***Лабораторна робота 4: Проектування фізичної моделі бази даних***

Група: АІ-243

Здобувач: Гаврилов О.В.

Варіант: 22

**Мета:**

Навчити студентів перетворювати логічну модель у фізичну модель бази даних, з подальшою реалізацією цієї моделі у конкретній СУБД.

**1) Завдання:**

1. Аналіз ER-діаграми:

*–* Переглянути ER-діаграму, створену в попередній лабораторній роботі.

1. Визначення типів даних:

*–* Визначити типи даних для атрибутів кожної сутності, відповідно до вимог СУБД

PostgreSQL.

1. Реалізація схеми бази даних у СУБД:

*–* Створити фізичну схему бази даних у СУБД, написавши SQL-запити для створення таблиць, зв'язків між ними, обмежень, а також для введення початкових даних.

1. Тестування бази даних:

*–* Виконати кілька SQL-запитів для перевірки коректності функціонування бази даних (наприклад, SELECT, INSERT, UPDATE, DELETE).

2) ***Предметна область:***

*Необхідно розробити базу даних для співробітників та клієнтів ресторану, яка повинна містити наступну інформацію:*

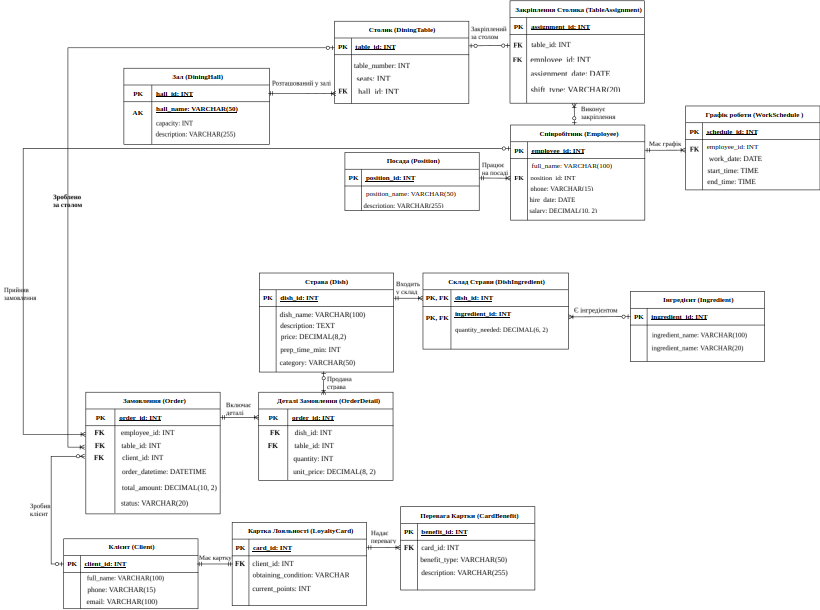
*– персонал ресторану та їх графік роботи;*

*– меню з зазначенням основних інгредієнтів, часу приготування та вартості;*

*– характеристика залів, столиків та їх закріплення за персоналом; – карти постійних клієнтів, умови їх отримання та переваги.*

*Крім того, структура бази даних повинна надавати можливість зберігати іншу інформацію, що, на думку студента, відноситься до даної предметної області й завдань, розв'язуваним розроблюваною системою.*

***3) Копія логічної моделі, яку створену в лабораторній роботі №3:***

**

***4) SQL-код створення об’єктів (домени, послідовності, UDT, таблиці) бази даних. Порядок створення об’єктів повинен бути збережений. Обов’язково прописати всі зв’язки між таблицями, ключі, обмеження.***

|  |
| --- |
| --- ##################################################################  -- 1. DDL: ВИДАЛЕННЯ ТА СТВОРЕННЯ ТАБЛИЦЬ  -- Використовуємо DROP TABLE IF EXISTS CASCADE для уникнення помилок типу "уже существует".  -- ##################################################################  DROP TABLE IF EXISTS Деталі\_замовлення CASCADE;  DROP TABLE IF EXISTS Замовлення CASCADE;  DROP TABLE IF EXISTS Закріплення CASCADE;  DROP TABLE IF EXISTS Перевага\_з\_картки CASCADE;  DROP TABLE IF EXISTS Картка\_лояльності CASCADE;  DROP TABLE IF EXISTS Столик CASCADE;  DROP TABLE IF EXISTS Склад\_Страви CASCADE;  DROP TABLE IF EXISTS Співробітник CASCADE;  DROP TABLE IF EXISTS Страва CASCADE;  DROP TABLE IF EXISTS Бенефіт CASCADE;  DROP TABLE IF EXISTS Зал CASCADE;  DROP TABLE IF EXISTS Клієнт CASCADE;  DROP TABLE IF EXISTS Інгредієнт CASCADE;  DROP TABLE IF EXISTS Графік\_роботи CASCADE;  DROP TABLE IF EXISTS Посада CASCADE;  -- Базові таблиці  CREATE TABLE Посада (  position\_id INT PRIMARY KEY,  position\_name VARCHAR(50) NOT NULL UNIQUE,  description VARCHAR(255)  );  CREATE TABLE Графік\_роботи (  schedule\_id INT PRIMARY KEY,  work\_date DATE NOT NULL,  start\_time TIME NOT NULL,  end\_time TIME NOT NULL,  UNIQUE (work\_date, start\_time, end\_time)  );  CREATE TABLE Інгредієнт (  ingredient\_id INT PRIMARY KEY,  ingredient\_name VARCHAR(100) NOT NULL UNIQUE  );  CREATE TABLE Клієнт (  client\_id INT PRIMARY KEY,  full\_name VARCHAR(100) NOT NULL,  phone VARCHAR(15) UNIQUE,  email VARCHAR(100) UNIQUE  );  CREATE TABLE Зал (  hall\_id INT PRIMARY KEY,  hall\_name VARCHAR(30) NOT NULL UNIQUE,  capacity INT,  description VARCHAR(255)  );  CREATE TABLE Бенефіт (  benefit\_id INT PRIMARY KEY,  benefit\_type VARCHAR(50) NOT NULL UNIQUE,  description VARCHAR(255)  );  CREATE TABLE Страва (  dish\_id INT PRIMARY KEY,  dish\_name VARCHAR(100) NOT NULL UNIQUE,  description TEXT,  price DECIMAL(8,2) NOT NULL CHECK (price >= 0),  prep\_time\_min INT CHECK (prep\_time\_min >= 0),  category VARCHAR(50)  );  -- Залежні таблиці  CREATE TABLE Співробітник (  employee\_id INT PRIMARY KEY,  position\_id INT NOT NULL,  schedule\_id INT,  full\_name VARCHAR(100) NOT NULL,  address VARCHAR(150),  hired\_date DATE NOT NULL,  salary DECIMAL(10,2) CHECK (salary >= 0),  CONSTRAINT FK\_Employee\_Position FOREIGN KEY (position\_id) REFERENCES Посада(position\_id) ON DELETE RESTRICT ON UPDATE CASCADE,  CONSTRAINT FK\_Employee\_Schedule FOREIGN KEY (schedule\_id) REFERENCES Графік\_роботи(schedule\_id) ON DELETE SET NULL ON UPDATE CASCADE  );  CREATE TABLE Склад\_Страви (  dish\_ingredient\_id INT PRIMARY KEY,  dish\_id INT NOT NULL,  ingredient\_id INT NOT NULL,  quantity\_needed DECIMAL(6,2) NOT NULL CHECK (quantity\_needed > 0),  CONSTRAINT FK\_DishIngredient\_Dish FOREIGN KEY (dish\_id) REFERENCES Страва(dish\_id) ON DELETE CASCADE ON UPDATE CASCADE,  CONSTRAINT FK\_DishIngredient\_Ingredient FOREIGN KEY (ingredient\_id) REFERENCES Інгредієнт(ingredient\_id) ON DELETE RESTRICT ON UPDATE CASCADE,  CONSTRAINT UQ\_DishIngredient UNIQUE (dish\_id, ingredient\_id)  );  CREATE TABLE Столик (  table\_id INT PRIMARY KEY,  hall\_id INT NOT NULL,  table\_number INT NOT NULL,  seats INT CHECK (seats > 0),  CONSTRAINT FK\_Table\_Hall FOREIGN KEY (hall\_id) REFERENCES Зал(hall\_id) ON DELETE RESTRICT ON UPDATE CASCADE,  CONSTRAINT UQ\_Table\_Hall\_Number UNIQUE (hall\_id, table\_number)  );  CREATE TABLE Картка\_лояльності (  card\_id INT PRIMARY KEY,  client\_id INT NOT NULL UNIQUE,  obtaining\_condition VARCHAR(100),  current\_points INT DEFAULT 0 CHECK (current\_points >= 0),  CONSTRAINT FK\_LoyaltyCard\_Client FOREIGN KEY (client\_id) REFERENCES Клієнт(client\_id) ON DELETE CASCADE ON UPDATE CASCADE  );  CREATE TABLE Закріплення (  assignment\_id INT PRIMARY KEY,  table\_id INT NOT NULL,  employee\_id INT NOT NULL,  assignment\_date DATE NOT NULL,  shift\_time VARCHAR(30),  CONSTRAINT FK\_Assignment\_Table FOREIGN KEY (table\_id) REFERENCES Столик(table\_id) ON DELETE RESTRICT ON UPDATE CASCADE,  CONSTRAINT FK\_Assignment\_Employee FOREIGN KEY (employee\_id) REFERENCES Співробітник(employee\_id) ON DELETE RESTRICT ON UPDATE CASCADE,  CONSTRAINT UQ\_Assignment\_Table\_Date UNIQUE (table\_id, assignment\_date, shift\_time)  );  CREATE TABLE Перевага\_з\_картки (  card\_id INT NOT NULL,  benefit\_id INT NOT NULL,  PRIMARY KEY (card\_id, benefit\_id),  CONSTRAINT FK\_CardBenefit\_Card FOREIGN KEY (card\_id) REFERENCES Картка\_лояльності(card\_id) ON DELETE CASCADE ON UPDATE CASCADE,  CONSTRAINT FK\_CardBenefit\_Benefit FOREIGN KEY (benefit\_id) REFERENCES Бенефіт(benefit\_id) ON DELETE RESTRICT ON UPDATE CASCADE  );  CREATE TABLE Замовлення (  order\_id INT PRIMARY KEY,  employee\_id INT NOT NULL,  client\_id INT,  table\_id INT,  order\_datetime TIMESTAMP NOT NULL, -- ВИПРАВЛЕНО: DATETIME замінено на TIMESTAMP  total\_amount DECIMAL(10,2) NOT NULL CHECK (total\_amount >= 0),  status VARCHAR(30),  CONSTRAINT FK\_Order\_Employee FOREIGN KEY (employee\_id) REFERENCES Співробітник(employee\_id) ON DELETE RESTRICT ON UPDATE CASCADE,  CONSTRAINT FK\_Order\_Client FOREIGN KEY (client\_id) REFERENCES Клієнт(client\_id) ON DELETE SET NULL ON UPDATE CASCADE,  CONSTRAINT FK\_Order\_Table FOREIGN KEY (table\_id) REFERENCES Столик(table\_id) ON DELETE SET NULL ON UPDATE CASCADE  );  CREATE TABLE Деталі\_замовлення (  order\_id INT NOT NULL,  dish\_id INT NOT NULL,  table\_id INT,  quantity INT NOT NULL CHECK (quantity > 0),  unit\_price DECIMAL(8,2) NOT NULL CHECK (unit\_price >= 0),  PRIMARY KEY (order\_id, dish\_id),  CONSTRAINT FK\_OrderDetail\_Order FOREIGN KEY (order\_id) REFERENCES Замовлення(order\_id) ON DELETE CASCADE ON UPDATE CASCADE,  CONSTRAINT FK\_OrderDetail\_Dish FOREIGN KEY (dish\_id) REFERENCES Страва(dish\_id) ON DELETE CASCADE ON UPDATE CASCADE, -- ВИПРАВЛЕНО: Додано ON DELETE CASCADE  CONSTRAINT FK\_OrderDetail\_Table FOREIGN KEY (table\_id) REFERENCES Столик(table\_id) ON DELETE SET NULL ON UPDATE CASCADE  ); |

***5) SQL-код додавання записів в таблиці. Як мінімум по 10 кортежів в кожну таблицю. Порядок заповнення таблиць повинен бути збережений.***

|  |
| --- |
| --- ##################################################################  -- 2. DML: ДОДАВАННЯ ЗАПИСІВ (INSERT)  -- ##################################################################  -- Посада  INSERT INTO Посада (position\_id, position\_name, description) VALUES  (1, 'Офіціант', 'Обслуговування клієнтів'), (2, 'Бармен', 'Приготування напоїв'),  (3, 'Кухар', 'Приготування гарячих страв'), (4, 'Адміністратор', 'Керування бронюваннями'),  (5, 'Посудомийник', 'Миття посуду'), (6, 'Менеджер залу', 'Контроль офіціантів'),  (7, 'Сомельє', 'Консультація щодо вина'), (8, 'Пекар', 'Приготування хліба'),  (9, 'Старший кухар', 'Керівництво кухнею'), (10, 'Технічний персонал', 'Обслуговування');  -- Графік\_роботи  INSERT INTO Графік\_роботи (schedule\_id, work\_date, start\_time, end\_time) VALUES  (1, '2025-10-01', '09:00:00', '17:00:00'), (2, '2025-10-01', '17:00:00', '01:00:00'),  (3, '2025-10-02', '09:00:00', '17:00:00'), (4, '2025-10-02', '17:00:00', '01:00:00'),  (5, '2025-10-03', '08:00:00', '16:00:00'), (6, '2025-10-03', '16:00:00', '00:00:00'),  (7, '2025-10-04', '10:00:00', '18:00:00'), (8, '2025-10-05', '12:00:00', '20:00:00'),  (9, '2025-10-06', '11:00:00', '19:00:00'), (10, '2025-10-07', '13:00:00', '21:00:00');  -- Інгредієнт  INSERT INTO Інгредієнт (ingredient\_id, ingredient\_name) VALUES  (1, 'Яловичина'), (2, 'Картопля'), (3, 'Помідори'), (4, 'Борошно'), (5, 'Сир Моцарела'),  (6, 'Лосось'), (7, 'Вершки'), (8, 'Шампіньйони'), (9, 'Оливкова олія'), (10, 'Базилік'),  (11, 'Куряче філе'), (12, 'Яйця');  -- Клієнт  INSERT INTO Клієнт (client\_id, full\_name, phone, email) VALUES  (1, 'Іваненко Петро', '0981112233', 'petro.i@example.com'), (2, 'Сидорова Марія', '0674445566', 'maria.s@example.com'),  (3, 'Ковальчук Андрій', '0507778899', 'andrii.k@example.com'), (4, 'Мельник Олена', '0631234567', 'olena.m@example.com'),  (5, 'Григоренко Віталій', '0959876543', 'vitaliy.h@example.com'), (6, 'Савчук Наталія', '0681122334', 'natalka.s@example.com'),  (7, 'Ткаченко Олег', '0934567890', 'oleg.t@example.com'), (8, 'Лисенко Юлія', '0978889900', 'yulia.l@example.com'),  (9, 'Шевченко Дмитро', '0663332211', 'dmytro.sh@example.com'), (10, 'Василенко Ірина', '0675554433', 'iryna.v@example.com');  -- Зал  INSERT INTO Зал (hall\_id, hall\_name, capacity, description) VALUES  (1, 'Основний', 80, 'Великий зал'), (2, 'Тераса', 40, 'Літній майданчик'),  (3, 'VIP', 10, 'Окрема кімната'), (4, 'Камінний', 30, 'Зал з каміном'),  (5, 'Банкетний', 100, 'Для великих свят'), (6, 'Лаунж', 25, 'Зона відпочинку'),  (7, 'Зимовий сад', 15, 'Зал з рослинами'), (8, 'Експрес-бар', 15, 'Для швидких напоїв'),  (9, 'Балкон', 20, 'Вид на місто'), (10, 'Ігрова', 15, 'Зона для дітей');  -- Бенефіт  INSERT INTO Бенефіт (benefit\_id, benefit\_type, description) VALUES  (1, 'Знижка 10%', 'Постійна знижка'), (2, 'Безкоштовний десерт', 'При замовленні від 500 грн'),  (3, 'Пріоритет бронювання', 'Бронювання поза чергою'), (4, 'Подвійні бали', 'Нарахування балів x2'),  (5, 'Безкоштовна кава', 'Одна чашка на день'), (6, 'Подарунок на день народження', 'Спец. подарунок'),  (7, 'Закрита дегустація', 'Запрошення на дегустації'), (8, 'Знижка на вино', 'Знижка 15%'),  (9, 'Кешбек 5%', 'Повернення 5%'), (10, 'Обслуговування без черги', 'Швидке обслуговування');  -- Страва  INSERT INTO Страва (dish\_id, dish\_name, description, price, prep\_time\_min, category) VALUES  (1, 'Борщ Український', 'Класичний борщ', 95.00, 20, 'Перші страви'), (2, 'Стейк Рібай', 'Стейк з яловичини', 450.00, 30, 'Основні страви'),  (3, 'Піца Маргарита', 'Класична піца', 180.00, 15, 'Піца'), (4, 'Цезар з куркою', 'Салат Цезар', 160.00, 15, 'Салати'),  (5, 'Лосось на грилі', 'Філе лосося', 380.00, 25, 'Риба'), (6, 'Чізкейк Нью-Йорк', 'Класичний чізкейк', 120.00, 5, 'Десерти'),  (7, 'Паста Карбонара', 'Паста з беконом', 190.00, 20, 'Паста'), (8, 'Овочі-гриль', 'Мікс овочів', 130.00, 15, 'Гарніри'),  (9, 'Еспресо', 'Міцний кавовий напій', 45.00, 2, 'Напої'), (10, 'Хлібна корзина', 'Свіжовипечений хліб', 50.00, 1, 'Хліб'),  (11, 'Деруни зі сметаною', 'Картопляні оладки', 110.00, 20, 'Українська кухня'), (12, 'Тірамісу', 'Італійський десерт', 135.00, 5, 'Десерти');  -- Співробітник  INSERT INTO Співробітник (employee\_id, position\_id, schedule\_id, full\_name, address, hired\_date, salary) VALUES  (101, 4, 1, 'Жук Богдан', 'вул. Центральна, 1', '2023-01-15', 35000.00), (102, 1, 2, 'Кравчук Анна', 'пр-т Миру, 5', '2023-03-10', 18000.00),  (103, 3, 3, 'Павленко Віктор', 'пл. Соборна, 12', '2022-11-20', 30000.00), (104, 2, 4, 'Гнатюк Олег', 'вул. Козацька, 8', '2024-05-01', 22000.00),  (105, 1, 5, 'Литвин Ірина', 'пров. Затишний, 3', '2024-02-14', 18500.00), (106, 6, 6, 'Мельник Тетяна', 'вул. Героїв, 25', '2023-07-22', 32000.00),  (107, 3, 7, 'Коваленко Сергій', 'вул. Садова, 15', '2024-01-01', 31000.00), (108, 1, 8, 'Степаненко Юрій', 'пр-т Перемоги, 10', '2024-04-05', 17500.00),  (109, 5, 9, 'Ігнатенко Марина', 'вул. Промислова, 14', '2024-06-18', 15000.00), (110, 9, 10, 'Сорока Іван', 'вул. Шевченка, 7', '2022-09-01', 40000.00);  -- Склад\_Страви  INSERT INTO Склад\_Страви (dish\_ingredient\_id, dish\_id, ingredient\_id, quantity\_needed) VALUES  (1, 1, 1, 0.15), (2, 1, 2, 0.2), (3, 2, 1, 0.35), (4, 3, 4, 0.1), (5, 3, 5, 0.15),  (6, 4, 11, 0.1), (7, 5, 6, 0.2), (8, 7, 7, 0.05), (9, 3, 10, 0.01), (10, 8, 3, 0.15),  (11, 12, 12, 0.02), (12, 11, 2, 0.3), (13, 7, 8, 0.05), (14, 2, 9, 0.01);  -- Столик  INSERT INTO Столик (table\_id, hall\_id, table\_number, seats) VALUES  (1, 1, 1, 4), (2, 1, 2, 4), (3, 1, 3, 2), (4, 1, 4, 6), (5, 2, 10, 4),  (6, 2, 11, 2), (7, 3, 20, 10), (8, 4, 30, 4), (9, 5, 40, 8), (10, 6, 50, 4),  (11, 1, 5, 8);  -- Картка\_лояльності  INSERT INTO Картка\_лояльності (card\_id, client\_id, obtaining\_condition, current\_points) VALUES  (1001, 1, 'Перше відвідування', 50), (1002, 2, 'Замовлення від 1000 грн', 200), (1003, 3, 'Постійний клієнт', 550),  (1004, 4, 'Акційна пропозиція', 100), (1005, 5, 'Накопичення балів', 80), (1006, 6, 'Перше відвідування', 20),  (1007, 7, 'Замовлення від 1500 грн', 300), (1008, 8, 'Постійний клієнт', 150), (1009, 9, 'Акційна пропозиція', 90),  (1010, 10, 'Накопичення балів', 400);  -- Закріплення  INSERT INTO Закріплення (assignment\_id, table\_id, employee\_id, assignment\_date, shift\_time) VALUES  (1, 1, 102, '2025-10-10', 'Вечір'), (2, 2, 102, '2025-10-10', 'Вечір'), (3, 3, 105, '2025-10-10', 'День'),  (4, 4, 105, '2025-10-10', 'День'), (5, 5, 108, '2025-10-10', 'Вечір'), (6, 6, 108, '2025-10-10', 'Вечір'),  (7, 7, 102, '2025-10-10', 'Вечір'), (8, 8, 105, '2025-10-11', 'День'), (9, 9, 108, '2025-10-11', 'Вечір'),  (10, 10, 102, '2025-10-11', 'Вечір');  -- Перевага\_з\_картки  INSERT INTO Перевага\_з\_картки (card\_id, benefit\_id) VALUES  (1001, 1), (1002, 3), (1003, 1), (1003, 4), (1004, 5),  (1005, 1), (1006, 10), (1007, 2), (1008, 6), (1009, 1),  (1010, 9);  -- Замовлення  INSERT INTO Замовлення (order\_id, employee\_id, client\_id, table\_id, order\_datetime, total\_amount, status) VALUES  (1, 102, 1, 1, '2025-10-10 18:30:00', 645.00, 'Завершено'),  (2, 105, 3, 4, '2025-10-10 13:15:00', 920.00, 'Завершено'),  (3, 102, 2, 7, '2025-10-10 20:00:00', 1250.00, 'Завершено'),  (4, 108, NULL, NULL, '2025-10-10 21:45:00', 310.00, 'На виніс'),  (5, 105, 5, 2, '2025-10-11 14:00:00', 485.00, 'Оплачено'),  (6, 102, 7, 5, '2025-10-11 19:10:00', 700.00, 'Готується'),  (7, 108, 4, 10, '2025-10-11 20:30:00', 555.00, 'Завершено'),  (8, 105, 10, 3, '2025-10-12 12:45:00', 250.00, 'Оплачено'),  (9, 102, NULL, 6, '2025-10-12 18:00:00', 395.00, 'Готується'),  (10, 108, 8, 9, '2025-10-12 21:00:00', 1010.00, 'Завершено');  -- Деталі\_замовлення  INSERT INTO Деталі\_замовлення (order\_id, dish\_id, table\_id, quantity, unit\_price) VALUES  (1, 1, 1, 2, 95.00), (1, 4, 1, 1, 160.00), (1, 9, 1, 3, 45.00),  (2, 2, 4, 2, 450.00), (2, 8, 4, 1, 130.00),  (3, 5, 7, 3, 380.00), (3, 6, 7, 1, 120.00),  (4, 7, NULL, 1, 190.00), (4, 11, NULL, 1, 110.00),  (5, 3, 2, 1, 180.00), (5, 12, 2, 2, 135.00),  (6, 4, 5, 2, 160.00), (6, 9, 5, 4, 45.00), (6, 10, 5, 1, 50.00),  (7, 1, 10, 1, 95.00), (7, 7, 10, 2, 190.00),  (8, 9, 3, 5, 45.00), (8, 6, 3, 1, 120.00),  (9, 5, 6, 1, 380.00),  (10, 2, 9, 1, 450.00), (10, 8, 9, 2, 130.00), (10, 12, 9, 1, 135.00); |

***6)* **Приклади SQL-коду видалення записів з таблиць з словесним описом дії (мінімум 5).****

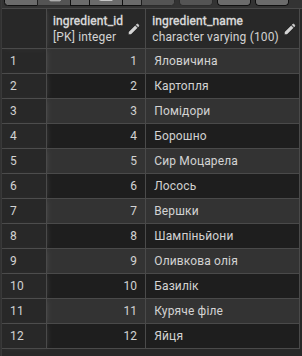
|  |  |
| --- | --- |
| -- ##################################################################  -- 3. DML: ПРИКЛАДИ ВИДАЛЕННЯ ЗАПИСІВ (DELETE)  -- Виправлено порушення зовнішніх ключів (FK)  -- ##################################################################  -- 1. Дія: Видалити всі записи про клієнтів, у яких не вказана електронна пошта.  DELETE FROM Клієнт  WHERE email IS NULL;  -- 2. Дія: Видалити деталі всіх замовлень, які були зроблені до 1 жовтня 2025 року.  DELETE FROM Деталі\_замовлення  WHERE order\_id IN (  SELECT order\_id  FROM Замовлення  WHERE order\_datetime < '2025-10-01 00:00:00'  );  -- 3. Дія: Видалити всі страви, ціна яких менше 50.00 грн.  -- ВИПРАВЛЕННЯ: Спочатку видаляємо посилання в Деталі\_замовлення.  DELETE FROM Деталі\_замовлення  WHERE dish\_id IN (  SELECT dish\_id  FROM Страва  WHERE price < 50.00  );  DELETE FROM Страва  WHERE price < 50.00;  -- 4. Дія: Видалити всі столики, які знаходяться в залі з назвою 'Тераса'.  -- ВИПРАВЛЕННЯ: Спочатку видаляємо посилання в Закріплення.  DELETE FROM Закріплення  WHERE table\_id IN (  SELECT table\_id  FROM Столик  WHERE hall\_id = (  SELECT hall\_id  FROM Зал  WHERE hall\_name = 'Тераса'  )  );  DELETE FROM Столик  WHERE hall\_id = (  SELECT hall\_id  FROM Зал  WHERE hall\_name = 'Тераса'  );  -- 5. Дія: Видалити картки лояльності, які були отримані за "Перше відвідування" і мають менше 50 балів.  DELETE FROM Картка\_лояльності  WHERE current\_points < 50 AND obtaining\_condition = 'Перше відвідування'; |  |

*****7)* Приклади SQL-коду зміни даних в таблицях з словесним описом дії (мінімум 5).****

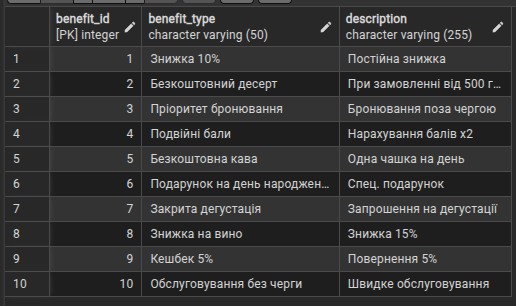
|  |  |
| --- | --- |
| --- ##################################################################  -- 4. DML: ПРИКЛАДИ ЗМІНИ ДАНИХ (UPDATE)  -- ##################################################################  -- 1. Дія: Підвищити зарплату на 15% усім співробітникам, які працюють на посаді 'Офіціант'.  UPDATE Співробітник  SET salary = salary \* 1.15  WHERE position\_id = (  SELECT position\_id  FROM Посада  WHERE position\_name = 'Офіціант'  );  -- 2. Дія: Збільшити ціну на 'Стейк Рібай' до 500.00 грн.  UPDATE Страва  SET price = 500.00  WHERE dish\_name = 'Стейк Рібай';  -- 3. Дія: Змінити статус усіх замовлень, які зараз знаходяться в стані 'Готується', на 'Оплачено'.  UPDATE Замовлення  SET status = 'Оплачено'  WHERE status = 'Готується';  -- 4. Дія: Нарахувати 50 бонусних балів на картку лояльності клієнта Ковальчук Андрій.  UPDATE Картка\_лояльності  SET current\_points = current\_points + 50  WHERE client\_id = (  SELECT client\_id  FROM Клієнт  WHERE full\_name = 'Ковальчук Андрій'  );  -- 5. Дія: Змінити графік роботи для співробітника Жук Богдан (employee\_id=101) на вечірню зміну.  UPDATE Співробітник  SET schedule\_id = 2 -- schedule\_id 2 - вечірня зміна 01.10.2025  WHERE employee\_id = 101; |  |

****8) Скріншоти створених та заповнених таблиць.****

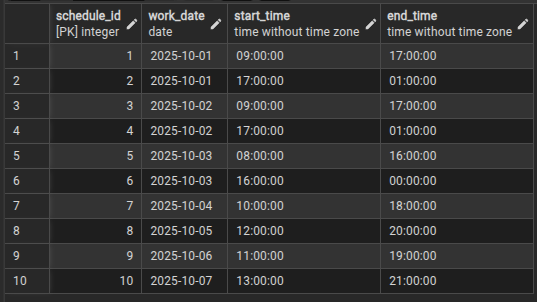
* Інгредієнт:



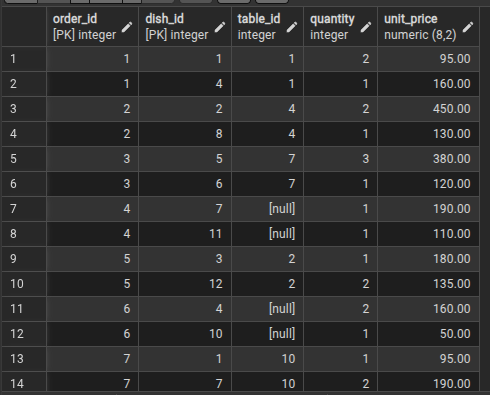
* Бенефіт:



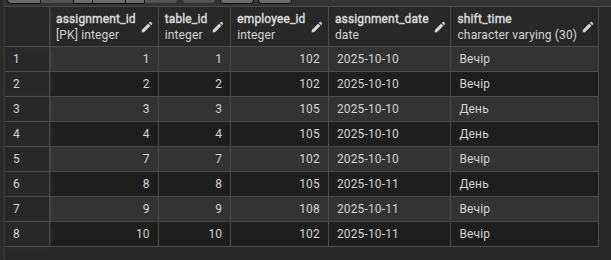
* Графік\_роботи:

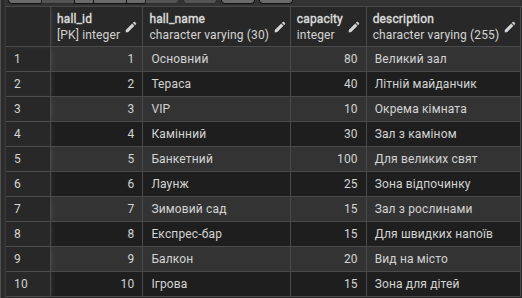


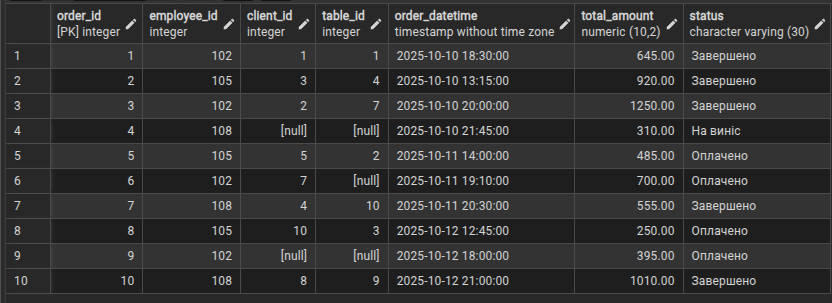
* Деталі\_замовлення:



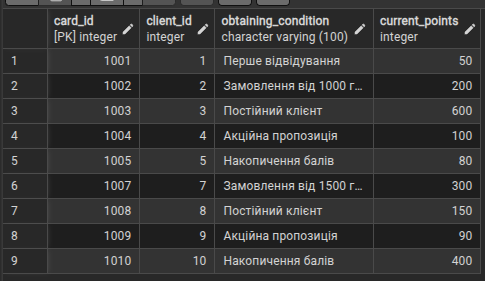
* Закріплення:



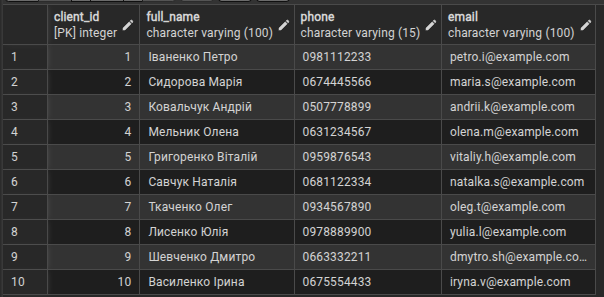
* Зал:
* Замовлення:



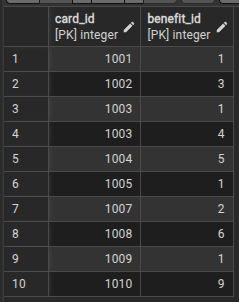
* Картка\_лояльності:



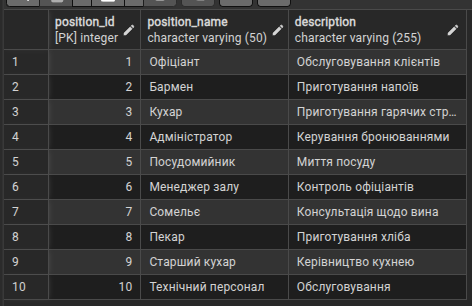
* Клієнт:



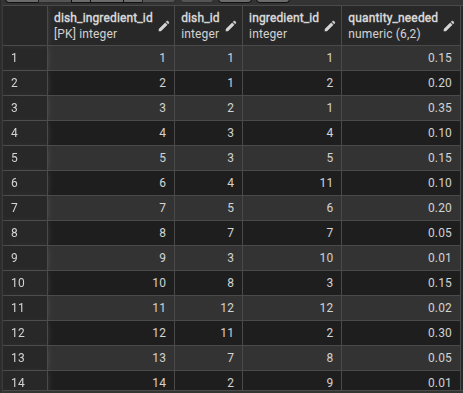
* Перевага\_з\_картки:



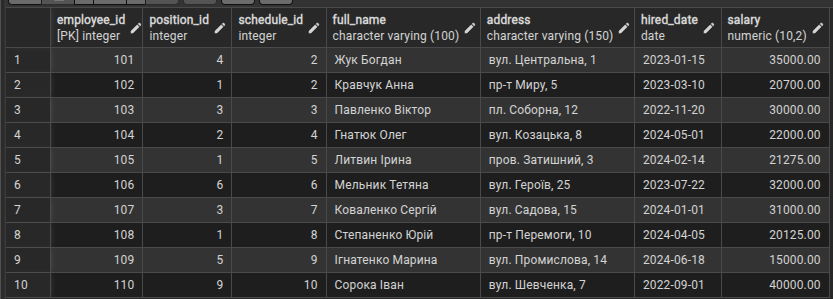
* Посада:



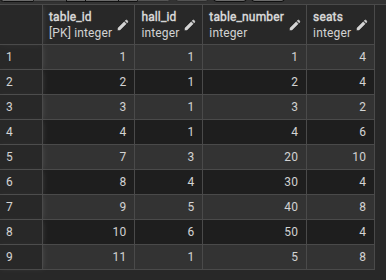
* Склад\_страви:



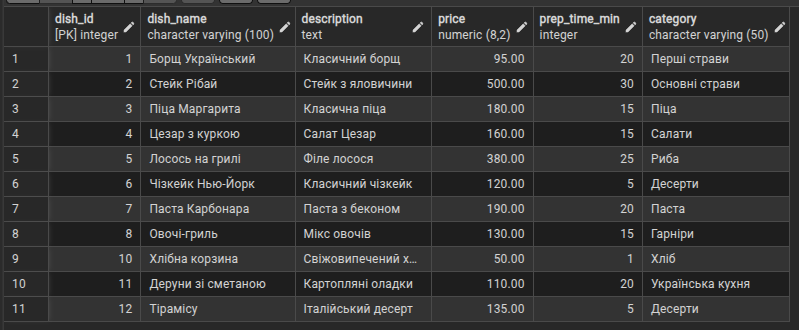
* Співробітник:



* Столик:



* Страва:



## ****Висновок:****

Протягом виконання лабораторної роботи було успішно здійснено комплексне проектування та реалізацію бази даних, що відповідає вимогам заданої предметної області (ресторанний бізнес).

### Основні досягнення та результати

1. **Створення логічної моделі (ER-діаграма)**: На початковому етапі було розроблено логічну модель, що чітко відображає сутності (таблиці) та зв'язки між ними, включаючи сутності **"Клієнт"**, **"Страва"**, **"Замовлення"**, **"Співробітник"** та інші.
2. **Реалізація структури (DDL)**: Був створений повний скрипт мовою SQL (Data Definition Language) для побудови всіх необхідних таблиць та визначення їхніх атрибутів, **первинних (PK)** та **зовнішніх (FK) ключів**. Важливим кроком було виправлення синтаксичних помилок, зокрема заміна некоректного типу даних DATETIME на **TIMESTAMP**, що забезпечило сумісність із середовищем PostgreSQL.
3. **Забезпечення цілісності даних**: Для уникнення порушень обмежень зовнішнього ключа (помилки 23503), було застосовано каскадні операції **ON DELETE CASCADE** у визначенні ключових залежностей (наприклад, між Страва та Деталі\_замовлення, а також між Столик та Закріплення). Це дозволило автоматично видаляти залежні записи при видаленні батьківської сутності, підвищуючи гнучкість системи при адмініструванні даних.
4. **Маніпуляція даними (DML)**: Робота включала виконання всіх основних операцій мови Data Manipulation Language:
   * **INSERT**: Наповнення всіх таблиць реальними тестовими даними.
   * **DELETE**: Розробка та успішне виконання команд видалення записів, включаючи складні видалення з використанням підзапитів та врахуванням міжтабличних залежностей.
   * **UPDATE**: Створення прикладів модифікації існуючих даних (наприклад, зміна зарплати, оновлення статусу замовлень), що продемонструвало можливість підтримки актуальності інформації в базі.

Виконана лабораторна робота підтвердила практичні навички роботи з реляційними базами даних, розуміння концепцій цілісності даних та вміння писати коректні та ефективні SQL-скрипти. Досвід виправлення системних помилок (пов'язаних із типом даних та обмеженнями FK) є критично важливим для подальшої роботи у сфері розробки та адміністрування баз даних.