

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 та засоби управління»**

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

Виконав: студент ІІІ курсу

ФПМ групи КВ-93

Зеленяк І.О.

Перевірив:

## Розробка моделі «сутність-зв’язок» роботи парковки

### Опис обраної предметної галузі

В даній лабораторній буде розглядатися принцип роботи парковки з розділеними персональними місцями для кожної автівки.

Вважаємо що в одного користувача може бути доступ до декількох автівок а в одного авто може бути декілька користувачів.

В кожного авто і користувача є власна унікальна перепустка за якою контролюється допуск на парковку.

Одному паркомісцю відповідає тільки одна машина яка може там стояти, якщо воно орендоване, інакше жодна машина не може там стояти.

В результаті розробки бази даних буде можливо реалізувати автоматизований контроль шлагбаумом парковки з контролем особи що виїжджає та автівки на якій виїжджає особа.

### Принцип роботи парковки

Для того щоб отримати місце на парковці потрібно зареєструвати данні власника, авто та людей які мають право на паркування цього авто. Після чого видаються перепустки на машину і на кожного користувача окремо.

При вході користувача на парковку сканується перепустка і розблоковуються всі місця до яких користувач має доступ, а при виїзді авто з парковки вони блокуються.

При в'їзді авто на парковку розблоковується тільки те місце на якому має стояти машина.

Всі події що відбуваються на парковці реєструються.

### Перелік сутностей з описом їх призначення

**Автівка** фізичний автомобіль який ідентифікується за державним номером та перепусткою та має атрибути:

* ***Державний номер*** - номер автівки виданий державним органом
* ***Власник авто*** - ідентифікаційний номер користувача, який є фактичним власником
* ***Марка*** - марка автомобіля

**Власник авто** особа яка є власником авто:

* ***Прізвище , Ім’я , По-батькові***
* ***Номер телефону*** - список номерів телефону

**Користувач -** особа яка має право на пересування автомобілем та має атрибути:

* ***Прізвище , Ім’я , По-батькові***
* ***Номер телефону*** - список номерів телефону

**Перепустка** - електронний ключ власника або авто, який надає доступ до паркомісця та має атрибути:

* ***Ідентифікатор перепустки*** - унікальний номер перепустки який не повторюється UID довжиною 4 байти.Цілочислене
* ***Тип перепустки*** - відповідає моделі перепустки тобто така що може бути видана особі або стаціонарно встановлена на авто.
* ***Статус перепустки*** - чи є перепустка активною

**Паркомісце -** фізичне місце для парковки автомобіля в межах паркувального майданчика тамає атрибути:

* ***Номер паркомісця*** - унікальний номер паркомісця який не повторюється
* ***Статус*** - вказує на те чи зайняте місце (орендоване на деякий період)
* ***Тип паркомісця*** - тип який вказує на те чи є паркомісце: гаражем, з дахом , без даху

### Перелік звязків між сутностями з описом їх призначення

Сутність **Перепустка** пов’язана із сутностями **Власник Авто, Автівка, Користувач** зв’язком **Ідентифікує** типу M:N наступним чином:

* у кожен момент часу кожна з сутностей: **Власник Авто, Автівка, Користувач** може бути **Ідентифікована** лише однією **Перепусткою**.
* У той самий час історичні зміни у роботі парковки можуть призвести до того, що однією і тою самою перепусткою користуються різні **Власники Авто, Автівки, Користувачі**.
* Таким чином одна **Перепустка** може бути надана багатьом сутностям **Власники Авто, Автівки, Користувачі**, а **Власники Авто, Автівки, Користувачі** у різний період часу можуть бути **ідентифіковані** різними **Перепустками**.

Сутність **Власник Авто** пов’язана із сутністю **Автівка** зв’язком **Володіє** типу 1:N наступним чином:

* Один **Власник Авто** може **Володіти** багатьма **Автівками**
* Одна **Автівка** може бути у **Власності** лише одного **Власника Авто**

Сутність **Користувач** пов’язана із сутністю **Автівка** зв’язком **Керує** типу N:M наступним чином:

* Один **Користувач** може **Керувати** багатьма **Автівками**
* Одна **Автівка** може бути у **Керована** багатьма **Користувачами**

Сутність **Автівка** пов’язана із сутністю **Паркомісце** зв’язком **Використовує** типу M:N наступним чином:

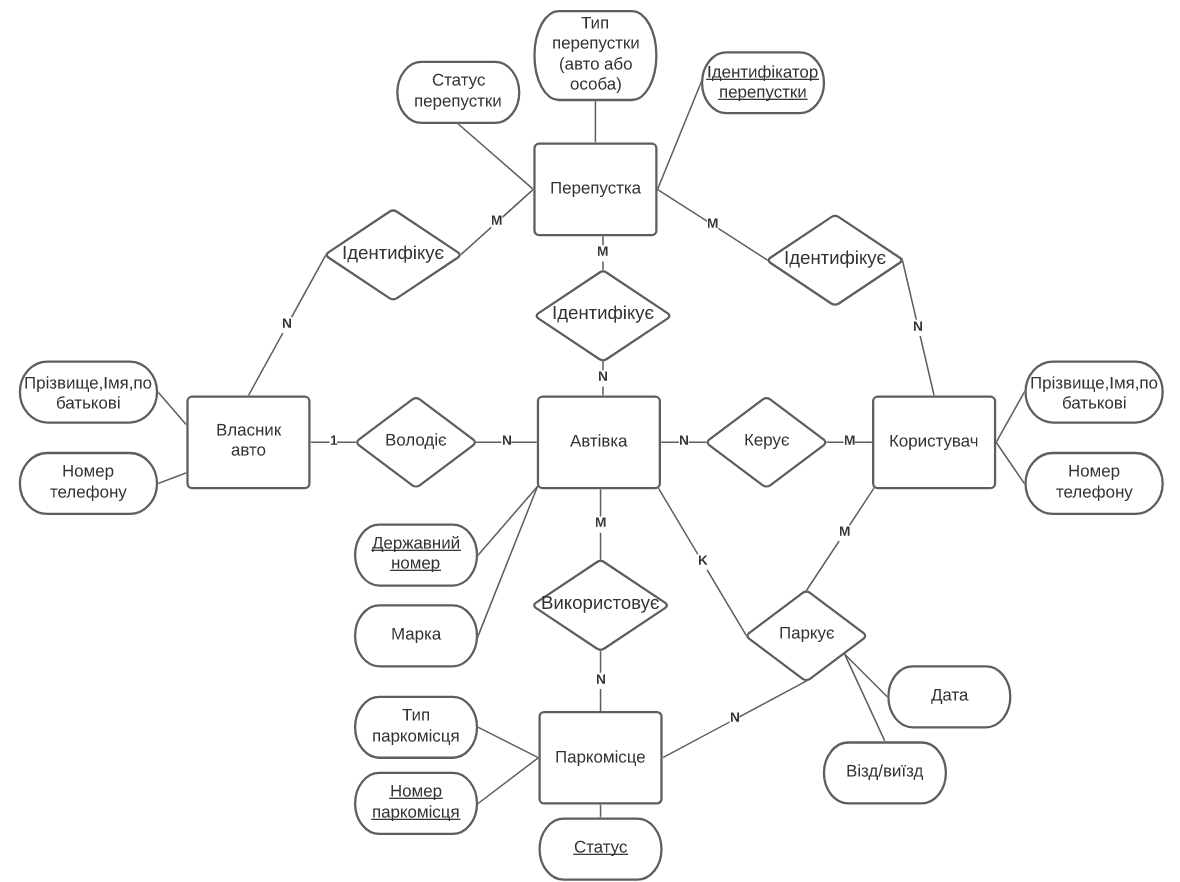
* У кожен момент часу одна **Автівка** може **Використовувати** лише одне **Паркомісце**
* У той самий час історичні зміни у роботі парковки можуть призвести до того, що одним і тим самим **Паркомісцем** **Користуються** різні **Автівки.**
* Таким чином одна **Автівка** може **Використовувати** різні **Паркомісця**, а одне **Паркомісце** може використовуватись багатьма **Автівками** у різний період часу.

Сутності **Автівка, Користувач, Паркомісце** пов’язані зв’язком **Паркує** типу K:M:N наступним чином:

* **Користувач** у довільний час може виїхати/заїхати на парковку на **Автівці** та використати **Паркомісце**
* Фактично цей зв’язок реалізує зберігання історії використання парковки

### Графічний файл розробленої моделі «сутність-зв’язок»

Рисунок



[Посилання на діаграму](https://lucid.app/lucidchart/1369ec07-2d40-43f5-9ef6-ad393ed7d69a/edit?viewport_loc=53%2C-214%2C2067%2C1650%2C0_0&invitationId=inv_4983367c-0fe9-400b-8fe2-440fcdc16171)

Обрана система нотацій Пітера Чена

### Перетворення діаграми в таблиці бази данних

Сутності **Власник авто** та **Користувач** фактично є людьми з однаковим набором атрибутів тому їх було перетворено в таблицю ***Persons***, а сутність **Автівка** було перетворено в таблицю ***Cars****.*

Сутність **Перепустка** було перетворено у таблицю ***RFIDs***

Сутність **Паркомісце** було перетворено у таблицю ***ParkingPlace***

Зв’язок (1:N) **Володіє** між **Власником авто** та **Автівка** реалізований за допомогою атрибута ***CarOwnerID*** у таблиці ***Cars***, що є зовнішнім ключем та посилається на поле ***PersonID*** з таблиці ***Persons***

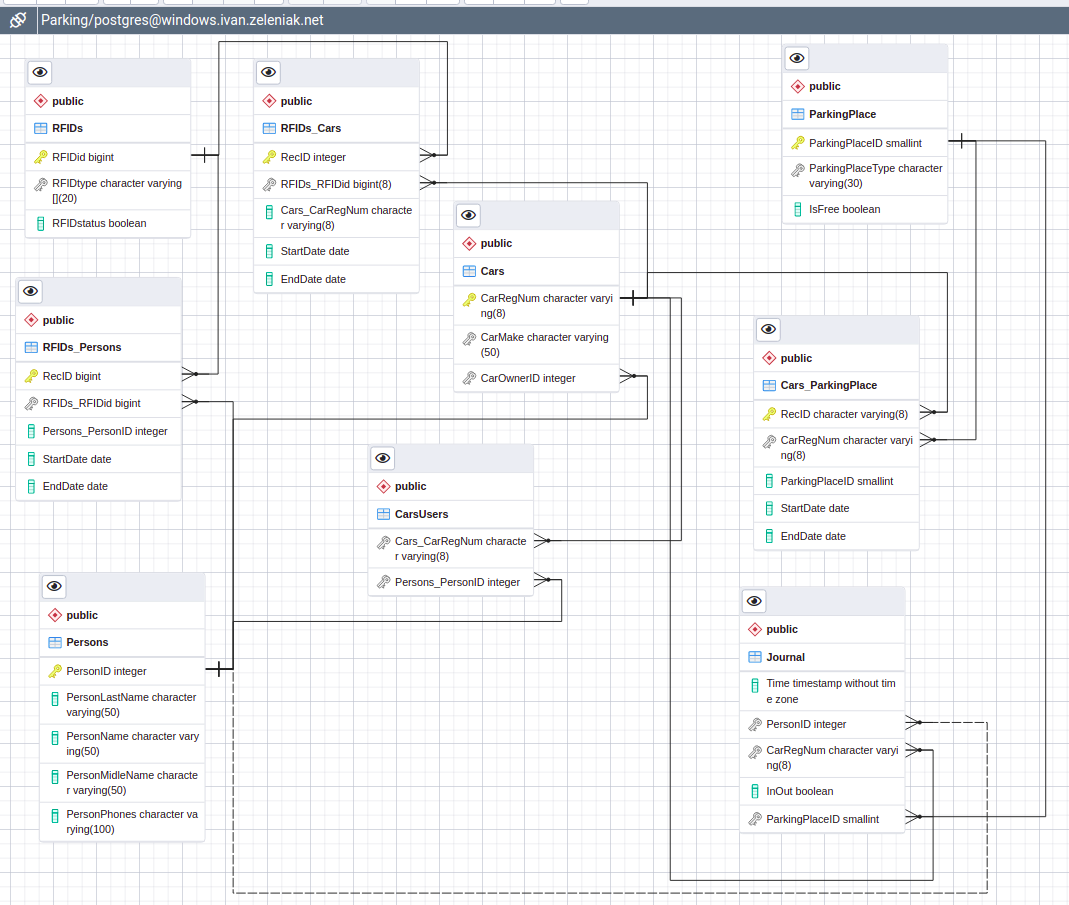
Зв’язок (M:N) **Керує** між **Користувач** та **Автівка** реалізований за допомогою додаткової таблиці ***CarsUsers***, що містить два зовнішні ключі на таблицю ***Cars*** поле ***CarRegNum*** та таблицю ***Persons*** поле ***PersonID***

Зв’язки (M:N) **Ідентифікує** між **Користувач** та **Перепустка** та **Власник Авто** та **Перепустка** реалізований за допомогою додаткової таблиці ***RFIDs\_Persons***, що містить два зовнішні ключі на таблицю ***RFIDs*** поле ***RFIDid*** та таблицю ***Persons*** поле ***PersonID***

Зв’язок (M:N) **Ідентифікує** між **Автівка** та **Перепустка** реалізований за допомогою додаткової таблиці ***RFIDs\_Cars***, що містить два зовнішні ключі на таблицю ***RFIDs*** поле ***RFIDid*** та таблицю ***Cars*** поле ***CarRegNum***

Зв’язок (M:N) **Використовує** між **Автівка** та **Паркомісце** реалізований за допомогою додаткової таблиці ***Cars\_ParkingPlace***, що містить два зовнішні ключі на таблицю ***ParkingPlace*** поле ***ParkingPlaceID*** та таблицю ***Cars*** поле ***CarRegNum***

Зв’язок **Паркує** між **Автівка**, **Паркомісце**, **Користувач** реалізований за допомогою додаткової таблиці ***Journal*** що містить зовнішні ключі на таблицю ***Cars*** поле ***CarRegNum, ParkingPlace*** поле ***ParkingPlaceID*** *та*  ***Persons*** поле ***PersonID****.*



Рисунок

### Нормалізація таблиць баз данних

При перетворенні сутності **Користувач** та **Власник авто** буде створена таблиця ***Persons***, та вона знаходитиметься в першій нормальній формі через те що існує поле з телефонами ***PersonPhones*** куди може бути записано декілька телефонів. Для того щоб нормалізувати таблицю потрібно створити ще одну таблицю ***Phones*** яка міститиме всі номера телефонів та їх відповідність до ідентифікаторів користувача. Обидві таблиці будуть знаходитися в 3 нормальній формі.

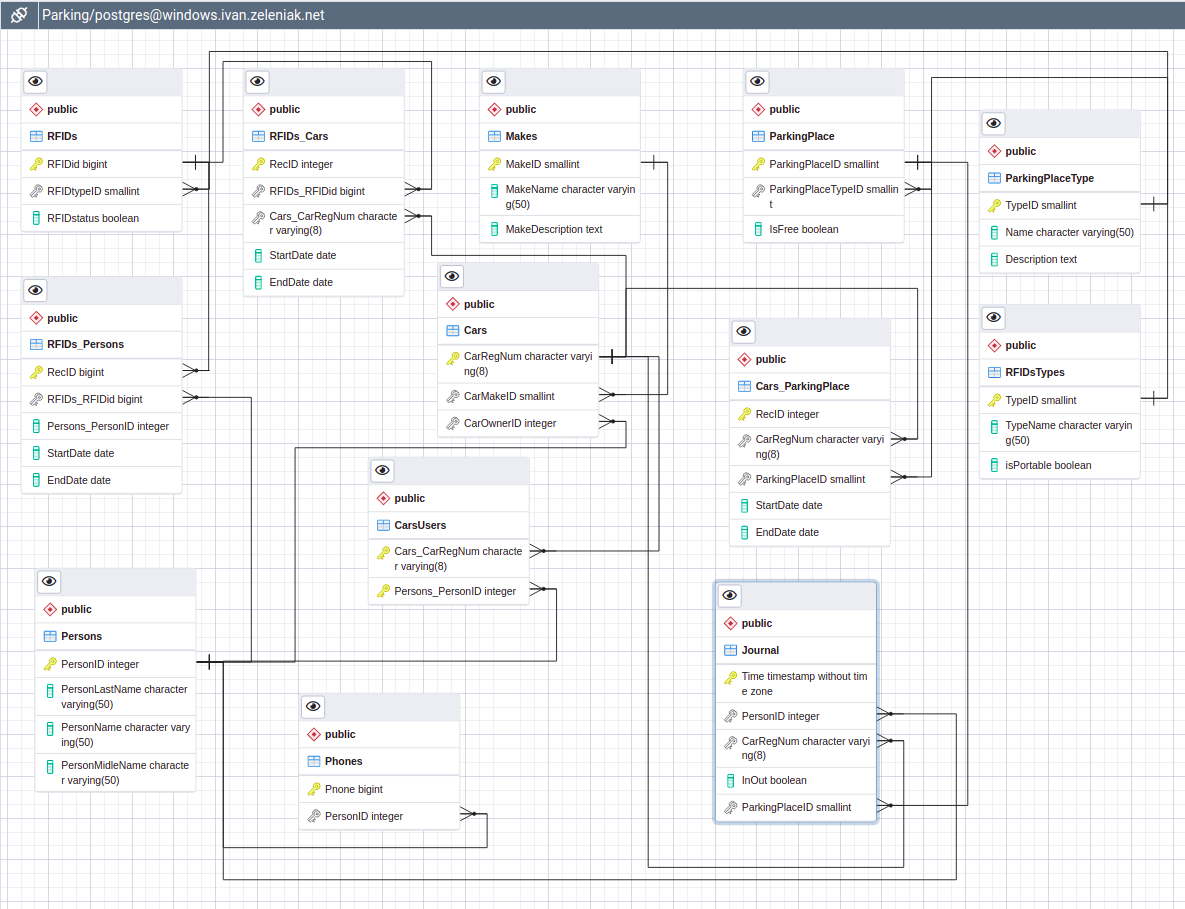
При створенні таблиці машин ***Cars*** які будуть паркуватися, вона буде знаходитися в другій нормальній формі, так, як поле “марка автомобіля” ***CarMake*** може бути записана по різному. Для усунення цієї проблеми потрібно додати таблицю ***Makes*** в якій будуть існувати всі можливі варіанти марок автомобілів. І таблиця машин перейде в третю нормальну форму.

За таким самим принципом нормалізуємо інші таблиці:

При створенні таблиці перепусток ***RFIDs*** вона буде знаходитися в другій нормальній формі через неоднозначність типу перепустки. Для переведення її в третю нормальну форму потрібно додати таблицю ***RFIDsTypes*** з типами перепусток.

При створенні таблиці паркомісць вона буде знаходитися в другій нормальній формі для переведення її в третю потрібно додати таблицю ***ParkingPlaceType*** з типами місць парковок.

Та в результаті отримуємо таблиці БД такого вигляду

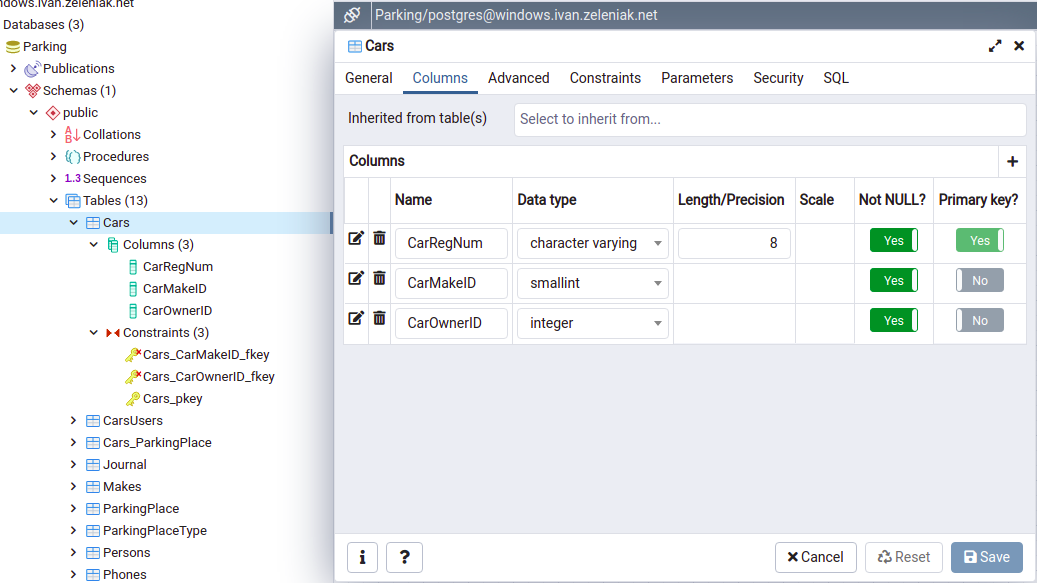


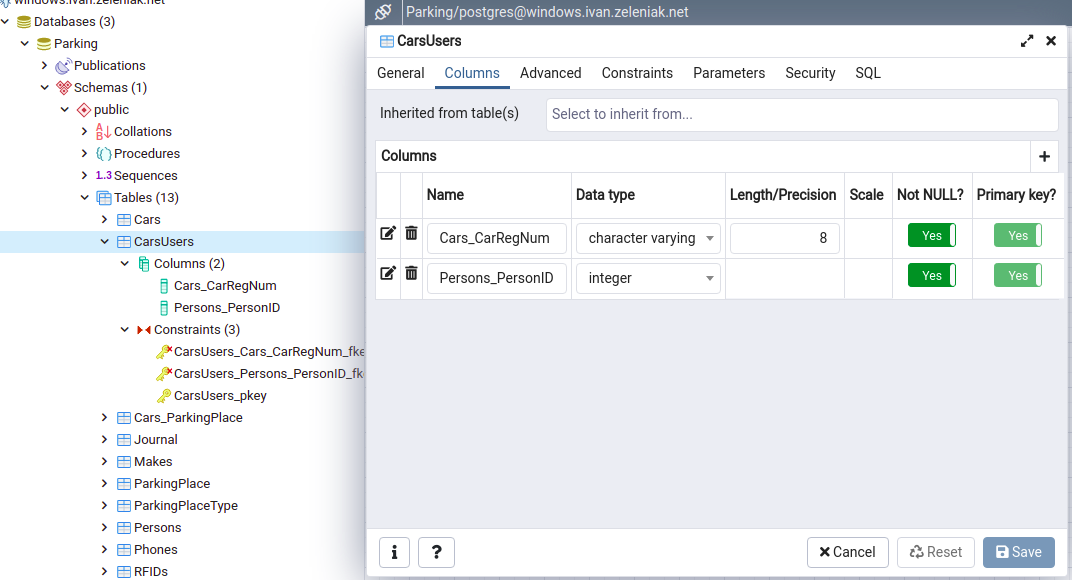
Рисунок

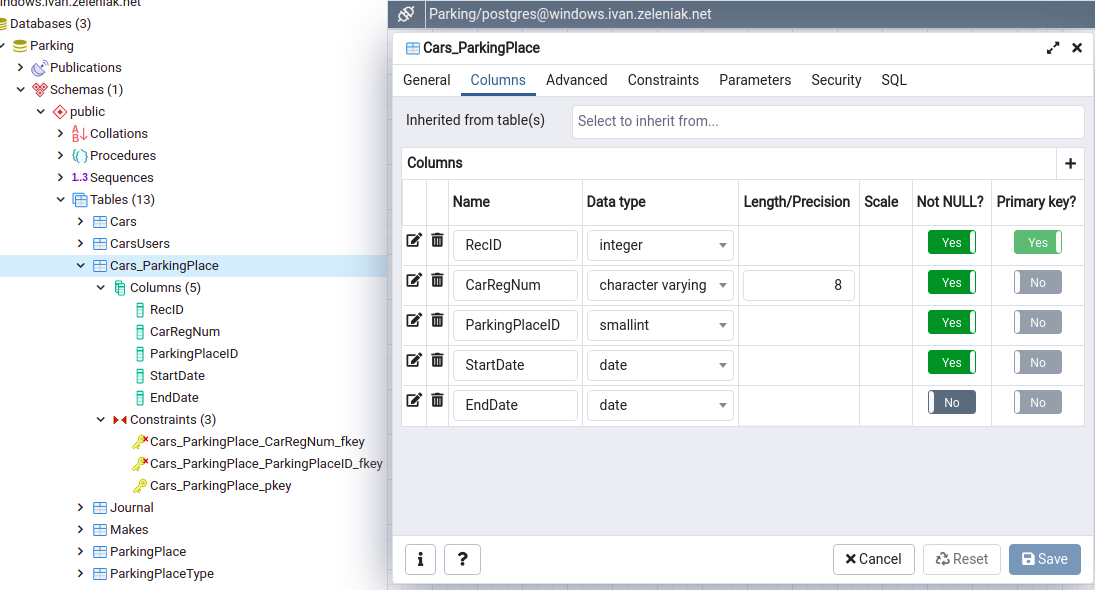
### Опис стовбців їх данних типів

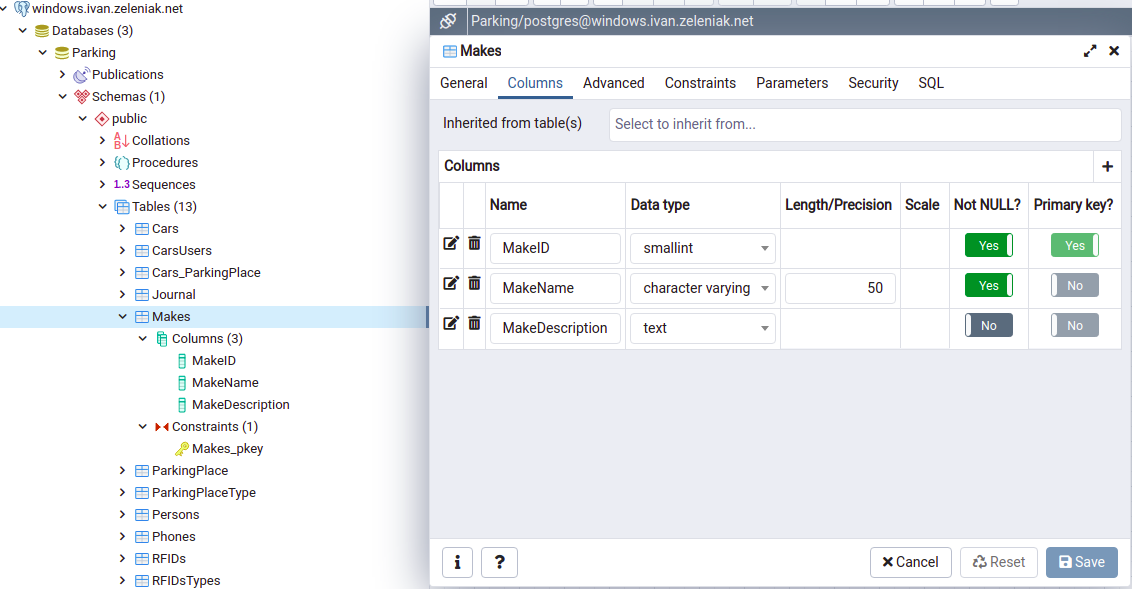
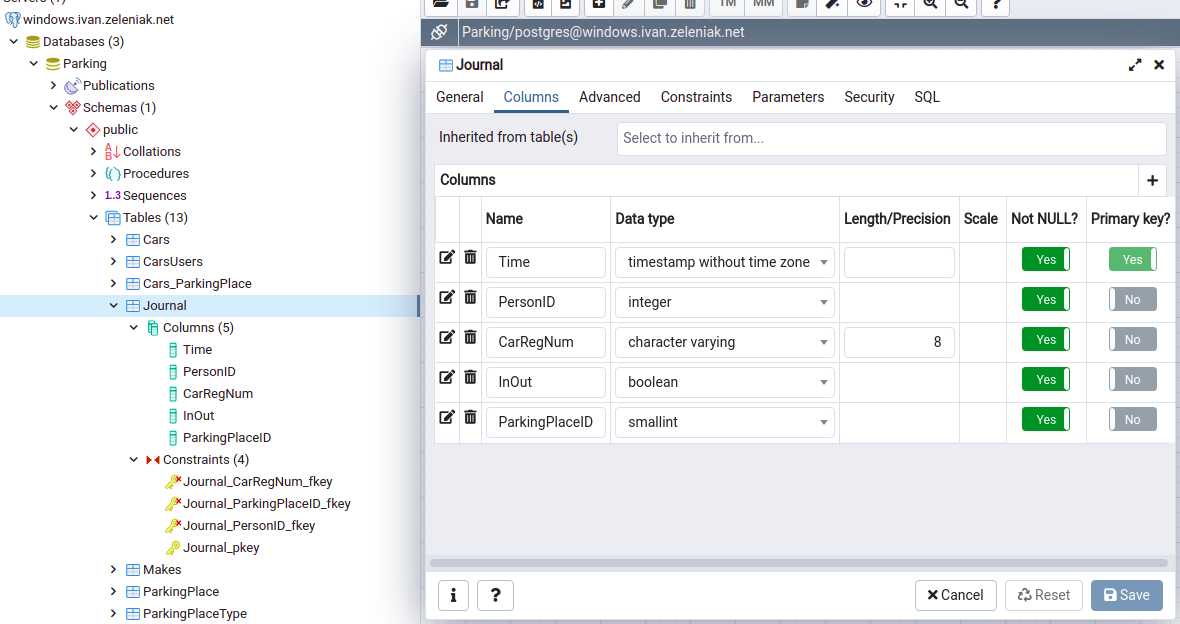
|  |  |  |  |
| --- | --- | --- | --- |
| **Назва таблиці/поля** | **тип**  **даних** | **обов’язково** | **опис** |
| **Таблиця "Cars"** |  |  |  |
| *"CarRegNum"* | text | Так | Номер автівки |
| *"CarMakeID"* | smallint | Так | Ідентифікатор виробника автівки |
| *"CarOwnerID"* | integer | Так | ідентифікатор власника автівки |
|  |  |  |  |
| **Таблиця "CarsUsers"** |  |  |  |
| *"Cars\_CarRegNum"* | text | Так | Номер автівки |
| *"Persons\_PersonID"* | integer | Так | Ідентифікатор користувача |
|  |  |  |  |
| Таблиця "Cars\_ParkingPlace" |  |  |  |
| *"RecID"* | integer | Так |  |
| *"CarRegNum"* | text | Так | Номер автівки |
| *"ParkingPlaceID"* | smallint | Так | Номер місця парковки |
| *"StartDate"* | date | Так | Дата видачі |
| *"EndDate"* | date |  | Дата видачі |
|  |  |  |  |
| Таблиця "Journal" |  |  |  |
| *"Time"* | timestamp | Так | час реєстрації |
| *"PersonID"* | integer | Так | ідентифікатор людини |
| *"CarRegNum"* | text | Так | номер автівки |
| *"InOut"* | boolean | Так | булева змінна для визначення в’їзд/виїзд |
| *"ParkingPlaceID"* | smallint | Так | ідентифікатор парковки |
|  |  |  |  |
| Таблиця "Makes" |  |  |  |
| *"MakeID"* | smallint | Так | ідентифікатор марки авто |
| *"MakeName"* | text | Так | назва марки |
| *"MakeDescription"* | text | Ні | опис |
|  |  |  |  |
| Таблиця "ParkingPlace" |  |  |  |
| *"ParkingPlaceID"* | smallint | Так | ідентифікатор парковки |
| *"ParkingPlaceTypeID"* | smallint | Так | тип паркувального місця |
| *"IsFree"* | boolean | Так | булева змінна вільності парковки |
|  |  |  |  |
| Таблиця "ParkingPlaceType" |  |  |  |
| *"TypeID"* | smallint | Так | ідентифікатор типу парковки |
| *"Name"* | text | Так | назва типу |
| *"Description"* | text | Ні | опис |
|  |  |  |  |
| Таблиця "Persons" |  |  |  |
| *"PersonID"* | integer | Так | ідентифікатор персони |
| *"PersonLastName"* | text | Так | фамілія персони |
| *"PersonName"* | text | Так | ім’я персони |
| *"PersonMidleName"* | text | Ні | по батькові персони |
|  |  |  |  |
| Таблиця "Phones" |  |  |  |
| *"Pnone"* | bigint | Так | номер телефону |
| *"PersonID"* | integer | Так | ідентифікатор персони |
|  |  |  |  |
| Таблиця "RFIDs" |  |  |  |
| *"RFIDid"* | bigint | Так | ідентифікатор перепустки |
| *"RFIDtypeID"* | smallint | Так | тип перепустки |
| *"RFIDstatus"* | boolean | Так | статус перепустки |
|  |  |  |  |
| Таблиця "RFIDsTypes" |  |  |  |
| *"TypeID"* | smallint | Так | ідентифікатор типу перепустки |
| *"TypeName"* | text | Так | назва типу перепустки |
| *"isPortable"* | boolean | Так | вказує тип перепустки для людини чи автівки |
|  |  |  |  |
| Таблиця "RFIDs\_Cars" |  |  |  |
| *"RecID"* | integer | Так | номер запису |
| *"RFIDs\_RFIDid"* | bigint | Так | ідентифікатор перепустки |
| *"Cars\_CarRegNum"* | text | Так | номер автівки |
| *"StartDate"* | date | Так | дата початку |
| *"EndDate"* | date | Ні | дата завершення |
|  |  |  |  |
| Таблиця "RFIDs\_Persons" |  |  |  |
| *"RecID"* | bigint | Так | номер запису |
| *"RFIDs\_RFIDid"* | bigint | Так | ідентифікатор перепустки |
| *"Persons\_PersonID"* | integer | Так | ідентифікатор персони |
| *"StartDate"* | date | Так | дата початку |
| *"EndDate"* | date | Ні | дата завершення |

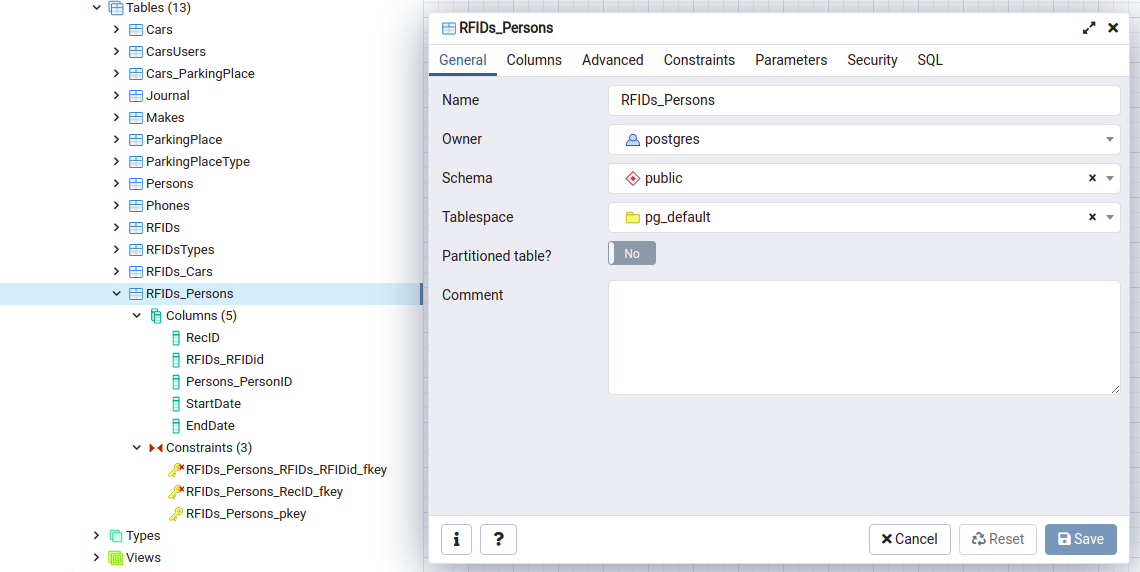
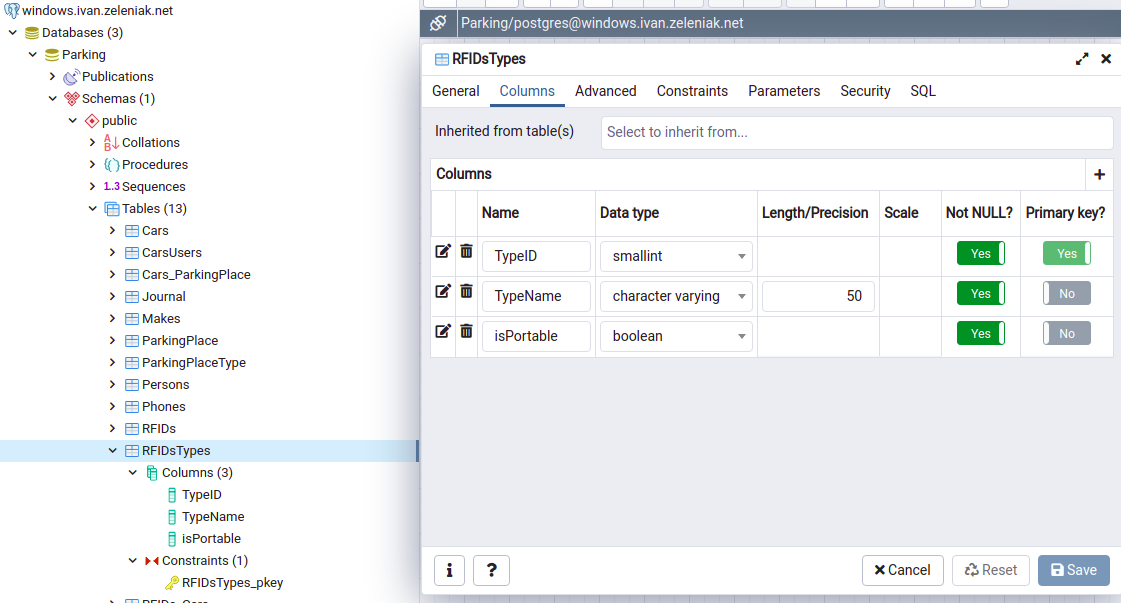
### Копії екрану з pgAdmin4, що відображають назви, типи та обмеження на стовпці

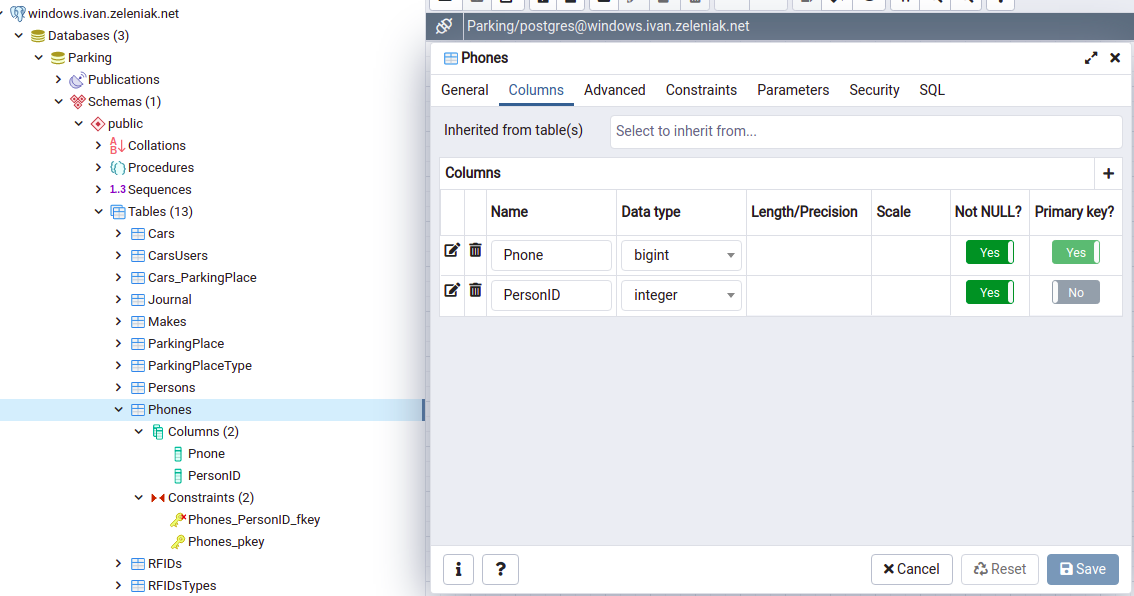
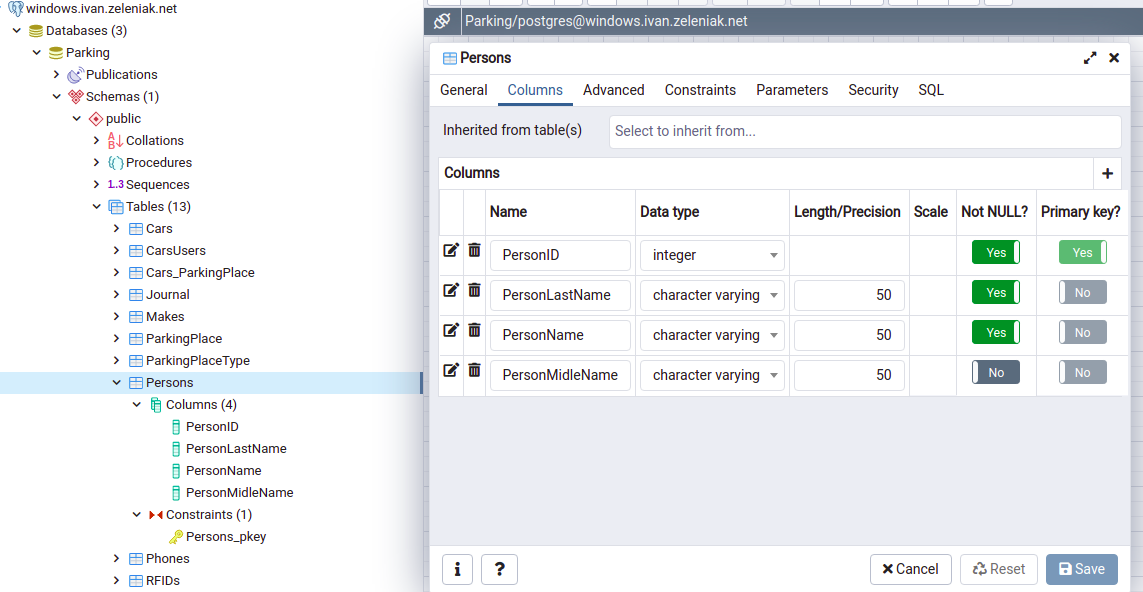
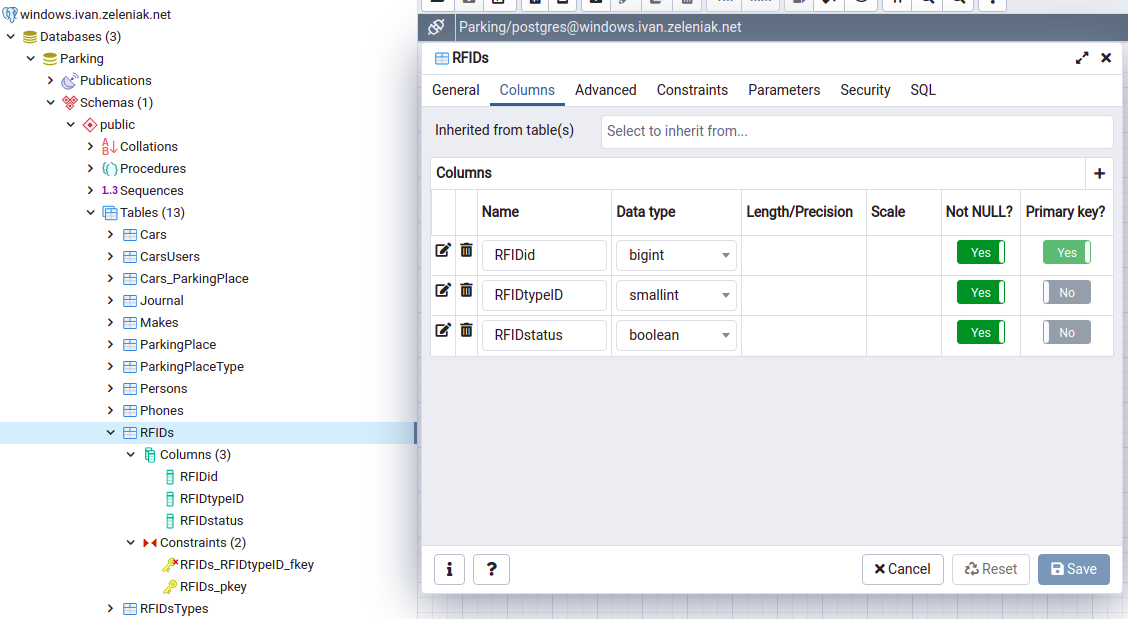


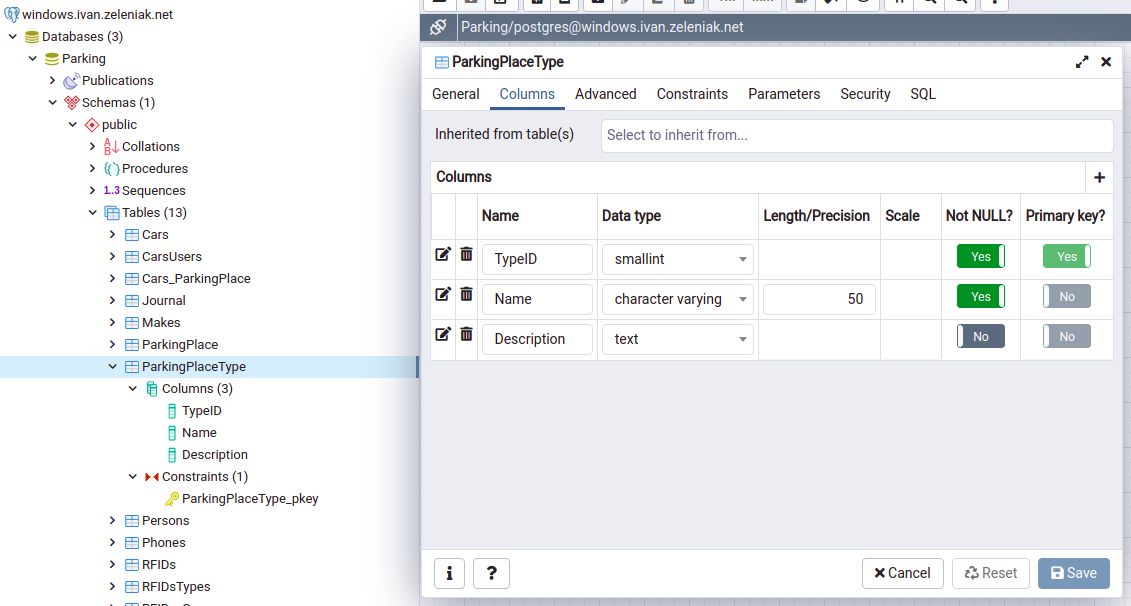


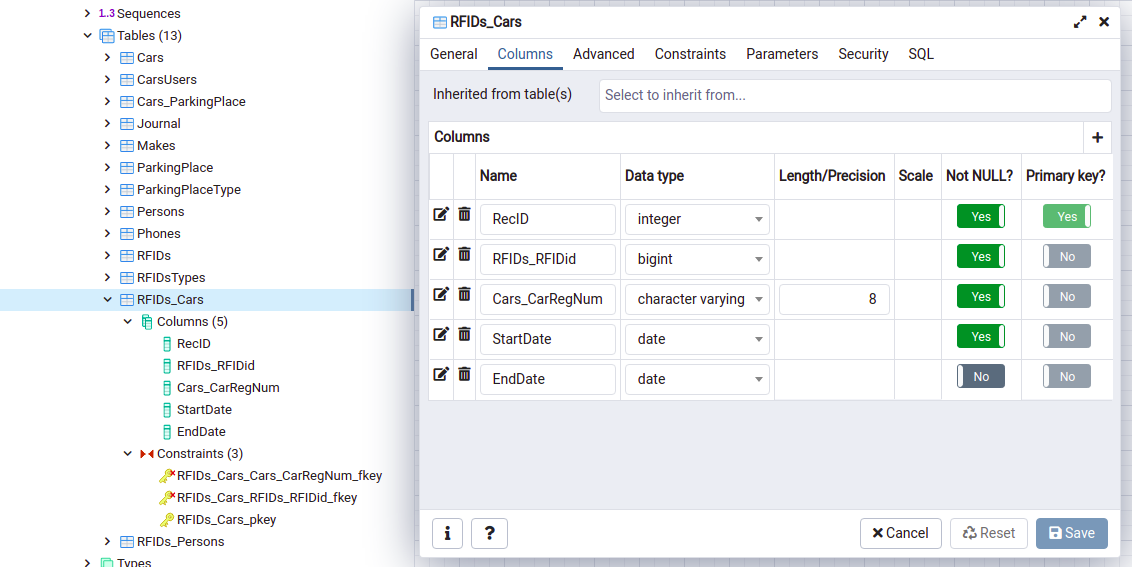


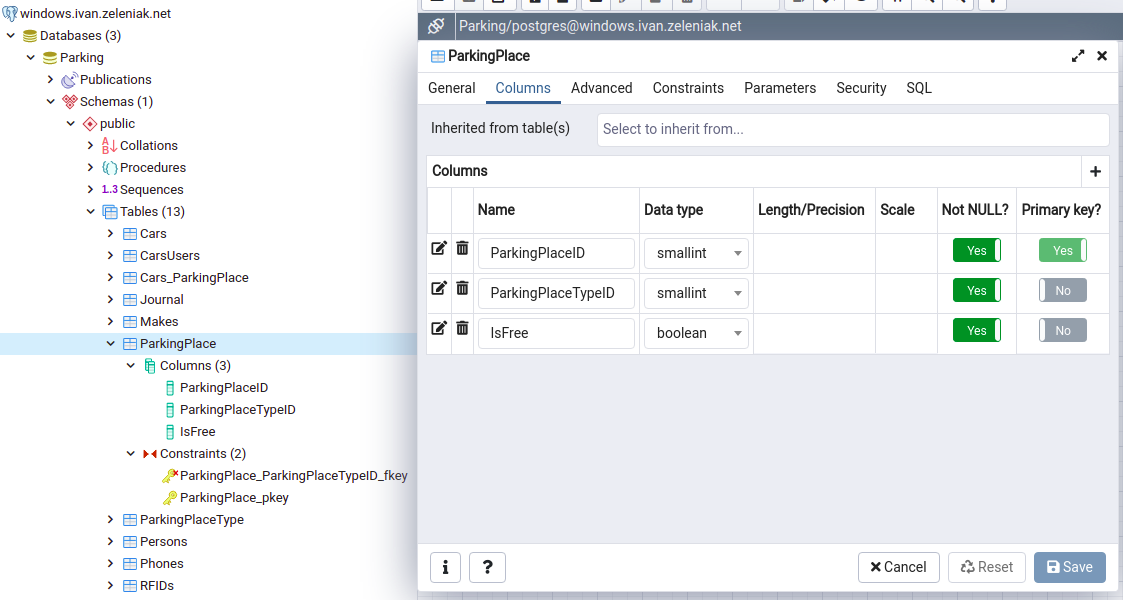


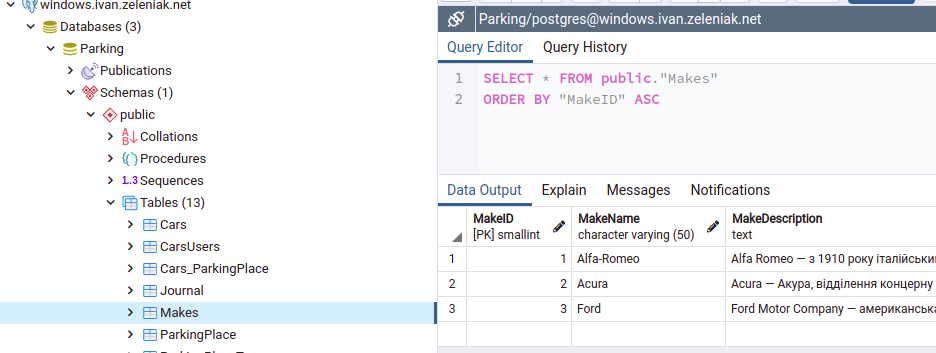


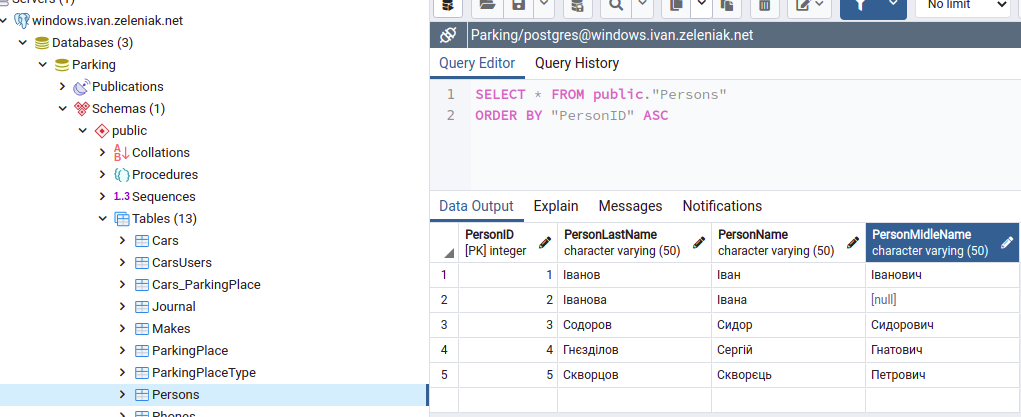


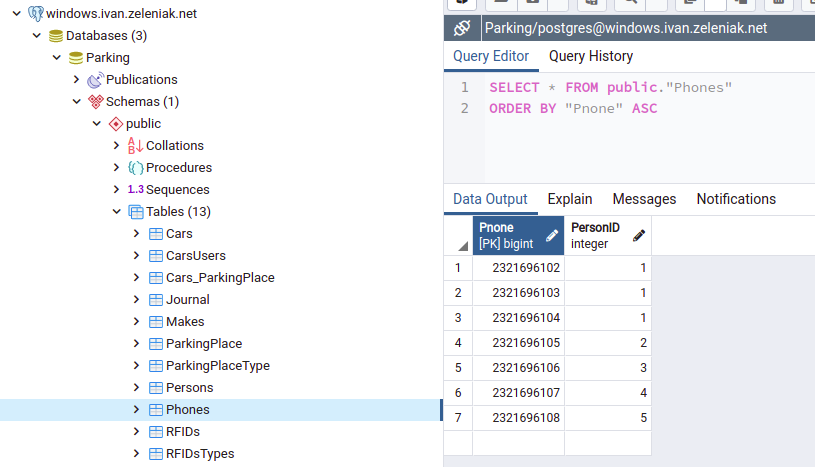


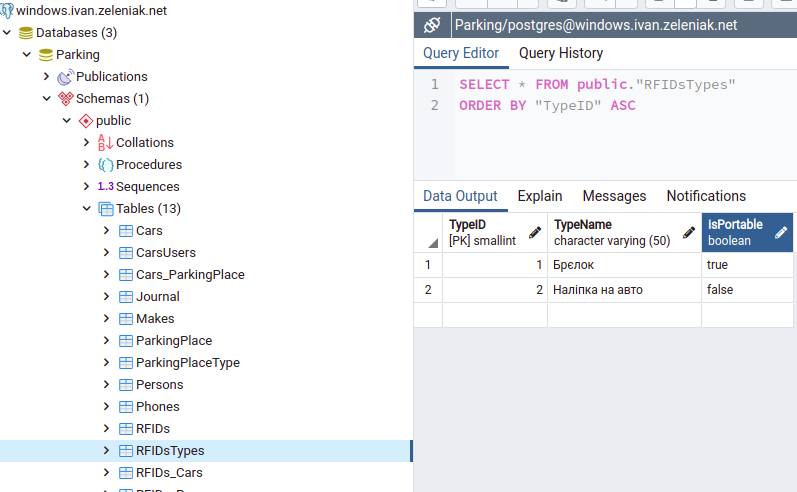


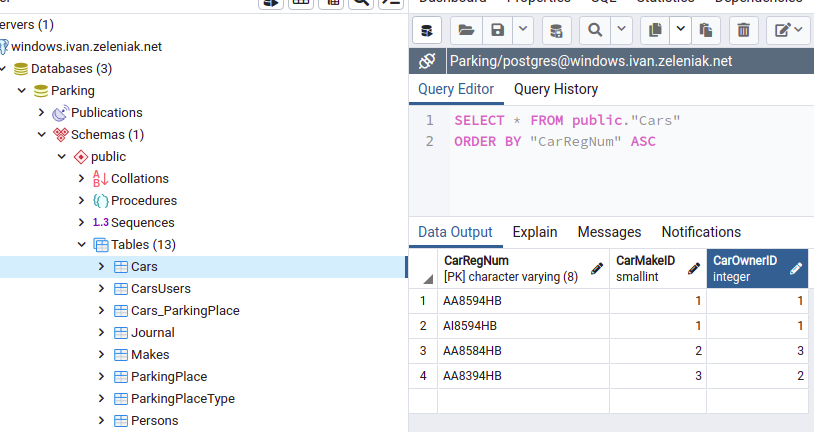


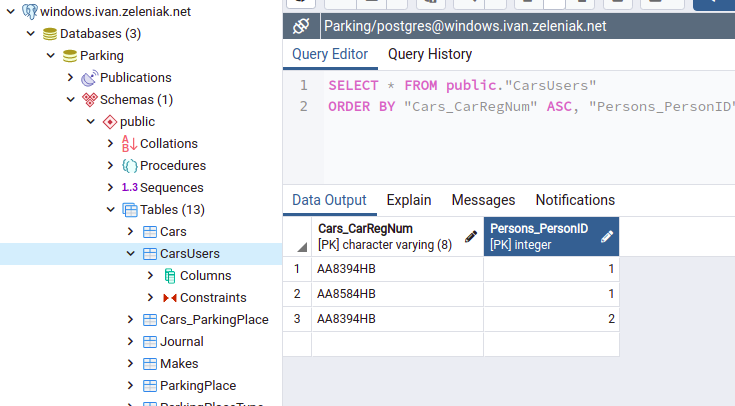


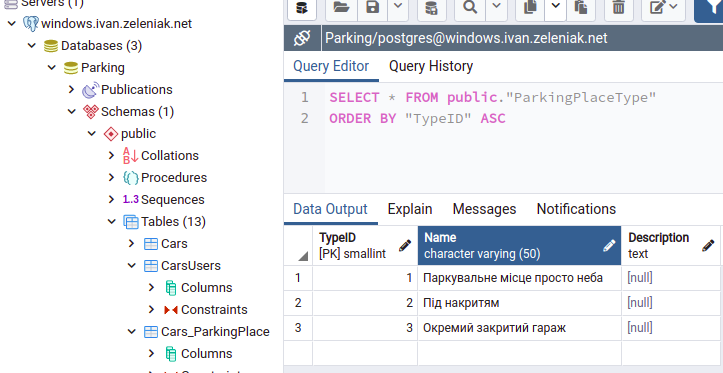


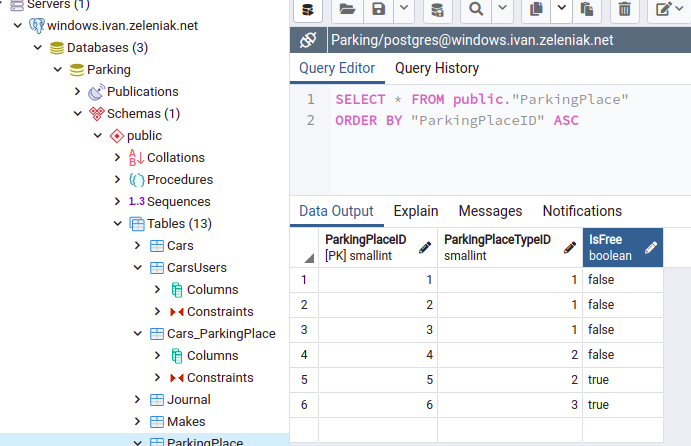


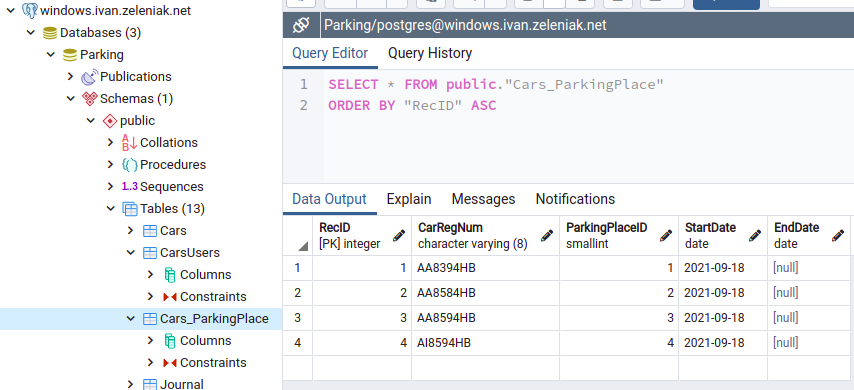


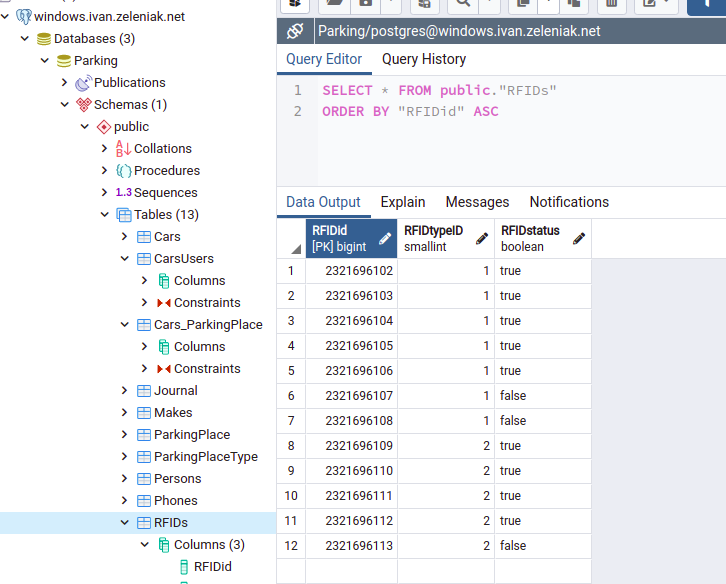












Файли що стосуються цієї лабораторної роботи завантажені в репозиторій git за адресою https://github.com/AmiriVANYCH/DataBase/tree/main/Lab1