

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

Бази даних та інформаційні системи

ЛАБОРАТОРНА РОБОТА №2

Тема: «Вивчення поняття домени бази даних. Створення таблиць бази даних»

Виконав:

Ст. Лук'янчук Денис

Група ПМІ-23

Тема: «Вивчення поняття домени бази даних. Створення таблиць бази даних».

Мета роботи: Ознайомлення з поняттям домен, його створенням і використанням. Створення таблиць бази даних відповідно до розробленої на лабораторній №1 концептуальної моделі “сутність-зв’язок” (ER-діаграми).

Завдання лабораторної роботи (14 варіант):

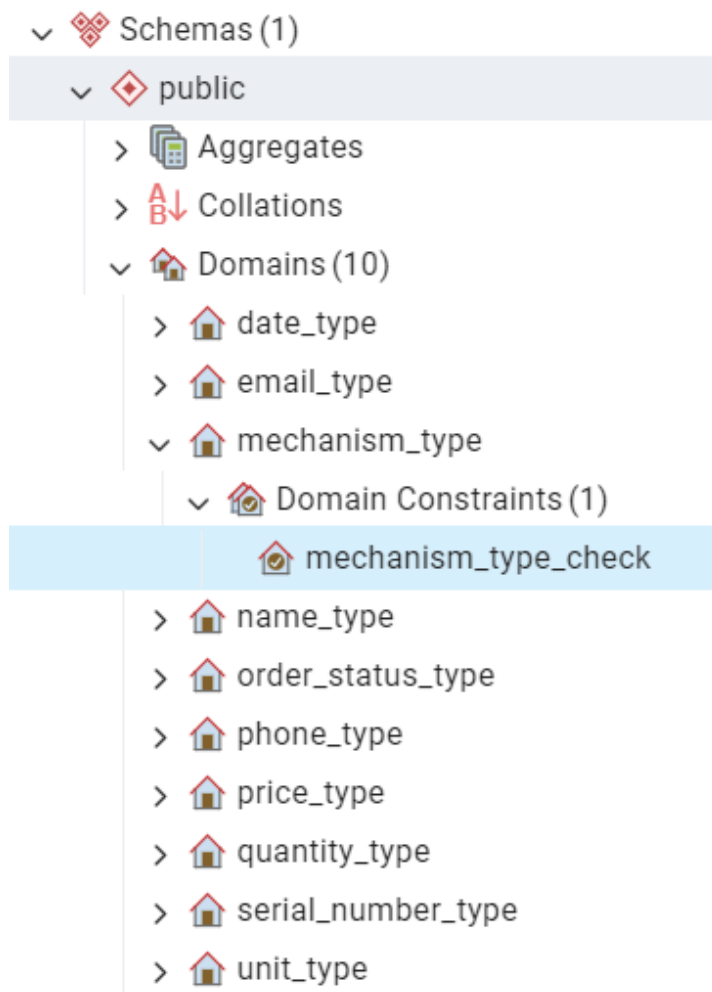
1. Опрацювати теоретичний матеріал.
2. Встановити сервер PostgreSQL.
3. Створити власну базу даних.
4. Створити домен(и).
5. Створити таблиці бази даних згідно розробленої на лабораторній роботі №1 власної концептуальної моделі “сутність-зв’язок” (ER-діаграми) з відображенням зв’язків між таблицями за допомогою первинних та зовнішніх ключів. Увести не менше 5-и кортежів даних в кожную таблицю.

Хід роботи

Створення Доменів:

Для забезпечення типізації даних із заданими обмеженнями створено 10 користувацьких доменів:

1. **name_type** - для назв, перевірка на непорожність і відповідність шаблону.
2. **email_type** - для адрес електронної пошти, перевірка на коректний формат і непорожність.
3. **phone_type** - для номерів телефону, перевірка на коректний формат і непорожність.
4. **price_type** - для цін, перевірка на значення ≥ 0 (тип DECIMAL(10, 2)).
5. **date_type** - для дат, перевірка на дату не пізніше поточної.
6. **serial_number_type** - для серійних номерів, перевірка на формат.
7. **mechanism_type** - для типів механізмів, перевірка на значення: 'Двигун', 'Гальма', 'Підвіска', 'Трансмісія', 'Рульове'.
8. **unit_type** - для одиниць виміру, перевірка на значення: 'см', 'кг', 'мм', 'м'.
9. **order_status_type** - для статусів замовлень, перевірка на значення: 'Новий', 'Обробляється', 'Доставлений'.
10. **quantity_type** - для кількостей, перевірка на значення ≥ 0 (тип INTEGER).

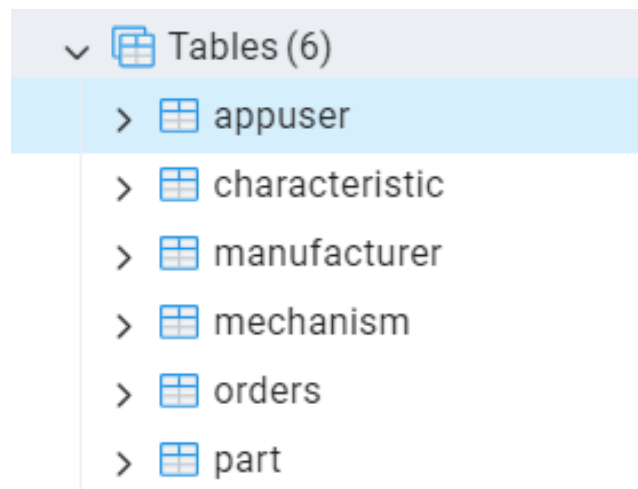


Створення Таблиць:

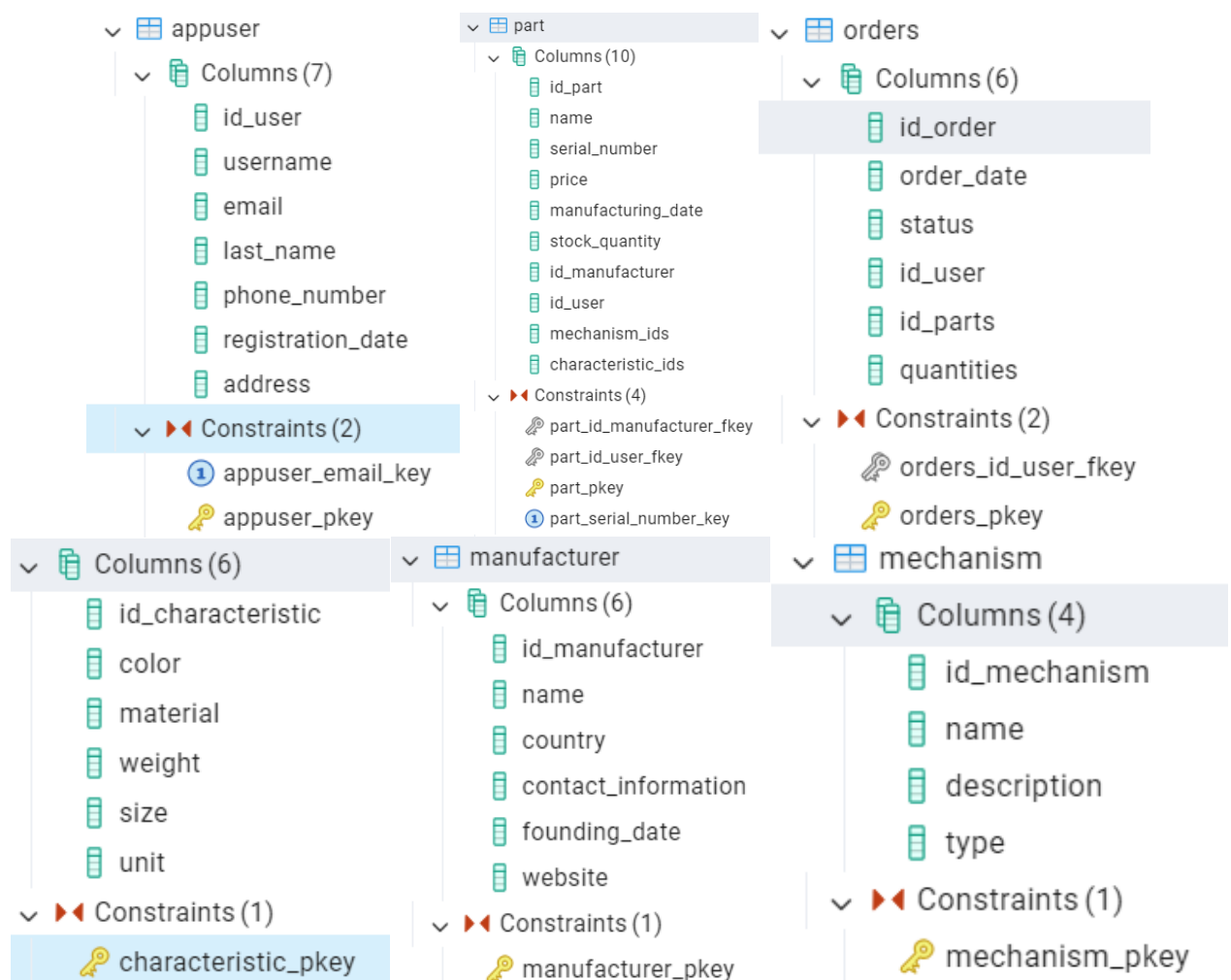
Після цього створено 6 таблиць із використанням доменів і налаштуванням зв'язків між ними:

1. Таблиця **Manufacturer** для зберігання даних про виробників із унікальним ідентифікатором, назвою, країною, контактною інформацією, датою заснування та сайтом.
2. Таблиця **Mechanism** для механізмів із ідентифікатором, назвою, описом і типом.
3. Таблиця **User** для даних про користувачів із унікальним ідентифікатором, ім'ям користувача, електронною поштою, прізвищем, номером телефону, датою реєстрації та адресою.
4. Таблиця **Characteristic** для характеристик із ідентифікатором, кольором, матеріалом, вагою, розміром і одиницею виміру.
5. Таблиця **Part** для запчастин із ідентифікатором, назвою, серійним номером, ціною, датою виробництва, кількістю на складі, ідентифікатором виробника, ідентифікатором користувача, масивами ідентифікаторів механізмів і характеристик.

6. Таблиця **Order** для замовлень із ідентифікатором, датою замовлення, статусом, ідентифікатором користувача, масивами ідентифікаторів запчастин і кількостей.



Для кожної таблиці встановлено первинні та зовнішні ключі для забезпечення зв'язків між таблицями, наприклад між Запчастина і Виробник, Запчастина і Користувач, Замовлення і Користувач. Створення таблиць перевірено через відображення їх у дереві об'єктів у pgAdmin із розгортанням розділів "Columns" і "Constraints".



Приклади заповнення таблиць:

	id_user [PK] integer	username character varying (100)	email character varying (255)	last_name character varying (100)	phone_number character varying (20)	registration_date date	address character varying (255)
1	1	Oleg	oleg@example.com	Petrenko	+380671234567	2025-03-09	Lenina St 1, Kyiv
2	2	Maria	maria@example.com	Ivanenko	+380671234568	2025-03-09	Shevchenko St 2, Lviv
3	3	Igor	igor@example.com	Sydorenko	+380671234569	2025-03-09	Franko St 3, Odesa
4	4	Anna	anna@example.com	Kovalenko	+380671234570	2025-03-09	Gogol St 4, Kharkiv
5	5	Pavlo	pavlo@example.com	Melnyk	+380671234571	2025-03-09	Pushkin St 5, Dnipro

	id_characteristic [PK] integer	color character varying (50)	material character varying (50)	weight numeric (10,2)	size character varying (20)	unit character varying (10)
1	1	Red	Metal	5.50	10x5	cm
2	2	Blue	Plastic	2.00	8x4	cm
3	3	Green	Wood	3.50	12x6	cm
4	4	Yellow	Glass	1.80	7x3	cm
5	5	White	Rubber	4.00	9x5	cm

	id_manufacturer [PK] integer	name character varying (100)	country character varying (100)	contact_information character varying (255)	founding_date date	website character varying (100)
1	1	Bosch	Germany	bosch@company.com	1886-11-15	www.bosch.com
2	2	Toyota	Japan	toyota@company.com	1937-08-28	www.toyota.com
3	3	Ford	USA	ford@company.com	1903-06-16	www.ford.com
4	4	Renault	France	renault@company.com	1899-02-25	www.renault.com
5	5	Skoda	Czech Republic	skoda@company.com	1895-04-18	www.skoda.com

Висновок: Під час виконання цієї лабораторної роботи я вперше застосував сервер PostgreSQL для розробки бази даних. На основі ER-діаграми, створеної раніше, реалізував таблиці та домени, розробивши структуру для торговельної системи, що складається з 6 таблиць і 10 доменів для моделювання виробників, механізмів, користувачів, характеристик, запчастин і замовлень. Перевірив правильність додавання даних через Query Tool, упевнившись у ефективності обмежень доменів (наприклад, перевірка формату номерів телефонів і дат) та зв'язків між таблицями (зовнішні ключі). У результаті створено робочу базу даних, заповнену тестовим набором даних.

Відповіді на контрольні питання

1. Наведіть приклади скалярних типів даних:

- Цілі числа (INTEGER, BIGINT, SMALLINT)
- Дробові числа (DECIMAL, NUMERIC, REAL, DOUBLE PRECISION)
- Рядки (TEXT, VARCHAR, CHAR)
- Дати та час (DATE, TIMESTAMP, TIME)
- Логічні значення (BOOLEAN)

2. Які типи даних використовують для оголошення грошових даних:

- DECIMAL(p, s)
- NUMERIC(p, s)

- MONEY (в PostgreSQL)
- 3. Дайте визначення поняття домена.** Домен – це користувацький тип даних, який визначає допустимі значення для стовпця, використовуючи обмеження (CHECK).
- 4. Підмова даних є сукупністю операторів для роботи з базами даних, таких як DDL (Data Definition Language) та DML (Data Manipulation Language).**
- 5. Які оператори відносяться до мови оголошення даних (DDL):**
- CREATE (створення таблиць, схем, баз даних)
 - ALTER (зміна структури таблиць)
 - DROP (видалення об'єктів)
- 6. Які оператори відносяться до мови обробки даних (DML):**
- SELECT (вибірка даних)
 - INSERT (додавання даних)
 - UPDATE (оновлення даних)
 - DELETE (видалення даних)
- 7. Напишіть оператор створення таблиці Товар зі стовпчиками Назва і Кількість.**
- ```
CREATE TABLE Товар (Назва VARCHAR(255), Кількість INT);
```
- 8. Змініть таблицю Товар додавши до неї стовпчик Ціна.**
- ```
ALTER TABLE Товар ADD COLUMN Ціна DECIMAL(10,2);
```
- 9. Напишіть оператор додавання у таблицю Товар рядка з назвою товару “Деталь1” кількістю 10 і ціною 200.**
- ```
INSERT INTO Товар (Назва, Кількість, Ціна) VALUES ('Деталь1', 10, 200);
```
- 10. Напишіть для таблиці Товар оператор зменшення ціни товару на 50% для товару “Деталь1”.**
- ```
UPDATE Товар SET Ціна = Ціна * 0.5 WHERE Назва = 'Деталь1';
```
- 11. Напишіть для таблиці Товар оператор видалення рядків з товарами яких залишилось 0.**
- ```
DELETE FROM Товар WHERE Кількість = 0;
```