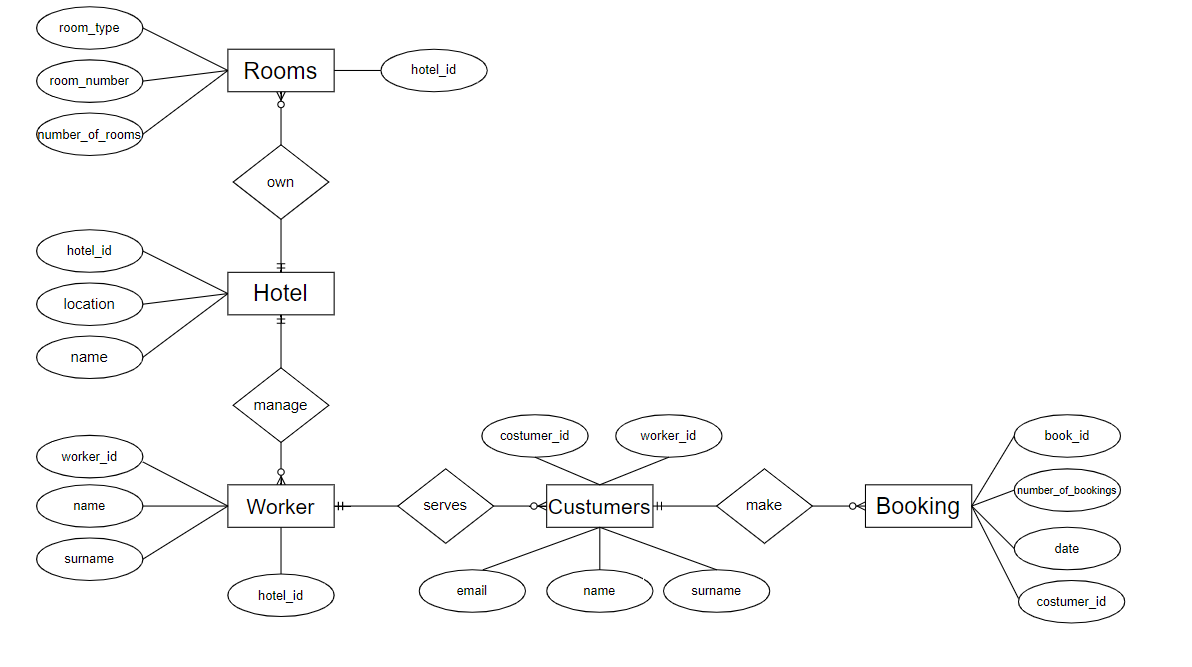
**Концептуальна модель предметної області “Готель”**

Рисунок 1 - ER-діаграма, побудована за нотацією Чена (інструмент: draw.io)

**Логічна модель (схема) БД “Готель”**

В логічній моделі (Рисунок 2):

1. Сутність “Кімнати” перетворена в таблицю “Rooms”.

2. Сутність “Готель” перетворена в таблицю “Hotel”.

3. Сутність “Працівник” перетворена в таблицю “Worker”.

4.Сутність “Відвідувач” перетворена в таблицю “Customers”.

5. Сутність “ Бронь ” перетворена в таблицю “Booking”.

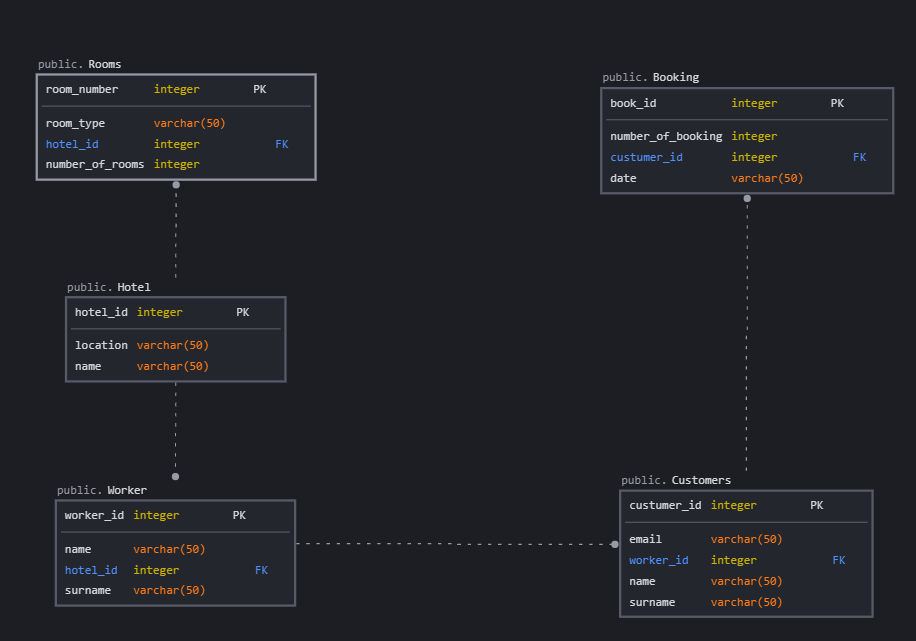


Рисунок 2. Схема бази даних (інструмент: sqldbm.com)

**Опис об’єктів бази даних у вигляді таблиці**

|  |  |  |
| --- | --- | --- |
| Сутність | Атрибут | Тип атрибуту |
| ***Hotel(****інформація про готель)* | ***Hotel\_id****:унікальний ідентифікатор готелю*  ***Location****:місце розташування готелю, не допускає NULL*  ***Name****:назва готелю, не допускає NULL* | **Integer(числовий)**  **Varchar(рядок)**  **Varchar(рядок)** |
| ***Rooms****(інформація про кімнати у готелі)* | ***Room\_number****: унікальний ідентифікатор кімнати*  ***Room\_type****: вид номеру, не допускає NULL*  ***Number\_of\_rooms****:кількість кімнат в номері, не допускає NULL*  ***Hotel\_id****:унікальний ідентифікатор готелю* | **Integer(числовий)**  **Varchar(рядок)**  **Integer(числовий)**  **Integer(числовий)** |
| ***Worker****(інформація про робітників готелю)* | ***Worker\_id:*** *унікальний ідентифікатор працівника*  ***Name:*** *ім’я працівника, не допускає NULL*  ***Surname:*** *прізвище працівника , не допускає NULL* | **Integer(числовий)**  **Varchar(рядок)**  **Varchar(рядок)** |
| ***Customers****(інформація про відвідувачів)* | ***Customer\_id:****унікальний ідентифікатор**відвідувача*  ***Email****: електронна пошта відвідувача, не допускає NULL*  ***Name:*** *ім’я відвідувача, не допускає NULL*  ***Surname:*** *прізвище відвідувача, не допускає NULL*  ***Worker\_id:*** *унікальний ідентифікатор працівника* | **Integer(числовий)**  **Varchar(рядок)**  **Varchar(рядок)**  **Varchar(рядок)**  **Integer(числовий)** |
| ***Booking(****інформація про бронювання номеру)* | ***Booking\_id****:унікальний ідентифікатор бронювання*  ***Number\_of\_booking:****кількість заброньованих**номерів, не допускає NULL*  ***Customer\_id:****унікальний ідентифікатор**відвідувача*  ***Date:****дата, не допускає NULL* | **Integer(числовий)**  **Integer(числовий)**  **Integer(числовий)**  **Varchar(рядок)** |

**Функціональні залежності для кожної таблиці**

Hotel:

Hotel\_id → name , location

Hotel\_id → name (назва готелю залежить від його коду)

Hotel\_id → location (розташування залежить від коду готелю)

Rooms:

Room\_number → room\_type, number\_of\_rooms, hotel\_id

Room\_number → room\_type (тип номеру залежить від його номеру(№))

Room\_number → number\_of\_rooms (кількість кімнат в номері залежить від номеру(№) готельного номера)

Room\_number → hotel\_id (код готелю залежить від номеру(№) готельного номера)

Worker:

Worker\_id → name, surname , hotel\_id

Worker\_id → name ( ім’я працівника залежить від його коду)

Worker\_id → surname(прізвище працівника залежить від його коду)

Worker\_id → hotel\_id(код готелю залежить від коду працівника)

Customers:

Customer\_id → email, name , surname , worker\_id

Customer\_id → email( email відвідувача залежить від його коду)

Customer\_id → name(ім’я відвідувача залежить від його коду)

Customer\_id → surname(прізвище відвідувача залежить від його коду)

Customer\_id → worker\_id(код працівника залежить від коду відвідувача)

Booking:

Booking\_id → number\_of\_booking, customer\_id, date

Booking\_id → number\_of\_booking( кількість бронювань залежить від коду бронювання)

Booking\_id → customer\_id(код відвідувача залежить від коду бронювання)

Booking\_id → date( дата залежить від коду бронювання )

**Структура БД “Готель”**

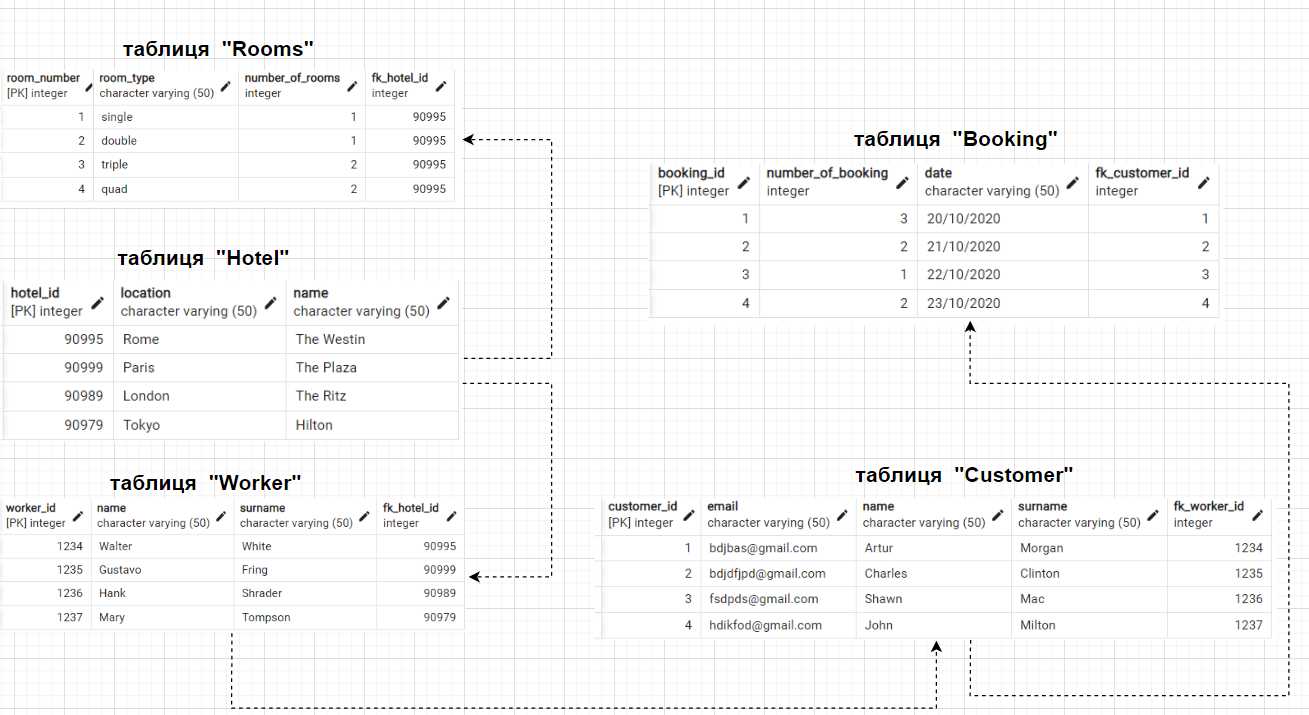


Рисунок 3. Структурна модель предметної області “Готель”(інструмент:draw.io)

Схема відповідає 1НФ, а саме:

1)Кожна таблиця повинна мати основний ключ (Primary key, PK): мінімальний набір атрибутів, які ідентифікують запис;

2)Кожен атрибут повинен мати лише одне значення, а не множину значень.

Схема відповідає 2НФ, а саме:

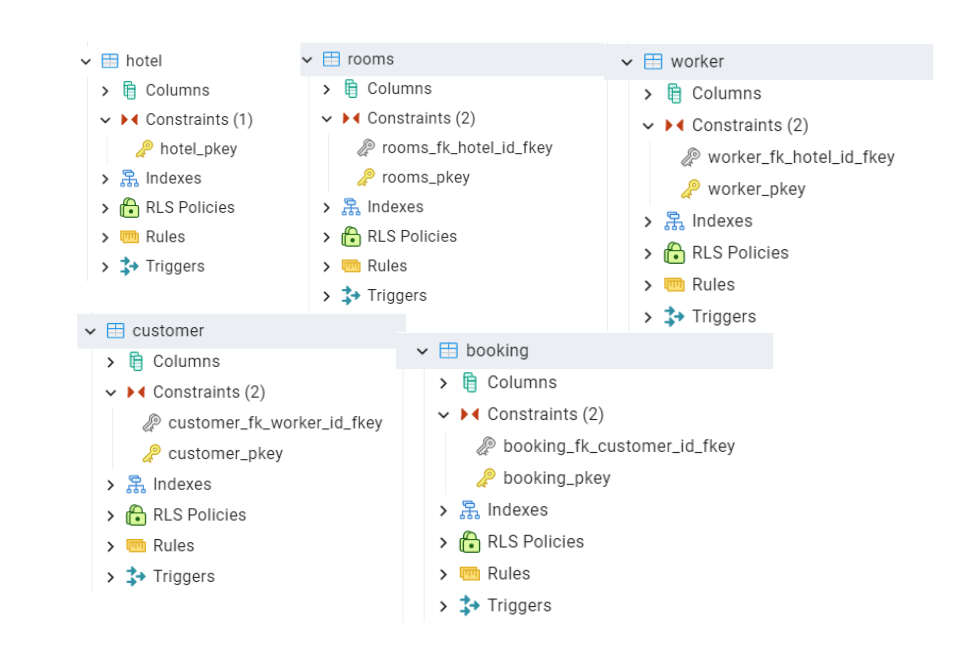
1)Схема повинна відповідати вимогам 1НФ;

2)Атрибути повинні повністю залежати від обох первинних атрибутів, а не від якогось окремого.

Схема відповідає 3НФ, а саме:

1)Схема повинна відповідати вимогам 2НФ;

2)Дані в таблиці повинні залежати винятково від основного ключа.



**Структура БД “Готель” в pgAdmin 4**

**Опис таблиць БД “Готель” в pgAdmin 4**

**CREATE** **TABLE** Hotel

(

hotel\_id **INTEGER** **PRIMARY** **KEY**,

location **VARCHAR**(50) **NOT** **NULL**,

name **VARCHAR**(50) **NOT** **NULL**

);

**INSERT** **INTO** Hotel(hotel\_id,location,name) **VALUES**

(90995,'Rome','The Westin'),

(90999,'Paris','The Plaza'),

(90989,'London','The Ritz'),

(90979,'Tokyo','Hilton');

**CREATE** **TABLE** Rooms

(

room\_number **INTEGER** **PRIMARY** **KEY**,

room\_type **VARCHAR**(50) **NOT** **NULL**,

number\_of\_rooms **INTEGER** **NOT** **NULL**,

fk\_hotel\_id **INTEGER** **REFERENCES** Hotel(hotel\_id) **NOT** **NULL**

);

**INSERT** **INTO** Rooms(room\_number,room\_type,fk\_hotel\_id,number\_of\_rooms) **VALUES**

(1,'single',90995,1),

(2,'double',90995,1),

(3,'triple',90995,2),

(4,'quad',90995,2);

**CREATE** **TABLE** Worker

(

worker\_id **INTEGER** **PRIMARY** **KEY**,

name **VARCHAR**(50) **NOT** **NULL**,

surname **VARCHAR**(50) **NOT** **NULL**,

fk\_hotel\_id **INTEGER** **REFERENCES** Hotel(hotel\_id) **NOT** **NULL**

);

**INSERT** **INTO** Worker(worker\_id,name,surname,fk\_hotel\_id) **VALUES**

(1234,'Walter','White',90995),

(1235,'Gustavo','Fring',90999),

(1236,'Hank','Shrader',90989),

(1237,'Mary','Tompson',90979);

**CREATE** **TABLE** Customer

(

customer\_id **INTEGER** **PRIMARY** **KEY**,

email **VARCHAR**(50) **NOT** **NULL**,

name **VARCHAR**(50) **NOT** **NULL**,

surname **VARCHAR**(50) **NOT** **NULL**,

fk\_worker\_id **INTEGER** **REFERENCES** Worker(worker\_id) **NOT** **NULL**

);

**INSERT** **INTO** Customer(customer\_id,email,name,surname,fk\_worker\_id) **VALUES**

(1,'bdjbas@gmail.com','Artur','Morgan',1234),

(2,'bdjdfjpd@gmail.com','Charles','Clinton',1235),

(3,'fsdpds@gmail.com','Shawn','Mac',1236),

(4,'hdikfod@gmail.com','John','Milton',1237);

**CREATE** **TABLE** Booking

(

booking\_id **INTEGER** **PRIMARY** **KEY**,

number\_of\_booking **INTEGER** **NOT** **NULL**,

**DATE** **VARCHAR**(50) **NOT** **NULL**,

fk\_customer\_id **INTEGER** **REFERENCES** Customer(customer\_id) **NOT** **NULL**

);

**INSERT** **INTO** Booking(booking\_id,number\_of\_booking,**DATE**,fk\_customer\_id) **VALUES**

(1,'3','20/10/2020',1),

(2,'2','21/10/2020',2),

(3,'1','22/10/2020',3),

(4,'2','23/10/2020',4)