

**Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”**

Факультет прикладної математики

**Кафедра системного програмування і 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: Зорєв М.А

Група: KB-11

Оцінка:

Київ - 2023

Постановка задачі

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.

Вимоги до ER-моделі

1. Сутності моделі предметної галузі мають містити зв'язки типу 1:N або N:M.
2. Кількість сутностей у моделі – 3-4. Кількість атрибутів у кожній сутності: від двох до п'яти
3. Передбачити наявність зв'язку з атрибутом.
4. Для побудови ER-діаграм використовувати одну із нотацій: Чена, “Пташиної лапки (Crow's foot)”, UML.

Предметна галузь лабораторної роботи

Система управління замовленнями їжі.

Посилання на репозиторій Github із файлами лабораторної роботи

https://github.com/carcharodon256/DB_LABS.git

Контакт у месенджері Telegram для швидкого зв'язку

@whiteShark816

Вимоги до звіту лабораторної роботи

У звіті щодо пункту №1 завдання має бути:

- перелік сутностей з описом їх призначення;
- графічний файл розробленої моделі «сутність-зв'язок»;
- назва нотації.

У звіті щодо пункту №2 завдання має бути:

- опис процесу перетворення (наприклад, “сутність А було перетворено у таблицю А, а зв'язок R (M:N) зумовив появу додаткової таблиці R1 тощо);
- схему бази даних у графічному вигляді **з назвами таблиць (!) та зв'язками між ними, а також необхідно намалювати перетворену ER-діаграму у ТАБЛИЦІ БД! Це означає, що тут не може бути зв'язку N:M, мають бути позначені первинні та зовнішні ключі, обмеження NOT NULL та UNIQUE і внести типи даних атрибутів.**

У звіті щодо пункту №3 завдання має бути:

- пояснення (**обґрунтування!**) щодо відповідності схеми бази даних нормальним формам НФ1, НФ2 та НФ3. Пояснення **полягає у наведенні функціональних залежностей**, що демонструють висновки. У випадку невідповідності надати опис необхідних змін у схемі;
- У випадку проведення змін у схемі бази даних надати оновлену версію схеми, інакше - не наводити схему.

У звіті щодо пункту №4 завдання має бути:

- навести копії екрану з pgAdmin4, що відображають назви, типи та обмеження на стовпці (доступне у закладці “Columns” та “Constraints” властивостей “Properties” таблиць дерева об'єктів у pgAdmin4);
- навести копії екрану з pgAdmin4, що відображають вміст таблиць бази даних у PostgreSQL. Таблиці на зображенні обов'язково **повинні мати назву!**

Виконання лабораторної роботи

1. Розробка моделі “сутність-зв’язок обраної предметної галузі”

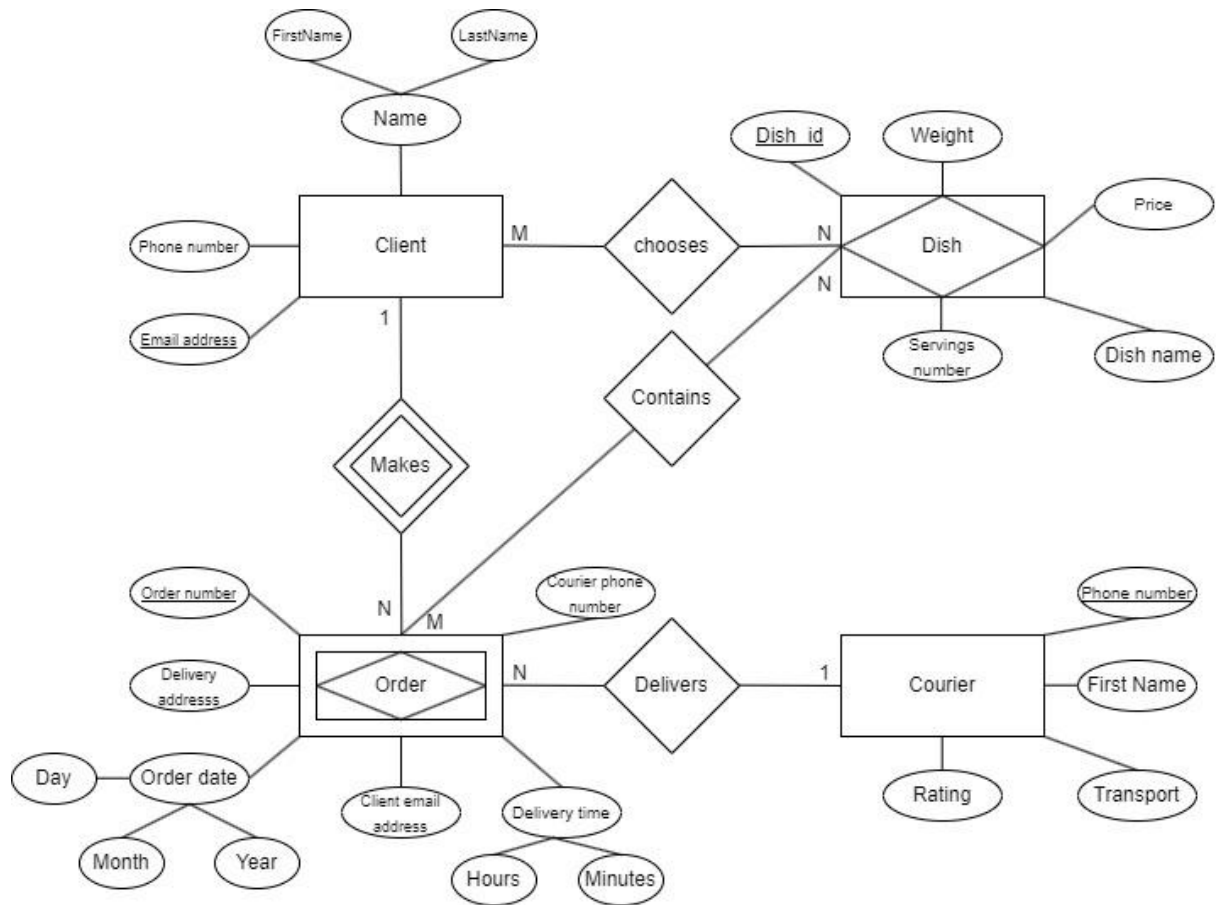
Перелік основних сутностей предметної галузі та опис їх призначення	
Назва сутності	Призначення
Client	Зберігає дані про клієнта, який замовив страву: ім’я, прізвище, адресу електронної пошти, номер телефону
Dish	Зберігає дані про страви, які замовив клієнт: ідентифікатор стави, назву, масу, ціну та обрану кількість порцій
Order	Зберігає дані про замовлення, яке оформив клієнт: номер замовлення, адресу доставки, дату та час доставки, електронну адресу клієнта, який оформив замовлення та номер телефону кур’єра, який доставляє дане замовлення
Courier	Зберігає дані про кур’єра, який доставляє замовлення до клієнтів: номер телефону кур’єра, ім’я кур’єра, вид транспорту, яким він користується та рейтинг кур’єра

Опис зв'язків між сутностями

Опис зв'язку	Вид зв'язку	Пояснення
Clients CHOOSE dishes	M : N	Як один клієнт може замовити багато страв, так і одна страва може бути замовлена багатьма клієнтами
Order CONTAIN dishes	M : N	Одне замовлення може містити багато страв. Так само одна страва може міститися в багатьох замовленнях
Client MAKES orders	1 : N	Один клієнт може оформити декілька замовлень, але одне замовлення може бути оформлене тільки одним клієнтом
Courier DELIVERS orders	1 : N	Один кур'єр може доставляти багато замовлень, але одне замовлення може бути доставлене лише одним кур'єром

Діаграма “сутність-зв’язок”

Використана нотація: нотація Пітера Чена

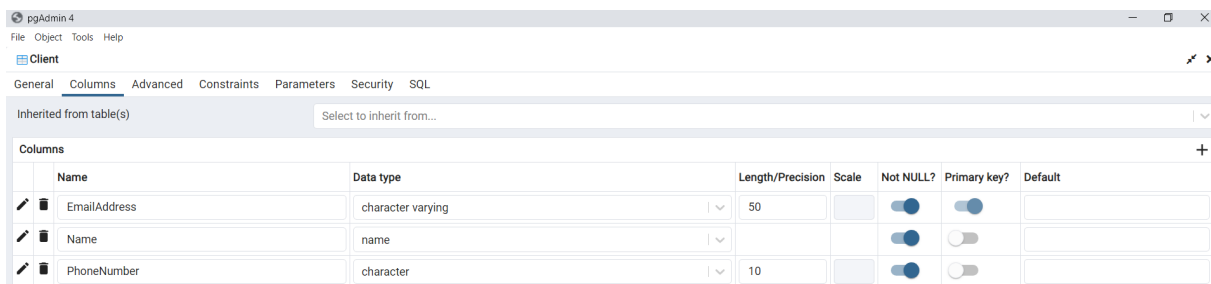


2. Перетворення ER-діаграми у схему бази даних

Наявні у ER-діаграмі сутності були перетворені на однойменні таблиці схеми бази даних.

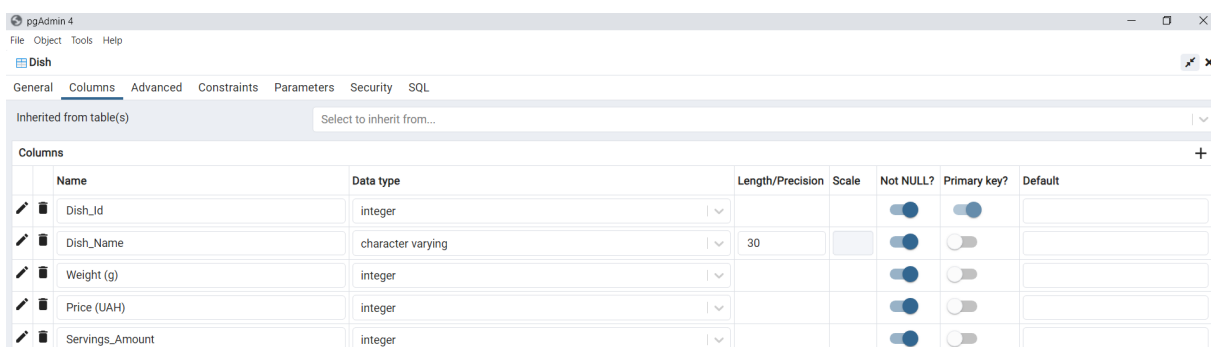
Створення основних таблиць

- Створення таблиці **Client**



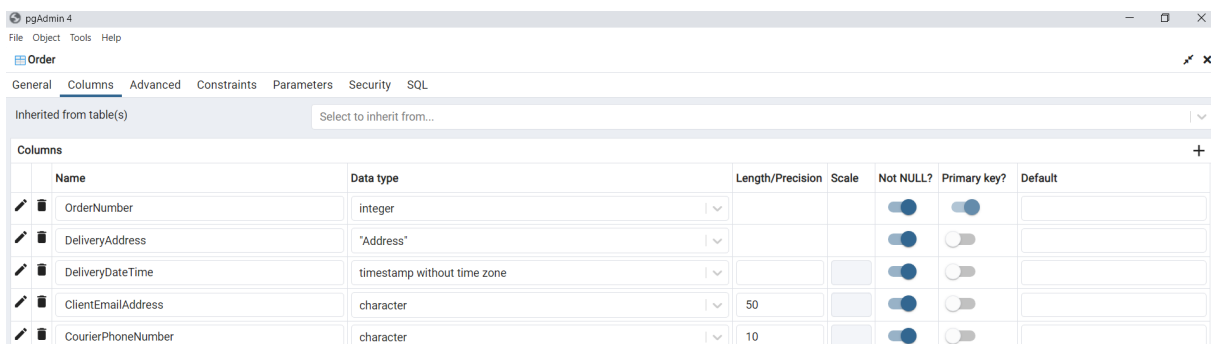
Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
EmailAddress	character varying	50		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Name	name			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
PhoneNumber	character	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

- Створення таблиці **Dish**



Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
Dish_Id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Dish_Name	character varying	30		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Weight (g)	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Price (UAH)	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Servings_Amount	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

- Створення таблиці **Order**



Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
OrderNumber	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DeliveryAddress	text			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
DeliveryDateTime	timestamp without time zone			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
ClientEmailAddress	character	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
CourierPhoneNumber	character	10		<input checked="" type="checkbox"/>	<input type="checkbox"/>	

- Створення таблиці **Courier**

The screenshot shows the pgAdmin 4 interface with the 'Courier' table selected. The 'Columns' tab is active, displaying a table with the following columns:

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
PhoneNumber	character	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
FirstName	character varying	30		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
TransportKind	"Transport"			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
CourierRating	"Rating"			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Наявність зв'язку типу “багато до багатьох” між парами сутностей “клієнт-страва” та “замовлення-страва” зумовила створення двох додаткових таблиць: **Client-Dish** та **Order-Dish**.

Створення додаткових таблиць

- Створення таблиці **Client-Dish**

The screenshot shows the pgAdmin 4 interface for creating a new table. The 'Columns' tab is active, displaying a table with the following columns:

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
EmailAddress	character varying	50		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DishId	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

At the bottom of the interface, there are buttons for 'Close', 'Reset', and 'Save'.

- Створення зовнішніх ключів таблиці **Client-Dish**

Client-Dish

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Primary Key

Foreign Key

Check

Unique

Exclude

Name	Columns	Referenced Table
DishId_FK	(DishId) -> (Dish_Id)	public.Dish
EmailAddress_FK	(EmailAddress) -> (EmailAddress)	public.Client

General

Definition

Columns

Action

Local	Referenced	Referenced Table
EmailAddress	EmailAddress	public.Client

i

?

Close

Reset

Save

- Створення таблиці **Order-Dish**

order-Dish

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
OrderNumber	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DishId	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

i

?

Close

Reset

Save

- Створення зовнішніх ключів таблиці **Order-Dish**

order-Dish

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Primary Key



Foreign Key

Check

Unique

Exclude

+

	Name	Columns	Referenced Table
 	OrderNumber_FK		




General

Definition

Columns


Action


Columns

Local column	 OrderNumber <div>✕ ▾</div>
References	 public."Order" <div>✕ ▾</div>
Referencing	 OrderNumber <div>✕ ▾</div>

Add


Local	Referenced	Referenced Table
-------	------------	------------------





✕ Close

↺ Reset

 Save

Реалізація обов'язкових зв'язків 1 : N

- Реалізація зв'язку 1 : N між таблицями **Client** та **Order**

Order

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Primary KeyForeign KeyCheckUniqueExclude

Name	Columns	Referenced Table
FK_ClientEmailAddress		

GeneralDefinitionColumnsAction

Columns

Local column

ClientEmailAddress

×

▼

References

public."Client"

×

▼

Referencing

EmailAddress

×

▼

Add

Local	Referenced	Referenced Table
-------	------------	------------------

CloseResetSave

- Реалізація зв'язку 1 : N між таблицями **Courier** та **Order**

Order

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

Primary KeyForeign KeyCheckUniqueExclude

	Name	Columns	Referenced Table
	FK_ClientEmailAddress	(ClientEmailAddress) -> (EmailAddress)	public.Client
	FK_ClientPhoneNumber		

GeneralDefinitionColumnsAction

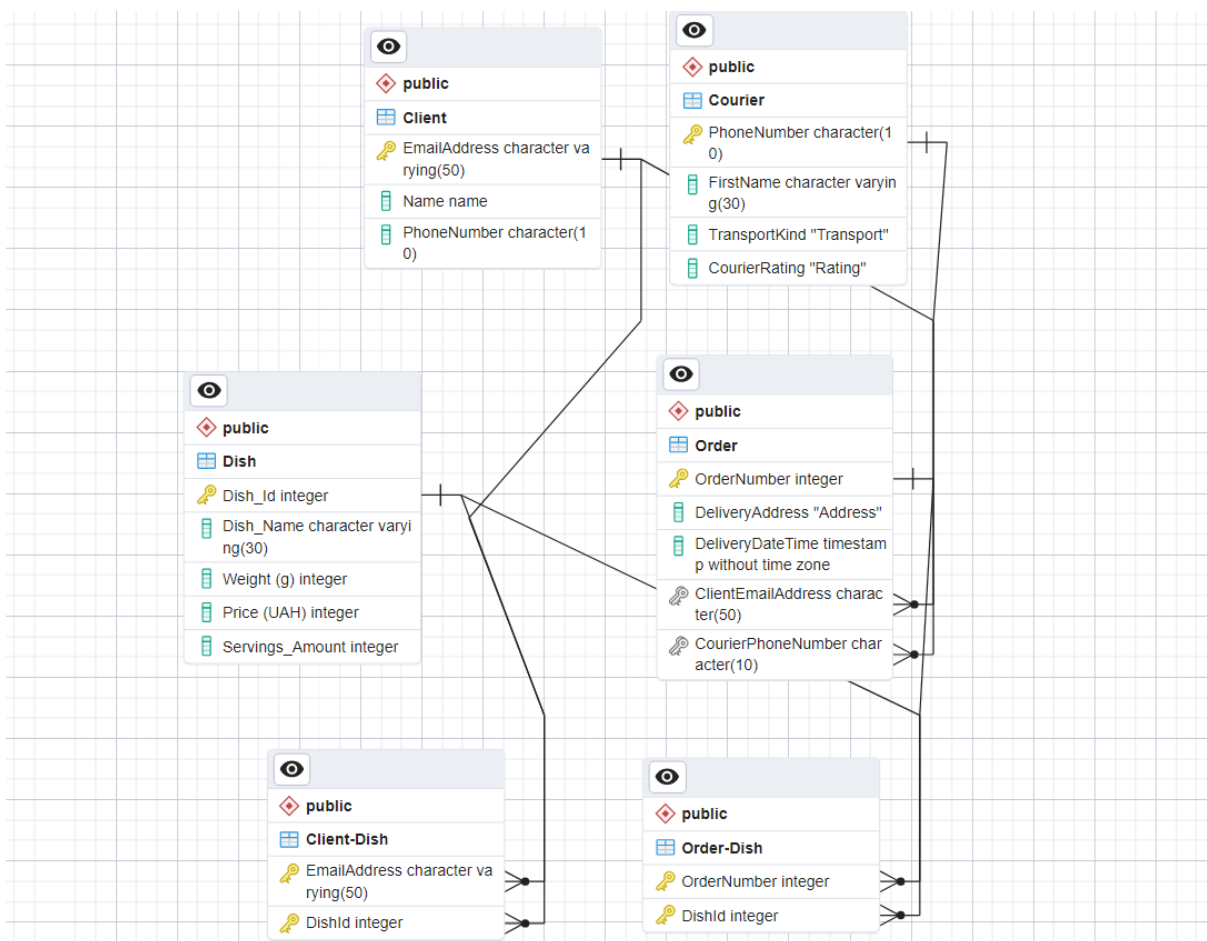
Columns

Local column	CourierPhoneNumber	
References	public."Courier"	
Referencing	PhoneNumber	

Add

Close Reset Save

ER-діаграма, згенерована утилітою pgAdmin4 на основі створених таблиць та зв'язків



3. Нормалізація схеми бази даних

- Перевірка відповідності першій нормальній формі (НФ1)

Кожна з 4-х основних таблиць відповідає вимогам НФ1. Кожне значення є атомарним (значення складених атрибутів також можна вважати атомарними, оскільки згідно вимог предметної галузі, не буде потреби у зверненні окремо до кожного простого атрибуту, що є частиною складеного атрибуту (наприклад, до значення атрибуту “Назва вулиці” складеного атрибуту “Адреса доставки”). Відсутність повторень рядків у одній таблиці забезпечується наявністю первинного ключа у кожній таблиці.

- Перевірка відповідності другій нормальній формі (НФ2)

Кожна з основних таблиць схеми бази даних відповідає вимогам НФ1, а також у кожній з цих таблиць наявний первинний ключ (див. попередній абзац поточного розділу). Крім того, для всіх таблиць виконується й інша умова їх відповідності НФ2 - всі атрибути кожної таблиці, що не є первинними ключами або їх частинами залежать від первинного ключа в цілому, а не від його окремих складових. Останнє не є можливим, оскільки у кожній з таблиць схеми бази даних є лише прості (утворені одним атрибутом) ключі.

- Перевірка відповідності третій нормальній формі (НФ3)

Кожна з основних таблиць відповідає вимогам НФ2. Перевіримо чи відповідає вона правилу НФ3: кожний неключовий атрибут має нетранзитивно залежати лише від первинного ключа.

Розглянемо функціональні залежності таблиці **Client**:

EmailAddress → PhoneNumber

Name → PhoneNumber

Розглянемо функціональні залежності таблиці **Dish**:

DishId → Weight

DishId → DishName

DishId → Price

DishId → Servings amount

Розглянемо функціональні залежності таблиці **Order**:

DeliveryAddress → OrderNumber

DeliveryDateTime → OrderNumber

ClientEmailAddress → OrderNumber

CourierPhoneNumber → OrderNumber

Розглянемо функціональні залежності таблиці **Courier**:

FirstName → PhoneNumber

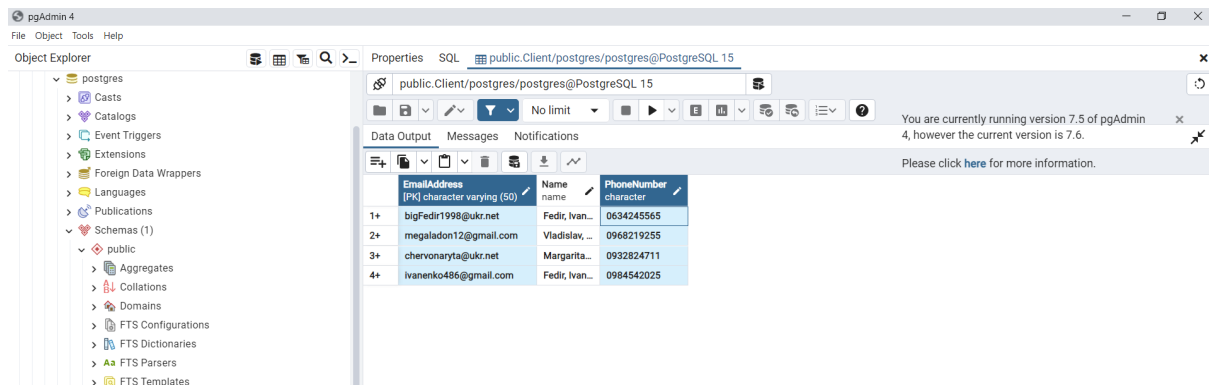
TransportKind → PhoneNumber

CourierRating → PhoneNumber

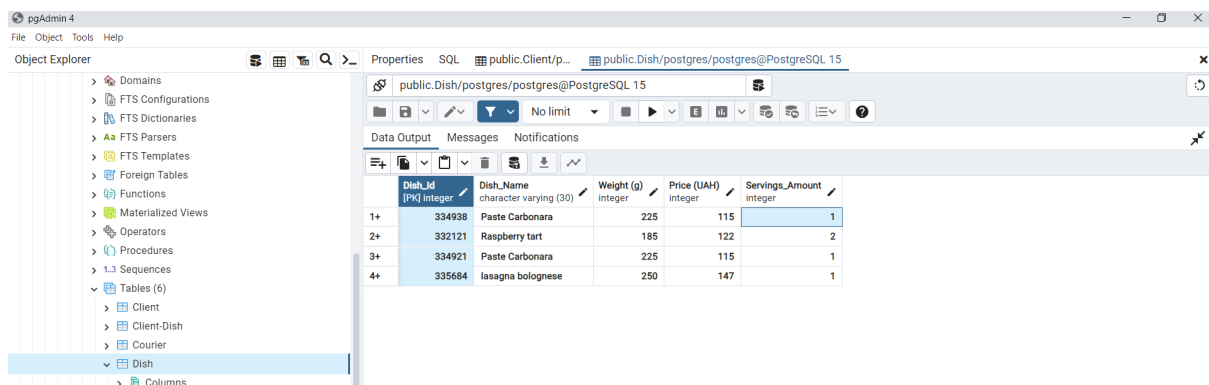
Неключові атрибути кожної з таблиць нетранзитивно залежать лише від її первинного ключа. Отже, можна стверджувати, що кожна з таблиць відповідає вимогам НФЗ.

4. Внесення рядків до таблиць бази даних

- Внесення даних до таблиці **Client**



- Внесення даних до таблиці **Dish**



- Внесення даних до таблиці **Order**

	OrderNumber [PK] integer	DeliveryAddress "Address"	DeliveryDateTime timestamp without time zone	ClientEmailAddress character	CourierPhoneNumber character
1+	1073380	Radunska, 6, 201	2023-09-26 10:00:00	chervonaryta@ukr.net	0639813340
2+	1073325	Gradynska, 1, 22	2023-09-26 10:30:00	ivanenko486@gmail...	0639813340
3+	1049684	Ivana Gontl, 25, 1...	2023-09-26 10:00:00	megaladon12@gmail...	0983565225
4+	1049621	Olexandri Exter, 1...	2023-09-26 09:45:00	bigFedir1998@ukr.net	0639813340

- Внесення даних до таблиці **Courier**

	PhoneNumber [PK] character	FirstName character varying (30)	TransportKind "Transport"	CourierRating "Rating"
1+	0639813340	Oleg	Bike	4.5
2+	0631234111	Serhii	Car	5
3+	0972273304	Michael	Motorbike	4
4+	0983565225	Oleg	Motorbike	5

- Внесення даних до додаткової таблиці **Client-Dish**

	EmailAddress [PK] character varying (50)	Dishid [PK] integer
1+	bigFedir1998@ukr.net	332121
2+	chervonaryta@ukr.net	335684
3+	ivanenko486@gmail.com	334921
4+	megaladon12@gmail.com	334938

- Внесення даних до таблиці **Order-Dish**

