

Міністерство освіти та науки України
Національний університет “Львівська політехніка”

Інститут прикладної математики
і фундаментальних наук
Кафедра прикладної математики

Курсова робота
з курсу
«Робота з великим базами даних»

Виконала:
студентка групи ПМ-31
Бабенко В. О.
Прийняв:
Любінський Б.Б.

Львів 2022

ЗМІСТ

| | |
|----------------------------------|----|
| ЗМІСТ | 2 |
| ОСНОВНІ ПОЛОЖЕННЯ..... | 3 |
| ПРЕДМЕТНА ОБЛАСТЬ | 4 |
| СТРУКТУРА БАЗИ ДАНИХ | 5 |
| ОГЛЯД ЗАСОСУНКУ | 9 |
| Список використаних джерел | 15 |

ОСНОВНІ ПОЛОЖЕННЯ

Метою даної курсової роботи являється вивчення способів проектування великих без даних.

Основні завдання включають в себе, розробку бази даних відповідно до предметної області, автоматизація її роботи з програмним забезпеченням

Для розробки бази даних цього типу використала декларативну мову програмування SQL. SQL — декларативна мова програмування для взаємодії користувача з базами даних, що застосовується для формування запитів, оновлення і керування реляційними БД, створення схеми бази даних та її модифікації, системи контролю за доступом до бази даних. За середовище розробки обрала програмний продукт Microsoft SQL Server Management Studio.

Для надання можливості користувачу взаємодіяти з базою даних, розробила за стосунком використовуючи графічний інтерфейс .NET Framework - Windows Forms, з використанням технології доступу до баз даних Entity Framework.

Entity Framework — це набір технологій в ADO.NET, які підтримують розробку програмно-орієнтованих на дані програмних додатків.

Для створення додатку обрала принцип Database First.

ПРЕДМЕТНА ОБЛАСТЬ

Варіант 2 “Оренда приміщень”.

Деяка організація здає приміщення в оренду іншим організаціям. Інформаційна система повинна проводити облік використання приміщень, нарахування оплати за оренду та розрахунки з орендарями. Кожне приміщення має свій номер та площу, воно є приміщенням певного типу (кімната, склад, тощо). Кожен тип приміщення має свою ціну з розрахунку на квадратний метр. Орендар – це фірма яка має назву, юридичну адресу, банк в якому має рахунок, директора, характеристику. Орендар орендує декілька приміщень, кожне з яких на певний термін. На початку кожного місяця повинен генеруватися рахунок на оплату за оренду для кожного орендаря. Сума рахунку повинна включати податок 20%. Кожен рахунок повинен бути оплачений на протязі 15 днів. За кожен прострочений день нараховується пеня в розмірі 0.2% від неоплаченої суми. Пеня повинна бути врахована в сумі чергового рахунку.

Система повинна враховувати надходження оплати від орендарів по рахунках.

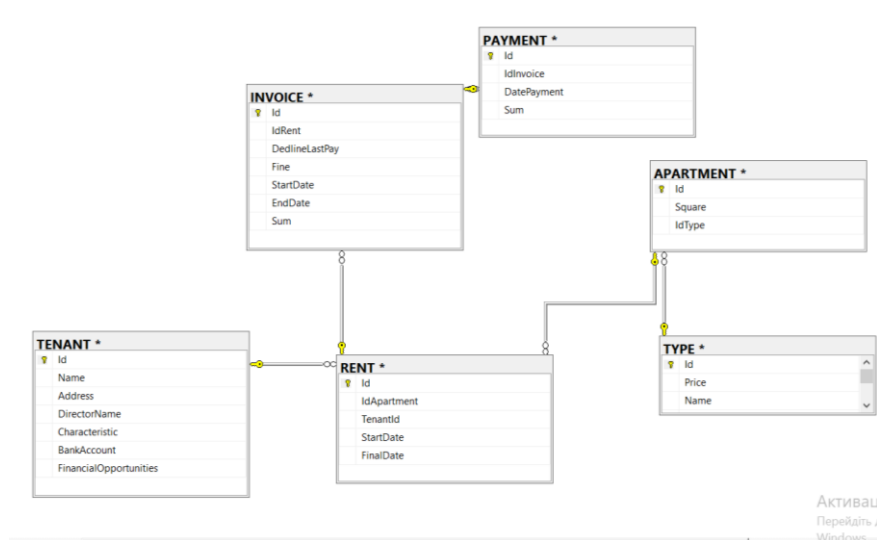
Кімната не може одночасно бути здана двом орендарям. Орендар, який має борг більший ніж сума трьохмісячної оренди не може отримувати в оренду нові приміщення.

Система повинна надавати наступні звіти :

- Список орендарів з інформацією які приміщення він орендує, місячною сумою оренди та сумою боргу.
- Список приміщень з вказанням орендаря на певний день.
- Довідка для орендаря (параметр) з вказанням приміщень які він коли-небудь орендував, рахунків на оплату та дат оплати цих рахунків.

СТРУКТУРА БАЗИ ДАНИХ

Створила базу даних відповідно до предметної області:



Згодом заповнила її відповідними даними.

Створила відповідні тригери для контролю бази даних.

--Приміщення не може одночасно бути зданим двом орендарям

CREATE OR ALTER TRIGGER RC

ON RENT

AFTER INSERT, UPDATE AS

BEGIN

DECLARE @IdApartment INT = (

SELECT inserted.IdApartment

FROM inserted

)

DECLARE @StartRent DATE = (

SELECT inserted.StartDate

FROM inserted

)

DECLARE @EndRent DATE = (

SELECT inserted.FinalDate

Львів 2022

```

FROM inserted

)

DECLARE @Idrent INT = (

    SELECT inserted.Id

    FROM inserted

)

IF EXISTS(

    SELECT *

    FROM RENT

    WHERE RENT. IdApartment= @IdApartment

    AND  ((@StartRent  >=  RENT.StartDate  AND  @StartRent  <=

RENT.FinalDate)

        OR

        (@EndRent >= RENT.StartDate AND @EndRent <= RENT.FinalDate)

        OR

        (@StartRent <= RENT.StartDate AND @EndRent >= RENT.FinalDate))

    AND  @RentId != RENT.Id

)

BEGIN

    ROLLBACK

END

END

```

GO

-- Орендар, який має борг більший ніж сума трьохмісячної орендної плати не може отримувати в оренду нові приміщення.

```

CREATE OR ALTER FUNCTION CS(@IdTenant INT) RETURNS NUMERIC

BEGIN

```

```

DECLARE @Sum NUMERIC = 0

SELECT @Sum = Sum(Invoice.Sum)

FROM BILL JOIN RENT ON INVOICE.idRent = RENT.Id

WHERE RENT.TenantId = @IdTenantId

RETURN ISNULL(@Sum, 0)

END

GO

CREATE OR ALTER FUNCTION PaymentSum(@IdTenant INT) RETURNS
NUMERIC

BEGIN

    DECLARE @Sum NUMERIC = 0

    SELECT @Sum = Sum(PAYMENT.Sum)

    FROM PAYMENT LEFT JOIN INVOICE ON PAYMENT.IdInvoice =
INVOICE.Id RIGHT JOIN RENT ON INVOICE.IdRent = RENT.Id

    WHERE RENT. IdTenant = @IdTenant

    RETURN ISNULL(@Sum, 0)

end

GO

CREATE OR ALTER FUNCTION MaxLimit(@IdTenant INT, @ContractDate
DATE) RETURNS NUMERIC

BEGIN

    DECLARE @Sum NUMERIC = 0

    DECLARE @PreContractDate DATE = DATEADD(MONTH, -3,
@ContractDate)

    SELECT @Sum = Sum(INVOICE.Sum)

    FROM BILL JOIN RENT ON INVOICEL.idRent = RENT.Id

```

```
WHERE RENT.IdTenant = @IdTenant

AND (INVOICE.endDate <= @ContractDate AND INVOICE EndDate >=
@PreContractDate)

RETURN ISNULL(@Sum, 0)

END

GO
```

```
CREATE or ALTER TRIGGER Create_Id
ON dbo.TYPE
INSTEAD OF INSERT
AS

    SET IDENTITY_INSERT [TYPE] OFF

    INSERT INTO dbo.[TYPE] (Id, Price, Name, counter)

    SELECT NEXT VALUE FOR Id, Price, Name, counter

    FROM inserted

    SET IDENTITY_INSERT [TYPE] ON

CREATE SEQUENCE seq

START WITH 1

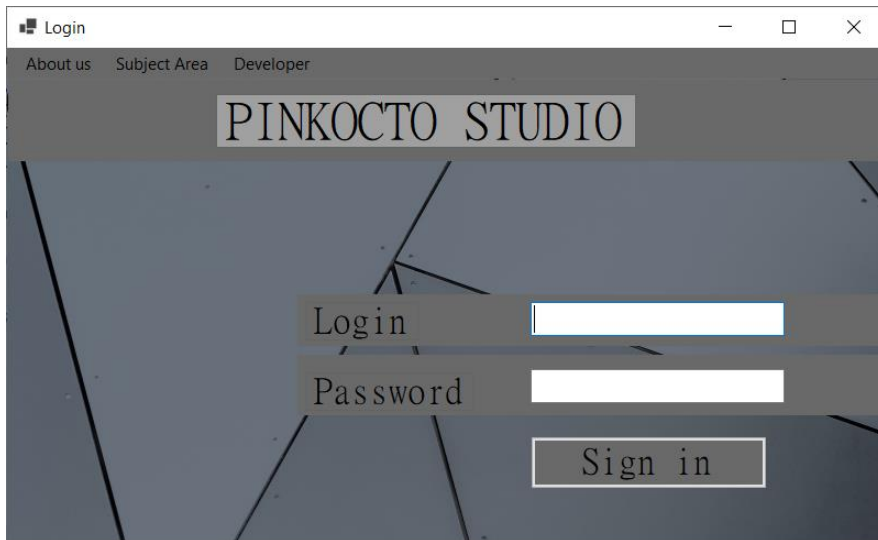
INCREMENT BY 1;

SET IDENTITY_INSERT TYPE ON
```


ОГЛЯД ЗАСОСУНКУ

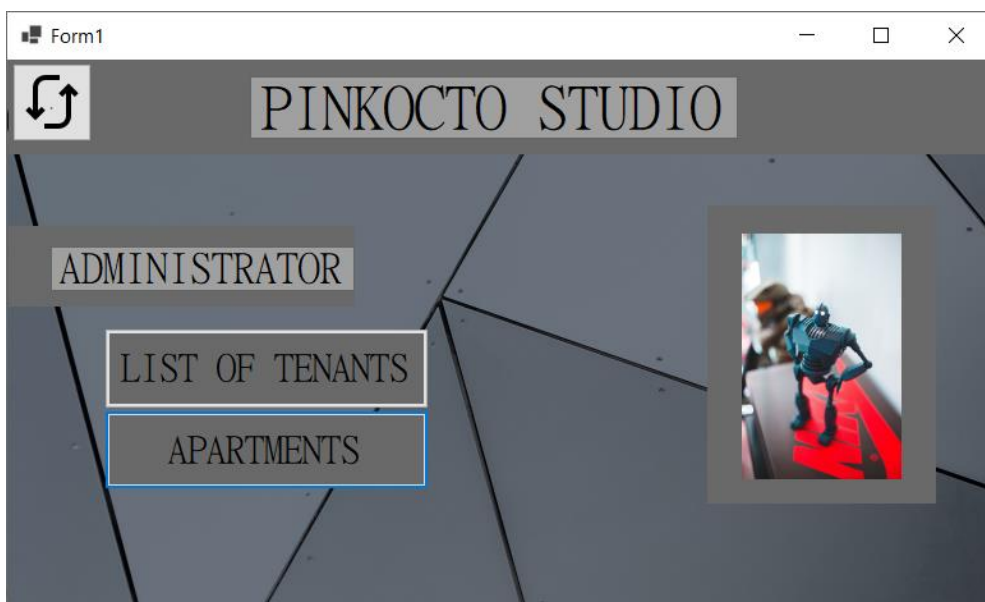
В даній роботі я розробила клієнтський застосунок на основі Entity Framework, з графічним інтерфейсом на Windows Forms.

Після запуску програми, користувача зустрічає вікно з автентифікацією, де потрібно ввести логін та пароль користувача, щоб отримати доступ перегляду до бази даних. Вхід може для адміністратора та для користувача відбувається аналогічно

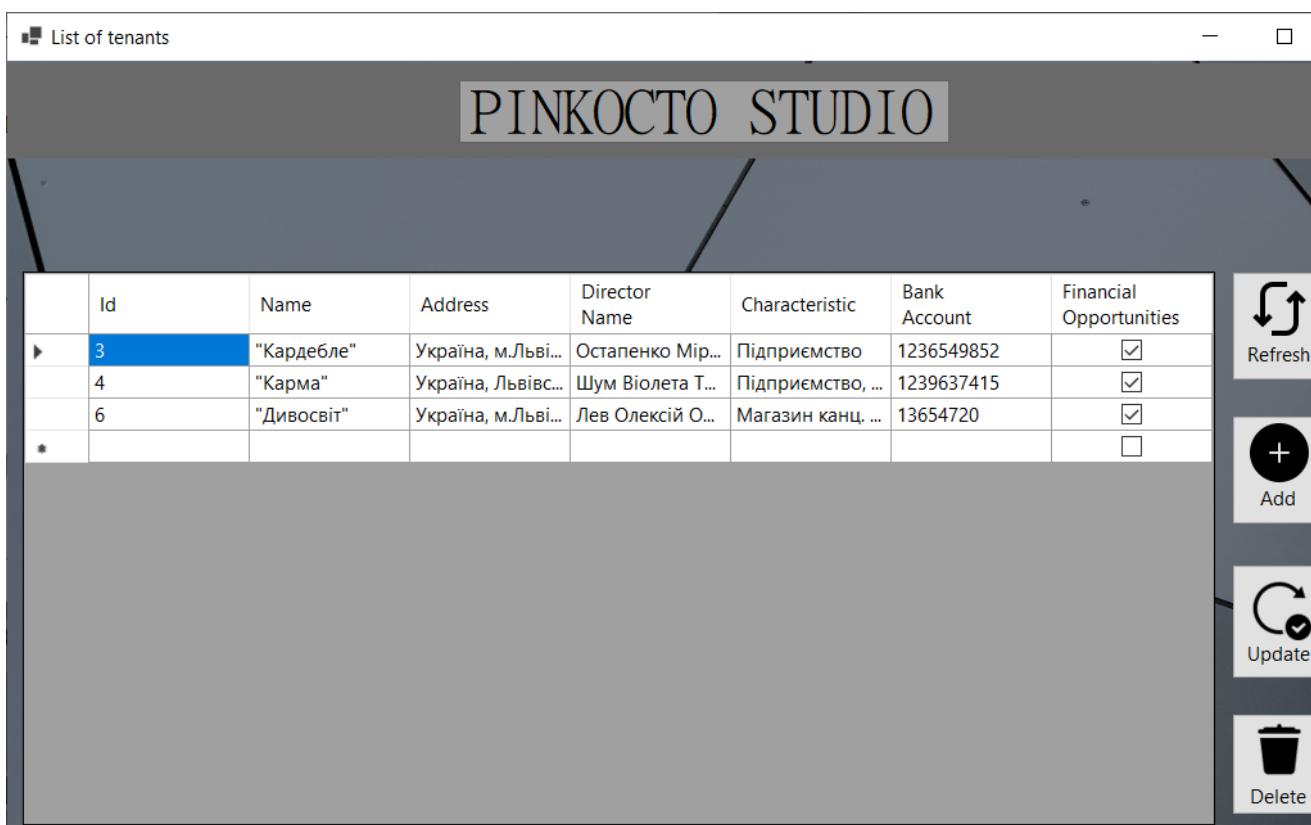


Після успішної автентифікації користувач та адміністратор потрапляють у головне меню. І далі в залежності від ролі функціонал змінюється.

При вході в акаунт адміністратора йому відкривається форма з переходами на списки орендарів, списки апартаментів та повернення на форму входу.



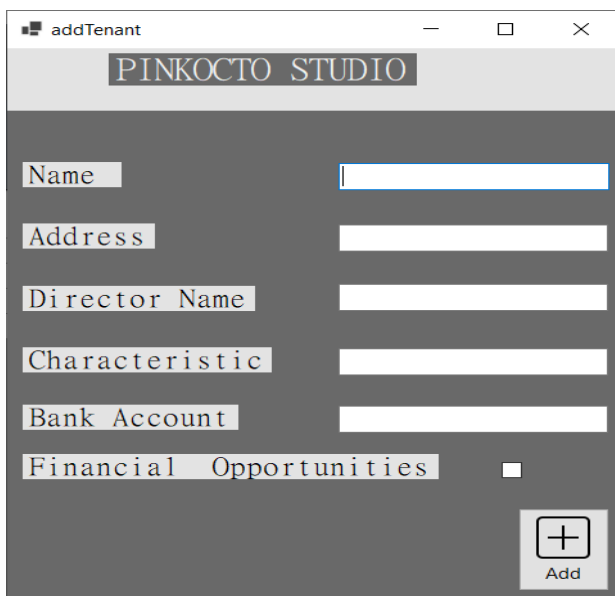
При відображенні списку орендарів є можливість додавати, міняти та видаляти дописи.



| | Id | Name | Address | Director Name | Characteristic | Bank Account | Financial Opportunities |
|---|----|------------|---------------------|------------------|-------------------|--------------|-------------------------------------|
| ▶ | 3 | "Кардебле" | Україна, м.Львів... | Остапенко Мір... | Підприємство | 1236549852 | <input checked="" type="checkbox"/> |
| | 4 | "Карма" | Україна, Львівс... | Шум Віолета Т... | Підприємство, ... | 1239637415 | <input checked="" type="checkbox"/> |
| | 6 | "Дивосвіт" | Україна, м.Львів... | Лев Олексій О... | Магазин канц. ... | 13654720 | <input checked="" type="checkbox"/> |
| * | | | | | | | <input type="checkbox"/> |

Для додавання нового користувача варто натиснути кнопку Add, що викликає форму додавання.

На самій формі відображаються, поля необхідні для заповнення ключовими даними про користувача.



addTenant

PINKOCTO STUDIO

Name

Address

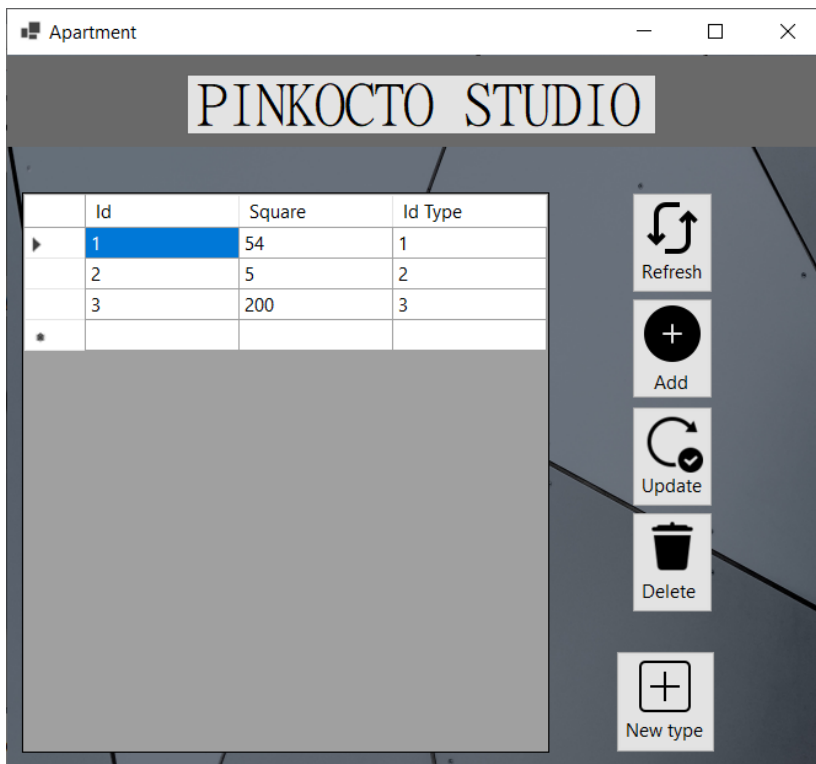
Director Name

Characteristic

Bank Account

Financial Opportunities ☐

При відображенні списку приміщень реалізовані аналогічні можливості та додаткова можливість додавати новий тип приміщення.



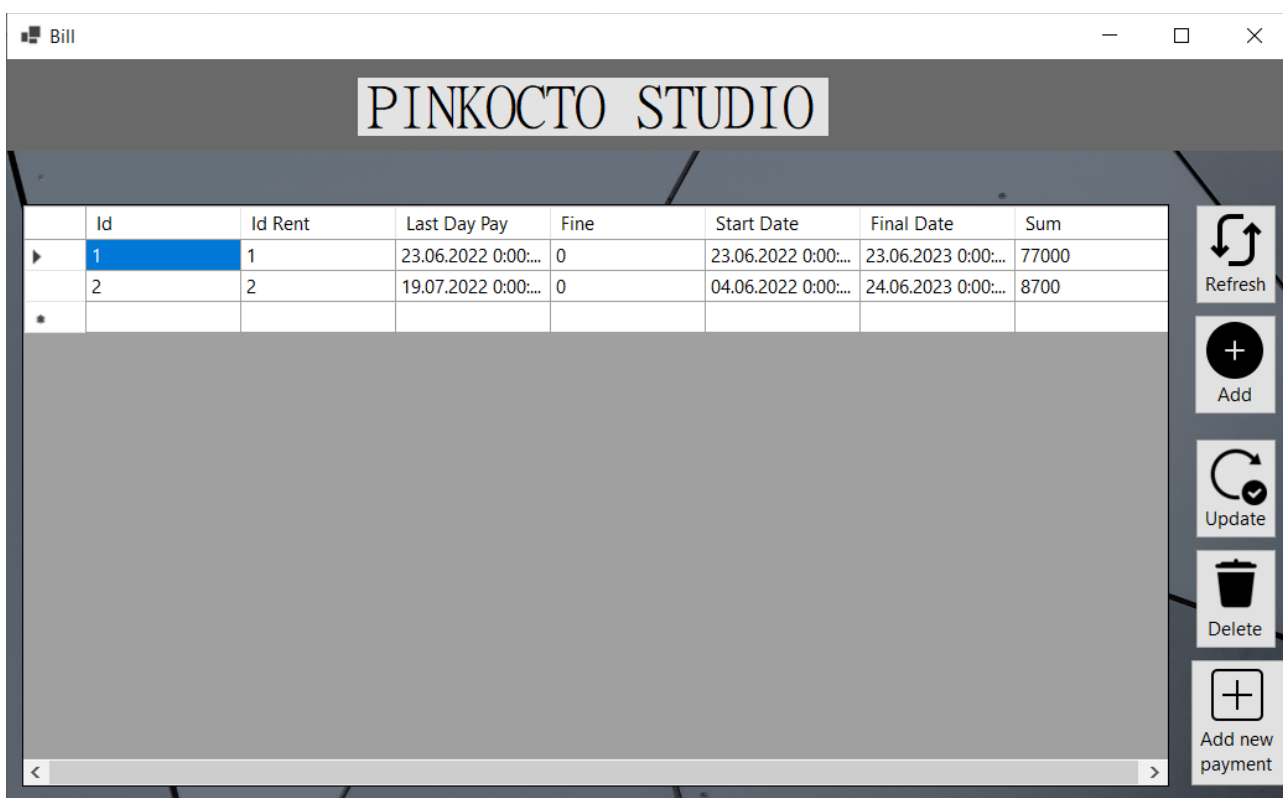
З можливістю редагування та видалення.



При вході в акаунт користувача відкривається форма з кнопками що відображають, списки рахунків, орендування нового приміщення та повернення на форму входу.



При переході на список рахунків, ми бачимо відображення списків, з можливістю додавання нових платежів через додаткову форму.



З можливістю редагування даних.

| PINKOCTO STUDIO | | | | | |
|-----------------|----|------------|---------------------|-------|---------|
| | Id | Id Invoice | Date Payment | Sum | |
| ▶ | 1 | 1 | 25.06.2022 0:00:... | 70000 | Refresh |
| | 2 | 1 | 27.06.2022 0:00:... | 7000 | |
| | 3 | 2 | 27.06.2022 0:00:... | 8500 | |
| • | | | | | Add |
| | | | | | Update |
| | | | | | Delete |

При переході на пункт орендування нового приміщення, відображається список орендованих приміщень з можливістю додавання, редагування та видалення.

| PINKOCTO STUDIO | | | | | | |
|-----------------|----|--------------|-----------|---------------------|---------------------|---------|
| | Id | Id Apartment | Id Tenant | Start Date | Final Date | |
| ▶ | 1 | 1 | 3 | 15.06.2022 0:00:... | 15.06.2023 0:00:... | Refresh |
| | 2 | 2 | 3 | 22.04.2022 0:00:... | 22.04.2023 0:00:... | |
| | 3 | 3 | 6 | 23.06.2022 0:00:... | 23.06.2023 0:00:... | |
| • | | | | | | Add |
| | | | | | | Update |
| | | | | | | Delete |

Орендування нового приміщення відбувається через кнопку Add, що переносить на форму додавання.

The screenshot shows a software window titled 'addRent' with the 'PINKOCTO STUDIO' logo. It contains four input fields: 'Id Apatment' with the value 'b', 'Id Tenant' with the value '0', 'Start Date', and 'Final Date'. Below the date fields are two calendar pop-ups for June 2022. In both calendars, the 24th is selected and highlighted with a blue box. At the bottom center of the form is a large button with a plus sign and the text 'Add'.

У формі варто задати Дані про приміщення, Орендаря, та дати початку та кінця оренди.

Список використаних джерел

1. Стаття про Database First [Електронний ресурс] – Режим доступу: [Database First - EF6 | Microsoft Docs](#);
2. Стаття про SQL [Електронний ресурс] – Режим доступу: https://uk.wikipedia.org/wiki/SQL#%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_SQL ;
3. Стаття про EF Core [Електронний ресурс] – Режим доступу: [Getting Started with EF Core](#) ;