**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ**

**«МАШИНОБУДІВНИЙ ФАХОВИЙ КОЛЕДЖ**

**ДНІПРОВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ**

**ІМЕНІ ОЛЕСЯ ГОНЧАРА»**

(повне найменування вищого навчального закладу)

Циклова комісія прикладної математики

(повна назва кафедри, циклової комісії)

**КУРСОВА РОБОТА**

з навчальної дисципліни «Бази даних та інформаційні системи»

на тему:

База даних «Банківські вклади».

Студента 4 курсу, групи ПМ-21

Спеціальність \_\_113 Прикладна математика\_\_ Освітньо-професійна програма «Прикладна математика»

Ткаченко Прохор Климович \_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище та ініціали**)**

Керівник: викладач Ізмайлова Марія Костянтинівна

(посада, прізвище та ініціали)

Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М. К. Ізмайлова

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.В. Таранова

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

м. Дніпро – 2024 рік

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ**

**«МАШИНОБУДІВНИЙ ФАХОВИЙ КОЛЕДЖ**

**ДНІПРОВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ**

**ІМЕНІ ОЛЕСЯ ГОНЧАРА»**

Відділення \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_економічне \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_

Циклова комісія \_ прикладної математики

Освітньо-професійний ступень «фаховий молодший бакалавр»

Спеціальність \_\_113 Прикладна математика\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Освітньо-професійна програма «Прикладна математика»

Група \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ПМ-21\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ЗАТВЕРДЖУЮ**

Голова циклової комісії прикладної математики \_\_\_\_\_\_\_\_\_М. К. Ізмайлова

«\_20\_\_» \_\_\_\_\_\_\_09\_\_\_\_\_\_\_2024 року

**З А В Д А Н Н Я**

**НА КУРСОВУ РОБОТУ СТУДЕНТУ**

\_\_з навчальної дисципліни «**Бази даних та інформаційні системи**» \_\_\_\_\_\_

Ткаченко Прохор Климович\_\_\_\_

(прізвище, ім’я, по батькові)

Тема роботи База даних «Банківські вклади» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Керівник проекту \_\_\_\_\_\_\_\_\_\_\_\_ Ізмайлова Марія Костянтинівна

( прізвище, ім’я, по батькові)

Дата видачі завдання \_\_\_\_\_\_\_\_\_\_\_\_ 20.09.2024 р.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Строк подання студентом проекту \_\_\_01.12.2024 р.

Зміст пояснювальної записки (перелік питань, які потрібно розробити)

1. Теоретична частина
   1. Аналіз предметної області та вимог користувача до проектованої бази даних.
   2. Виділення об’єктів предметної області та встановлення відношень між об’єктами.
   3. Побудова концептуальної моделі даних проектованої бази даних.
   4. Спрощення концептуальної моделі даних проектованої бази даних.
   5. Удосконалення концептуальної схеми предметної області.
   6. Виділення таблиць реляційної бази даних. Перевірка таблиць за допомогою концепції послідовної нормалізації.
   7. Побудова ER – діаграми реляційної бази даних.
2. Практична частина:
   1. Створення бази даних, таблиць.
   2. Введення даних в таблиці.
   3. Модифікація таблиць бази даних.
   4. Формулювання запиту.
   5. Опис основних запитів і алгоритмів обробки даних.
   6. Виконання запитів та одержування звітів за даними таблиць бази даних.
   7. Реалізація і тестування системи.
3. Висновки.
4. Перелік використаної літератури.

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |
| --- | --- | --- | --- |
| **№**  **з/п** | **Назва етапів курсової роботи** | **Строк виконання етапів роботи** | **Примітка** |
|  | Вступ | 01.10.2024 |  |
| 1 | Теоретична частина |  |  |
| 1.1 | Аналіз предметної області та вимог користувача до проектованої бази даних. | 03.10.2024 |  |
| 1.2 | Виділення об’єктів предметної області та встановлення відношень між об’єктами. | 08.10.2024 |  |
| 1.3 | Побудова концептуальної моделі даних проектованої бази даних. | 10.10.2024 |  |
| 1.4 | Спрощення концептуальної моделі даних проектованої бази даних. | 15.10.2024 |  |
| 1.5 | Удосконалення концептуальної схеми предметної області. | 17.10.2024 |  |
| 1.6 | Виділення таблиць реляційної бази даних. | 19.10.2024 |  |
| 1.7 | Перевірка таблиць за допомогою концепції послідовної нормалізації. | 22.10.2024 |  |
| 2 | Розрахункова частина |  |  |
| 2.1 | Побудова ER – діаграми реляційної бази даних. | 24.10.2024 |  |
| 2.2 | Створення бази даних, таблиць та індексів. | 27.10.2024 |  |
| 2.3 | Введення даних в таблиці. | 05.11.2024 |  |
| 2.4 | Формулювання запитів. | 08.11.2024 |  |
| 2.5 | Опис основних запитів і алгоритмів обробки даних. | 14.11.2024 |  |
| 2.6 | Виконання запитів та одержування звітів за даними таблиць бази даних. | 18.11.2024 |  |
| 2.7 | Реалізація і тестування системи. | 20.11.2024 |  |
| 3 | Висновки. | 25.11.2024 |  |
| 4 | Оформлення роботи. | 30.11.2024 |  |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_П.К. Ткаченко\_\_

(підпис )

Керівник роботи М. К. Ізмайлова\_

( підпис )

***Відгук***

***на курсову роботу з навчальної дисципліни***

«Бази даних та інформаційні системи»

***студента*** Ткаченка Прохора Климовича

***спеціальність*** 113 Прикладна математика ***Група:*** ПМ-21

***Тема роботи:***

|  |
| --- |
| База даних «Банківські вклади» |
|  |
| ***Теоретична частина*** |
| *Теоретична частина курсової роботи викладена логічно і послідовно.* | |
| *Тема курсової роботи розкрита, висвітлені основні складові теми:* | |
| *виконано аналіз предметної області, побудована концептуальна модель* | |
| *даних проектованої бази даних, проведена нормалізація виділених таблиць.* | |
|  | |
|  |
| ***Практична частина***  *Практична частина виконана вірно, вона містить* *користувацький інтерфейс,* |
| *має ряд сформованих форм і звітів, які можна роздрукувати в* |
| *зручному вигляді. Створення таблиць і запитів виконано в СУБД Access.* |
|  |
|  |
|  |
| ***Висновки*** |
| *Курсова робота виконана в повному обсязі і задовольняє вимогам методичних* |
| *вказівок.* |
| *Курсова робота допускається до захисту.* |
|  |
|  |
| ***„\_\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*** *2024 р****. \_\_\_\_\_\_\_\_\_\_\_\_\_*** *М. К. Ізмайлова* |

**РЕФЕРАТ**

**Курсова робота: 42 сторінки, 48 рисунків, 6 таблиць, 7 джерел**

**Мета роботи:** проектування та реалізації бази даних для підприємства з банківськими владами.

**Предмет дослідження:** бази даних для підприємства з банківськими вкладами.

**Об’єкт дослідження:** підприємства з банківських вкладів.

**Методи дослідження:** інформаційний, дослідження, програмування

# **ЗМІСТ**

[**ЗМІСТ** 5](#_Toc183337663)

[**ВСТУП** 6](#_Toc183337664)

[**1.** **ПРОЕКТУВАННЯ БАЗИ ДАНИХ** 7](#_Toc183337665)

[1.1. Опис об’єктів предметної області 7](#_Toc183337666)

[1.2. Аналіз предметної області 11](#_Toc183337667)

[1.3. Концептуальна схема предметної області 11](#_Toc183337668)

[1.4. Перевірка таблиць за допомогою концепції послідовної нормалізації 14](#_Toc183337669)

[1.5. Побудова ER - діаграми реляційної бази даних 15](#_Toc183337670)

[**2. РЕАЛІЗАЦІЯ БАЗИ ДАНИХ** 16](#_Toc183337671)

[2.1. Вибір інструментальних засобів 16](#_Toc183337672)

[2.2. Побудова таблиць БД за допомогою СУБД MS Access 17](#_Toc183337673)

[2.3. Проектування запитів 22](#_Toc183337674)

[2.4. Проектування форм та звітів 26](#_Toc183337675)

[2.5. Розробка інтерфейсу головної кнопкової форми 27](#_Toc183337676)

[**3.** **РОБОТА В MYSQL** 30](#_Toc183337677)

[3.1. Побудова таблиць засобами MySQL 31](#_Toc183337678)

[3.2. Використання запитів на мові SQL 38](#_Toc183337679)

[**ВИСНОВОК** 41](#_Toc183337680)

[**Список використаної літератури** 42](#_Toc183337681)

# **ВСТУП**

Темою курсової роботи є база даних «Банківкі вклади».

В нинішніх умовах швидкий доступ до правдивої інформації є важливою складовою сучасності. А бази даних відіграють у цьому дуже важливу роль, оскільки саме завдяки ним дані структуруються в одне ціле, а також користувачу надається швидкий і зручний спосіб отримання інформації. Кожна організація чи підприємство потребують у структуруванні даних з метою зберігання, адміністрування, використання та зручного редагування інформації.

Кількість інформації яку отримує будь яке підприємство чи організація на протязі робочого дня є дуже великою, і тому використання баз даних є необхідним для коректної і зручної її обробки.

Метою даної курсової роботи є вивчення та здобуття навичок у студентів з проектування та реалізації бази даних, розробки програмного забезпечення з їх управління з використанням СУБД. Під час виконання даної курсової роботи потрібно створити та розробити інформаційну базу даних для підприємства з банківських влкадів.

1. **ПРОЕКТУВАННЯ БАЗИ ДАНИХ**
   1. . Опис об’єктів предметної області

У даній курсовій роботі буде розглядатися предметна область «Банківські вклади».

У процесі проектування бази даних виділяються два основні етапи: розробка схеми даних і створення її фізичного втілення. Рекомендується розпочати створення бази даних саме з її проектування та розробки. Основною метою проектування є визначення структури таблиць, логічних зв'язків між ними та визначення атрибутів кожного стовпця, типу даних і розміру, а також ключових полів таблиць. Подальше встановлення структурних зв'язків між цими об'єктами сприяє створенню схеми даних.

Схема даних - це спосіб представлення інформації про предметну область користувачам та деталей її збереження в кожній окремій базі даних. Фізична організація бази даних включає в себе методи та засоби для збереження даних в зовнішній пам'яті. Аналіз цих даних допомагає визначити функціональні залежності між нашими атрибутами, які походять з різних джерел завантаження бази даних, і сприяє створенню концептуальної схеми.

Внутрішня модель служить для відображення логічної моделі даних, яка показує, як дані розміщуються в базі даних, як вони упорядковуються, як організовані зв'язки і як можливо вибирати інформацію та інше.

При створенні бази даних важливо правильно виділяти об'єкти предметної області та структурувати таблиці, в яких ці об'єкти можна визначити. Об'єкти в предметній області характеризуються набором атрибутів. Отже, правильне виділення об'єктів і їх інтерпретація є важливим аспектом створення та розвитку бази даних.

Розробка інфологічної моделі є найважливішим етапом у процесі проектування та створення бази даних. Метою створення такої моделі є забезпечення зручних для людини методів збору та подання інформації, яка має бути збережена в базі даних. Основними конструктивними компонентами цих моделей є сутності, зв'язки між ними та їх атрибути.

На основі аналізу предметної області та постановки завдання було виділено наступні об’єкти:

1. Клиент: ID клиент, Имя, Адрес, Телефон, Электронная работа.
2. Вклад: Сумма вклада, Процентная ставка, Дата открытия, Дата закрытия, ID кредита, ID клиента.
3. Тип вклада : ID типа вклада, Название типа вклада, Описание условия вклада
4. Кредит: ID кредит, Тип кредита, Сумма кредита, Дата начала, Дата окончания, ID Клиента.
5. Счёт: Номер счёта, Тип счёта, Баланс, Дата открытия, ID Клиента.
6. Транзакция: ID транзакция, Номер счёта, дата транзакции, Сумма, Тип транзакция.

Зобразимо кожний графічний об’єкт і його властивості (див. [Рис. 1.1](#Рисунок_1_1)-[1.5](#Рисунок_1_8))

1. «Кліент»:

Кліент

Рисунок 1.1 – Зображення зв’язку «Об’єкт - Властивість» для об’єкта «Кліент»

1. «Вклад»

Вклад

Рисунок 1.2 – Зображення зв’язку «Об’єкт - Властивість» для об’єкта «Вклад»

1. «Кредит»

Кредит

Рисунок 1.3 – Зображення зв’язку «Об’єкт - Властивість» для об’єкта «Кредит»

1. «Рахунок»

Рахунок

Рисунок 1.4 – Зображення зв’язку «Об’єкт - Властивість» для об’єкта «Рахунок»

1. «Тип вклада»

Тип вкладу

Рисунок 1.5 – Зображення зв’язку «Об’єкт - Властивість» для об’єкта «Тип вкладу»

1. «Транзакция»

Транзакція

Рисунок 1.6 – Зображення зв’язку «Об’єкт - Властивість» для об’єкта «Транзакция»

* 1. Аналіз предметної області

Проаналізувавши предметну область було прийнято рішення створити шість відношень (таблиць): у таблиці «Клиент» будуть знаходитися дані які стосуються виключно Клиентів; у таблиці «Вклад» будуть відображатись дані про вклади які стосуються ліше вкладу (сумма вкладу, дати відкриття\закріття, ..); у таблиці «Кредит» відображатимуться данні необхідні для кредитування; у таблиці «Счёт» буде надано інформацію про рахунок кліента; у таблиці «Транзакция» надаватиметься інформація про будь які дії, які проводились з рахунком або «Счёт»;у таблиці «Тип вклада» назва видів вкладу та їх повний опис.

Отже, у наслідок аналізу предметної області було виділено та встановлено наступні об’єкти:Клиент, Вклад, Кредит, Счёт, Транзакция, Тип вклада .

* 1. Концептуальна схема предметної області

Одним з перших та одним з головних етапів у всьому процесі проектування бази даних є розробка інфологічної моделі.

Концептуальна модель представляє собою абстрактне описання основних інформаційних потреб, які лежать в основі структури бази даних. Вона включає в себе лише основні концепції та головні твердження про природу бази даних. Після створення концептуальної схеми необхідно визначити атрибути або характеристики для кожної сутності. У реляційних базах даних кожній сутності відповідає відповідна таблиця.

Отже, концептуальна схема описує структуру всієї бази даних для групи користувачів і приховує внутрішні деталі фізичного світу та цілі опису сутностей, способи їх взаємодії.

Створимо концептуальну модель для нашої предметної області ([рис.1.7](#Рисунок_1_9)).

Рисунок 1.7. Графічне зображення концептуальної схеми предметної області

Транзакція

Тип вкладу

Рахунок

Кліент

Кредит

Вклад

* 1. Перевірка таблиць за допомогою концепції послідовної нормалізації

Нормалізація - це процес декомпозиції відношень у базі даних з метою виключення аномалій модифікацій, надлишкових даних й приведення відношень до нормальних форм.

Нормальна форма – це властивість відношення у реляційній моделі даних, що характеризує його з точки зору надмірності, яка може спровокувати виникнення логічних помилок або зміни даних. Така форма визначається як сукупність вимог, які має задовольняти таблиця у базі даних.

Аномалії усуваються шляхом розбиття одного відношення на два або більше нових відношень. Також можна виділити два основних види аномалій: аномалію видалення та аномалію вставки.

Один із внутрішніх обмежень, які встановлює реляційна модель, - це унікальність кожного запису в кожній таблиці, тобто кожен запис має ідентифікатор, за яким його можна однозначно відрізнити від інших записів таблиці. Цей ідентифікатор може бути обраний серед атрибутів або їх сукупності.

Можна сказати, що в нашій курсовій роботі усі відношення знаходяться в першій нормальній формі, адже значення кожного атрибута є атомарним – воно є неподільним, не є списком, та немає вкладеності значень.

Належність відношень у даній курсовій роботі до другої нормальної форми можна визначити по тому, що відношення вже знаходяться в першій нормальній формі і усі неключові атрибути функціонально повно залежать від первинного ключа і повністю ним визначаються.

Наші відношення також належать і до третьої нормальної форми тому, що таблиці вже знаходяться у другій нормальній формі, і в них відсутні транзитивні залежності між атрибутами.

Отже, в даній курсовій роботі всі таблиці знаходяться в першій, другій і третій нормальних формах.

* 1. Побудова ER - діаграми реляційної бази даних

ER-модель – це модель даних, яка призначена для спрощення процесу проектування бази даних. Вона представляє базу даних у вигляді графічних діаграм, а також візуалізує процес, що визначає предметну область. З такої моделі можуть бути створені всі види бази даних: реляційні, ієрархічні, мережні, об’єктні.

В основі поняття ER-моделі лежать такі поняття як: сутність, зв’язок, атрибут. Для великих баз даних побудова цієї моделі дозволяє уникати помилок проектування.

Наступним кроком виконання курсової роботи є побудова реляційної схеми бази даних з самої ER-моделі. Для кожного простого об’єкта та його одиничних властивостей будується таблиця, атрибутами якої є ідентифікатор об’єкта і реквізити, що відповідають кожному з властивостей.

Для нашої предметної області побудова реляційної схеми бази даних буде виглядати таким чином ([рис. 1.8](#Рисунок_1_10)):

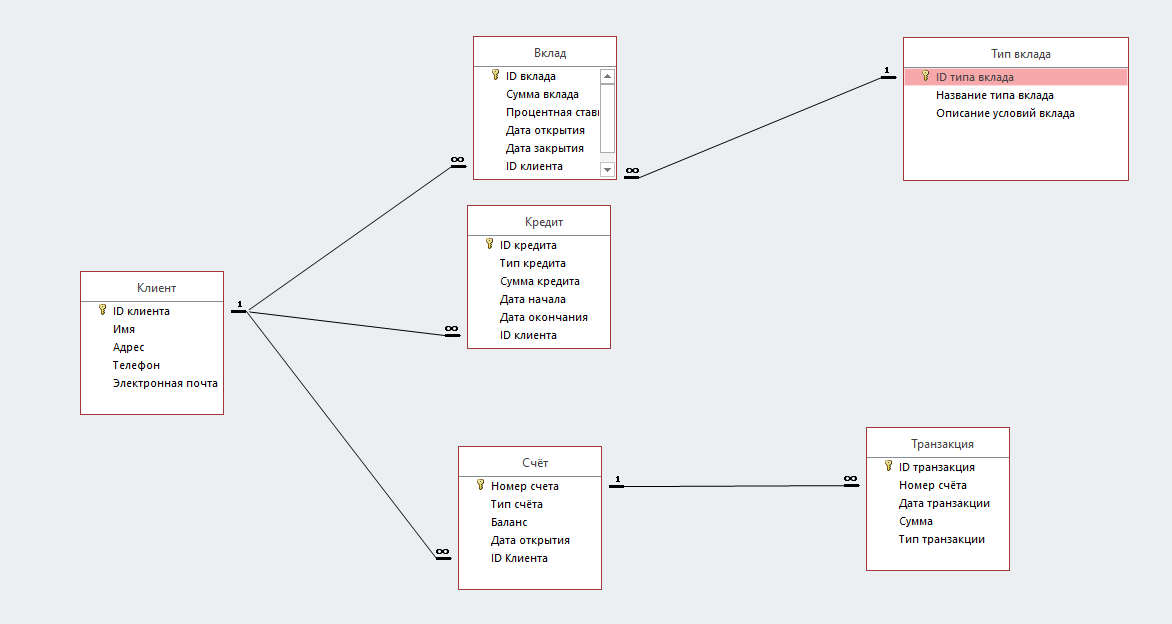


Рисунок 1.8. – ER-діаграма бази даних бази даних «банківські вклади та послуги»

# **2. РЕАЛІЗАЦІЯ БАЗИ ДАНИХ**

* 1. Вибір інструментальних засобів

Обрання правильної системи управління базами даних (СУБД) є критичним кроком у процесі створення бази даних, оскільки воно має вирішальний вплив на розвиток та функціонування інформаційної системи.

Теоретично, при здійсненні цього вибору потрібно брати до уваги десятки чинників, але на практиці розробники керуються лише власною інтуїцією і декількома найважливішими критеріями, до яких відносяться:

* тип моделі даних, яку підтримує дана СУБД;
* запас функціональних можливостей для подальшого розвитку інформаційної системи;
* ступінь оснащеності СУБД інструментарієм для персоналу
* адміністрування даними;
* зручність і надійність СУБД в експлуатації;
* наявність фахівців по роботі з конкретною СУБД;
* вартість СУБД і додаткового програмного забезпечення.

Практично всі існуючі системи управління базами даних (СУБД) обладнані інструментами розробки додатків, які можуть бути використані як програмістами, так і досвідченими користувачами для створення процедур, які автоматизують управління та обробку даних. В сучасних інформаційних системах СУБД є невід'ємною частиною для роботи з базами даних. Система керування базами даних (СКБД) - це комплекс програмних та технічних засобів, що дозволяють визначати, створювати, маніпулювати, контролювати, управляти та використовувати бази даних.

Однією з найвідоміших СУБД є Microsoft Access. Ця СУБД наділена різноманітними можливостями для управління реляційними базами даних, і вона підтримує архітектуру клієнт-сервер та відома також як SQL база даних.

Однією з причин, які сприяють популярності Microsoft Access, є його розширений набір функцій, включаючи можливість створення зв'язаних запитів, звітів, сортування за різними полями і зв'язок з зовнішніми таблицями та іншими базами даних. Важливо відзначити, що навіть користувачі без попереднього досвіду роботи з СУБД можуть легко освоїти Microsoft Access або ті, хто має обмежені навички в цій області.

Microsoft Access також надає максимальну свободу визначення типів даних, включаючи текст, числові дані, дати, час, грошові значення, малюнки, звуки і електронні таблиці. Користувачі можуть налаштовувати формати зберігання даних для відображення на екрані або друку, щоб переконатися, що база даних містить лише коректні дані, застосовуючи умови валідації різного рівня складності.

Крім того, Microsoft Access є інтегрованим додатком для операційної системи Windows і підтримує динамічний обмін даними (DDE) та об'єктноорієнтований зв'язок та використання об'єктів (OLE). DDE дозволяє обмінюватися даними між Microsoft Access та іншими програмами, які підтримують DDE. Завдяки OLE можна вставляти об'єкти з інших програм у базу даних Access, такі як зображення, діаграми, електронні таблиці та документи.

Крім цього, Microsoft Access використовує потужну мову запитів SQL (Structured Query Language) для обробки даних у базових таблицях. Використовуючи SQL, можна витягти необхідну інформацію з однієї чи декількох таблиць для вирішення конкретних завдань. І навіть без глибокого розуміння SQL, Microsoft Access полегшує завдання обробки даних, завдяки налаштованим зв'язкам між таблицями.

* 1. Побудова таблиць БД за допомогою СУБД MS Access

Створюючи базу даних Access, дані зберігаються в таблицях – тематичних списках, що складаються з рядків і стовпців.

Опишемо фізичну модель бази даних за допомогою її структури на мові опису даних СУБД MS Access. Для зберігання даних було створено таблиці, структура яких наведена в рисунках 2.1–2.6.

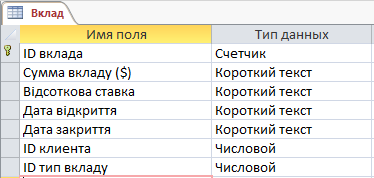


Рисунок 2.1- Структура таблиці «Вклад».

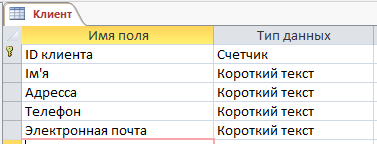


Рисунок 2.2 – Структура таблиці «Клиент».

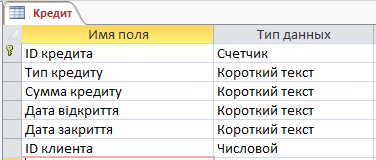


Рисунок 2.3 – Структура таблиці «Кредит».

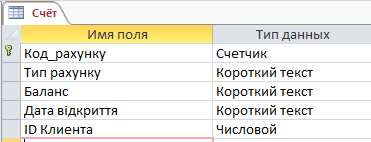


Рисунок 2.4- Структура таблиці «Рахунок».

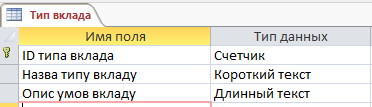


Рисунок 2.5 – Структура таблиці «Тип вкладу».

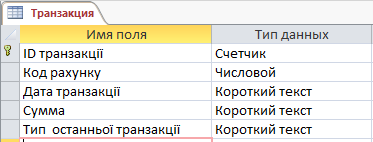
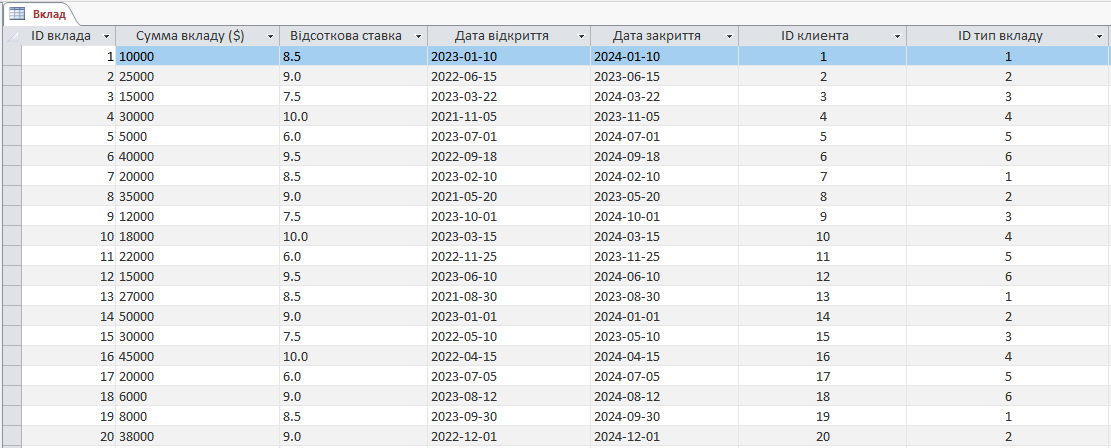


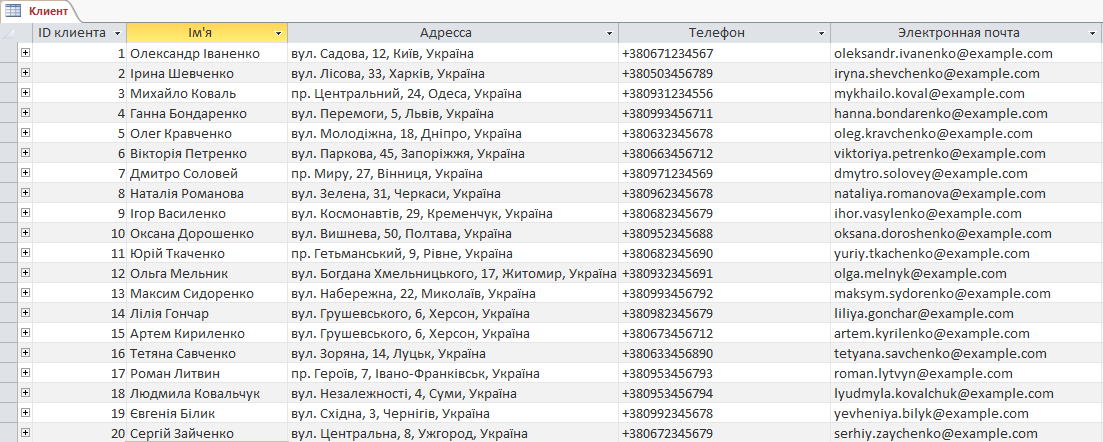
Рисунок 2.6- Структура таблиці «Транзакция».

Після цього в середовищі СУБД MS Access було виконано заповнення інформацією таблиць, які призначені для зберігання нормативно-довідкової інформації, а потім таблиць, призначених для зберігання оперативно-облікової інформації (Табл. 2.1-2.6).

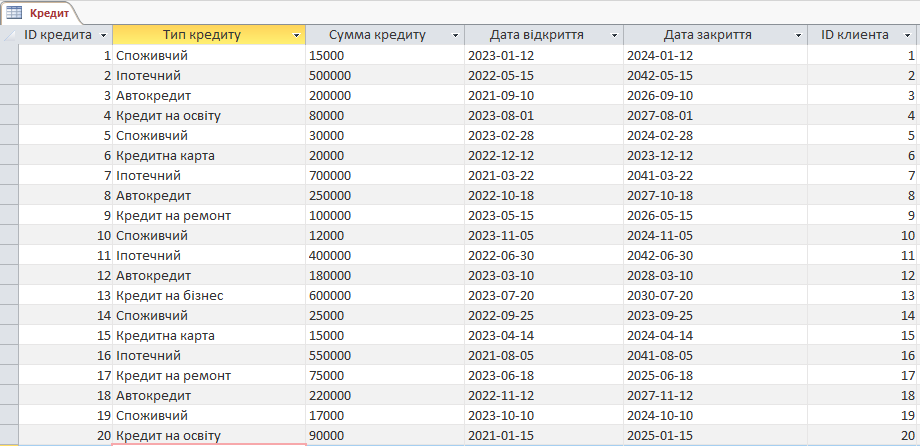
Таблиця 2.1 – Інформація таблиці «Вклад».



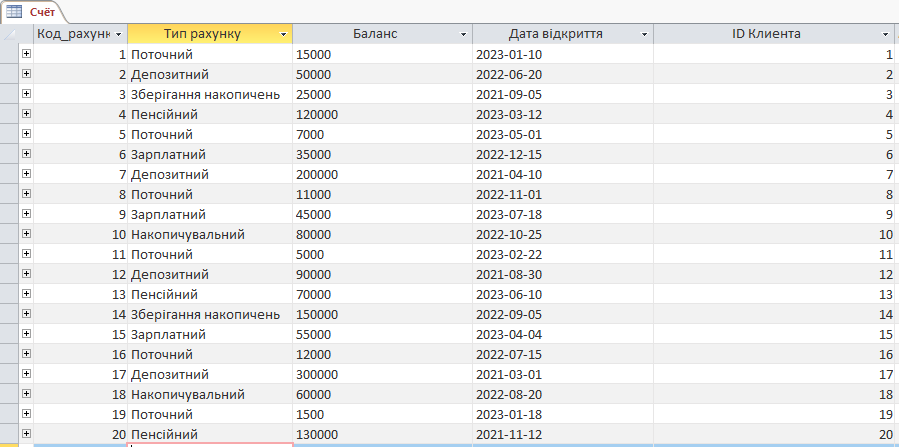
Таблиця 2.2 – Інформація таблиці «Кліент».



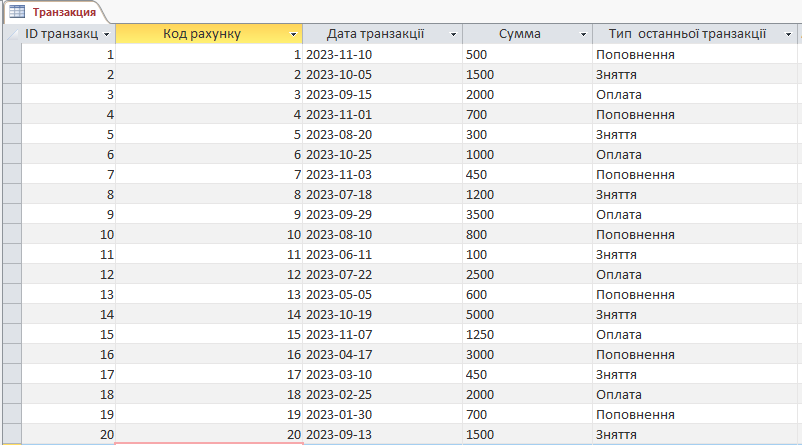
Таблиця 2.3 – Інформація таблиці «Кредит».



Таблиця 2.4 – Інформація таблиці «Рахунок».



Таблиця 2.5 – Інформація таблиці «Транзакція».



* 1. Проектування запитів

Однією з ключових функцій у створенні бази даних є можливість обробки інформації, включаючи операції вибору, створення, перегляду, редагування та видалення даних. Якщо таблиці в базі даних взаємопов'язані, то будь-які зміни в інформації в одній таблиці автоматично впливають на пов'язані таблиці. В MS Access існують інструменти для створення запитів з конкретними умовами. За їх допомогою можна вибирати необхідні поля, включаючи дані з кількох таблиць, та встановлювати умови для відбору записів, тобто налаштовувати фільтри для обробки інформації. Завдяки зв'язкам, при внесенні змін до однієї таблиці, дані в інших таблицях автоматично оновлюються.

Створення запитів дозволяє вибирати дані з однієї або кількох взаємопов'язаних таблиць. Як результат, створюється окрема таблиця з результатами, яку можна використовувати для подальшої обробки даних поряд з іншими таблицями. За допомогою запитів можна також переглядати, змінювати, видаляти або додавати дані в таблиці. Основною метою запитів є фільтрація та вибір необхідних даних відповідно до критеріїв пошуку, встановлених користувачем.

Створимо наступні запити:

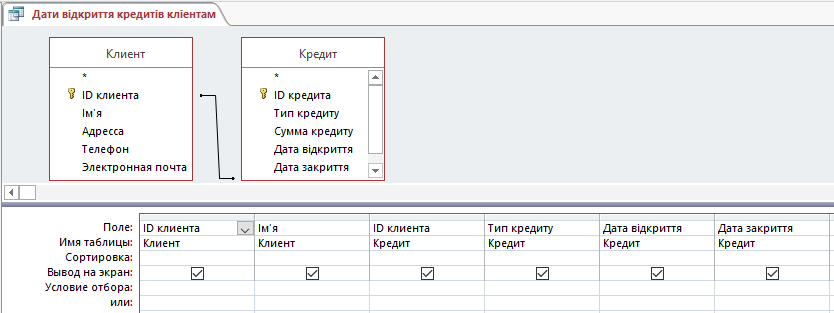


Рисунок 2.7. Структура запиту «Дати відкриття кредитів кліентам».

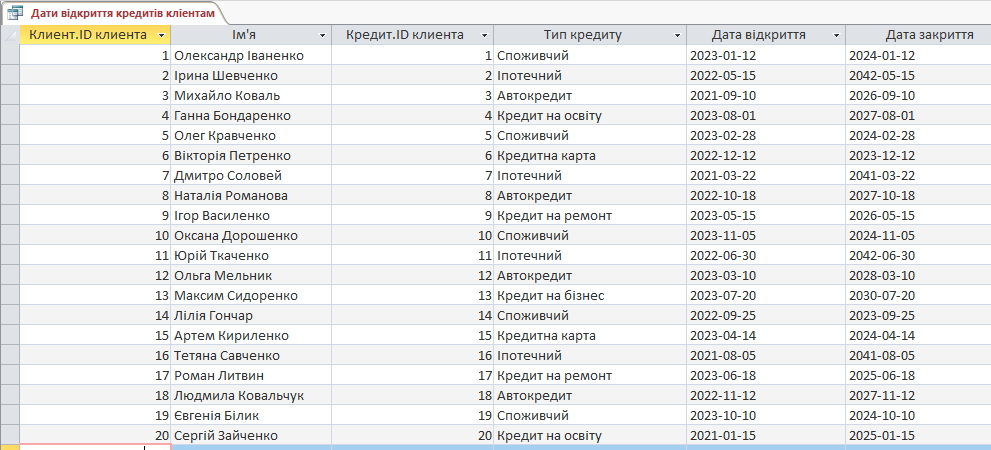


Рисунок 2.8 - Результат виконання запиту «Дати відкриття кредитів кліентам».

.

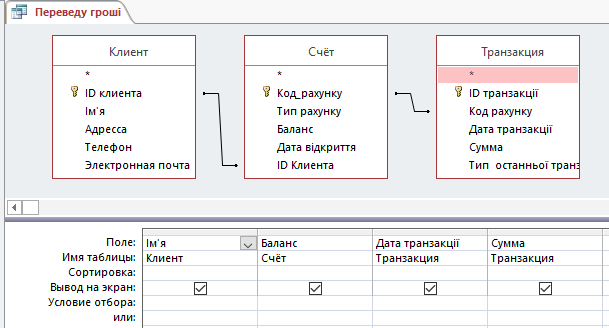


Рисунок 2.9- Структура запиту «Переведу гроші».

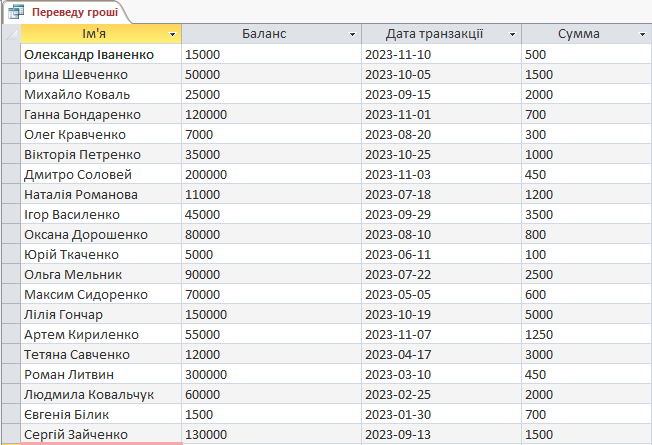


Рисунок 2.10 - Результат виконання запиту «Переведу гроші».

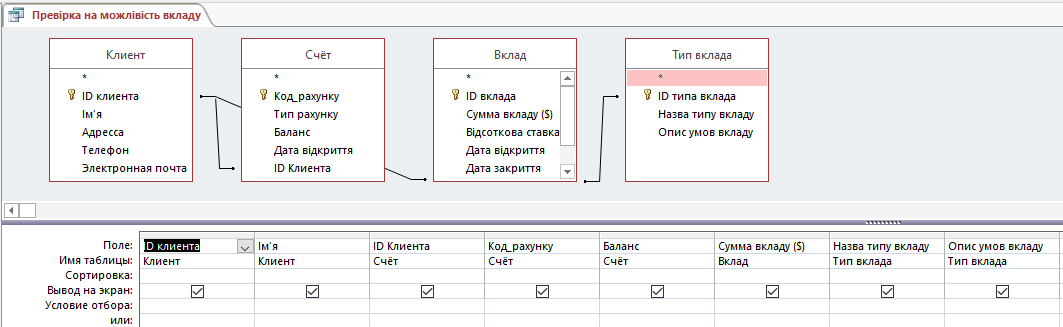


Рисунок 2.11- Структура запиту «Перевірка на можливість вкладу».

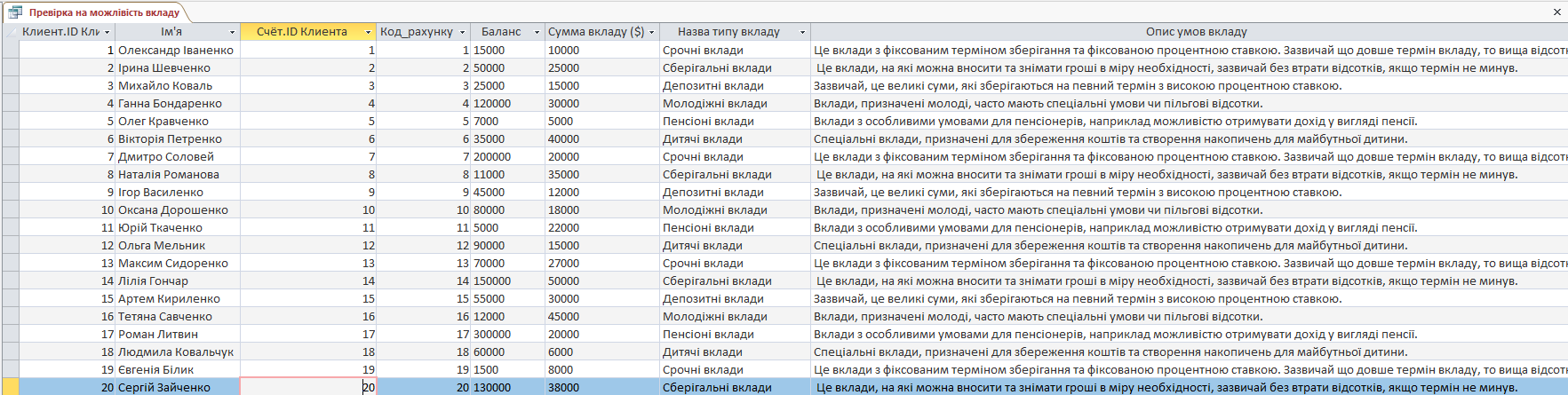


Рисунок 2.12 - Результат виконання запиту «Перевірка на можливість вкладу».

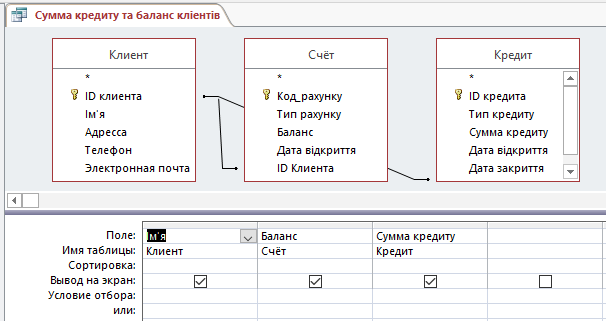


Рисунок 2.13- Структура запиту «Сумма кредиту та баланс кліентів».

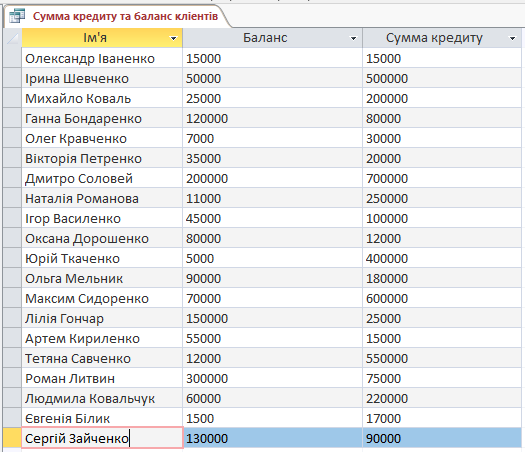


Рисунок 2.14 - Результат виконання запиту «Сумма кредиту та баланс кліентів».

* 1. Проектування форм та звітів

Форми використовуються для взаємодії з даними, які зберігаються в локальній базі даних Microsoft Access. Важливим аспектом є структура форми, особливо, якщо базу даних будуть використовувати кілька користувачів. Її слід належним чином спроектувати, щоб забезпечити ефективний та точний ввід даних. Для цього можна використовувати різні засоби оформлення, такі як кнопки та інші керуючі елементи.

Більшість користувачів баз даних використовують форми для перегляду та редагування даних у таблицях.

Розділена форма дозволяє одночасно переглядати дані у двох виглядах: у вікні форми та у вікні табличного вигляду даних. Робота з розділеними формами надає можливість використовувати переваги обох виглядів в одній формі.

Наприклад, ви можете використовувати табличний вигляд для швидкого пошуку записів, а потім перейти до вигляду форми для їх перегляду та редагування. Ці два вигляди пов'язані з одним джерелом даних та автоматично синхронізуються між собою.

Звіт - це зручний спосіб представлення інформації для користувачів, при цьому дані зазвичай представлені у таблицях. Звіти призначені для створення вихідних документів та їх подальшого друку. Звіти використовуються для форматування, сортування та зручного представлення інформації. Вони містять дані, які беруться з таблиць чи запитів, а також інформацію, яка додається в конструкторі звітів, таку як позначки, заголовки та графічні елементи. Також існує можливість автоматичного створення звітів, а процес створення звіту за допомогою Майстра подібний до процесу створення форми.

Форма яка виводить загальну інформацію про контактні данні кліента. Форма містить кнопки для переходу між записами та кнопку пошуку (Рис. 2.15).

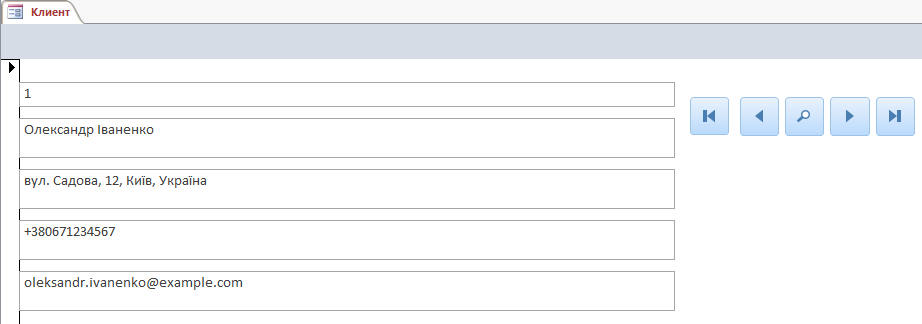


Рисунок 2.15 - Результат виконання запиту «Сумма кредиту та баланс кліентів».

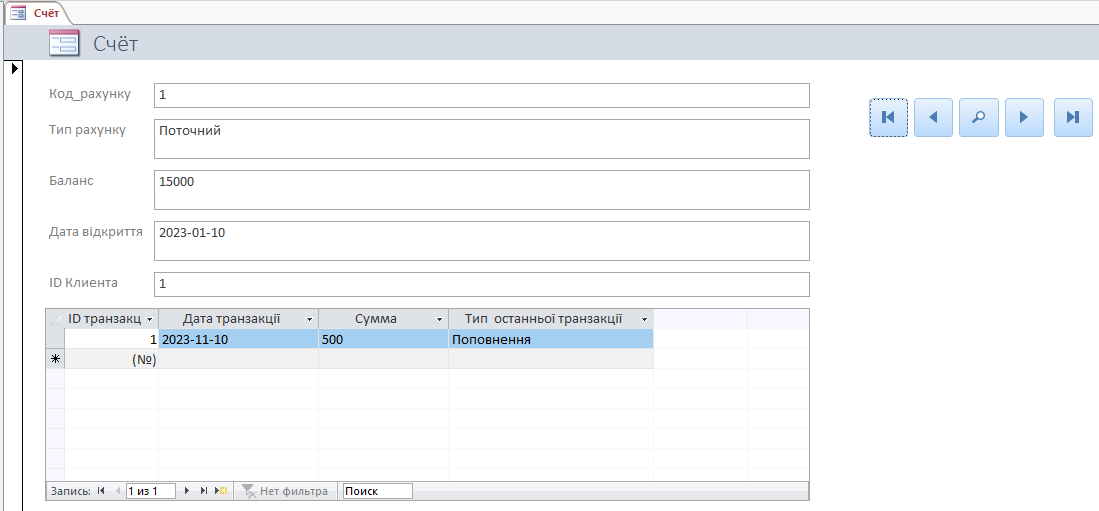


Рисунок 2.16 - Результат виконання запиту «Счёт».

* 1. Розробка інтерфейсу головної кнопкової форми

Кнопкова форма відіграє важливу роль в інтерфейсі бази даних, забезпечуючи зручний доступ до різних розділів даних за допомогою кнопок для швидкого переходу між сторінками. Така форма значно спрощує навігацію, особливо для користувачів, які не знайомі з усіма деталями бази даних. Головна кнопкова форма може виконувати функцію основного меню, звідки користувачі можуть отримати доступ до основних модулів або розділів бази даних.

Побудова кнопкової форми для нашої БД наведена нижче ([Рис. 2.17](#Рисунок_2_23)).

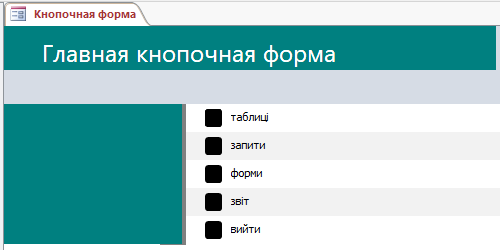


Рисунок 2.17 - Головна кнопкова форма

Завдяки створеній головній кнопковій формі, користувач отримує можливість швидкого доступу до потрібних таблиць, запитів, форм та звітів. На наступному рисунку представлено розгорнуте меню "Таблиці", яке дозволяє миттєво перейти до будь-якої необхідної таблиці, а після завершення роботи — повернутися назад до головної кнопкової форми для продовження навігації або виконання інших дій. ([Рис. 2.18](#Рисунок_2_24)).

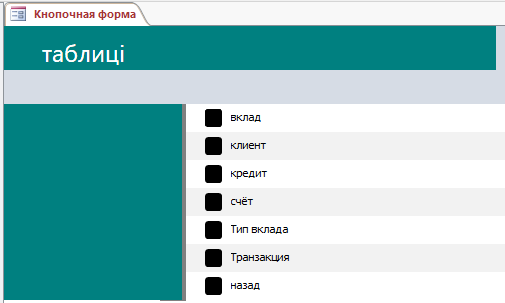


Рисунок 2.18 - Розгорнутий вигляд підменю «Таблиці»

Далі представлено розгорнутий вигляд підменю «Запити», де можна обрати необхідний для нас запит, або повернутись на головну кнопкову форму ([Рис. 2.19](#Рисунок_2_26)).

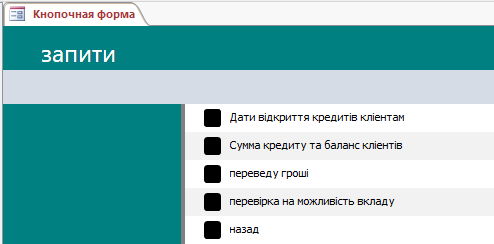


Рисунок 2.19 - Розгорнутий вигляд підменю «Запити»

На наступному рисунку представлено розгорнутий вигляд підменю «Форми», де можна обрати будь-яку необхідну форму або повернутись на головну кнопкову форму ([Рис. 2.20](#Рисунок_2_27)).

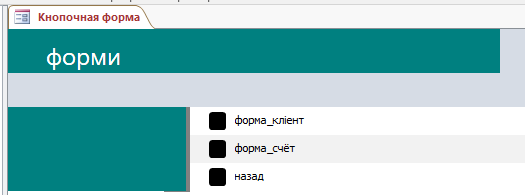


Рисунок 2.20 - Розгорнутий вигляд підменю «Форми»

Далі представлено розгорнутий вигляд підменю «Звіти», де користувач може обрати необхідний звіт, або повернутись на головну кнопкову форму ([Рис.2.21](#Рисунок_2_28)).

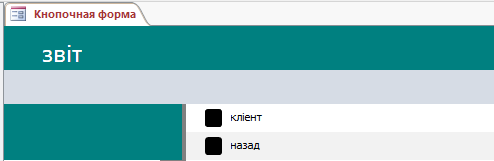


Рисунок 2.21 - Розгорнутий вигляд підменю «Звіти»

1. **РОБОТА В MYSQL**

MySQL — це одна з найпопулярніших систем управління реляційними базами даних, яка завдяки своїй простоті, продуктивності та надійності стала невід'ємною частиною багатьох сучасних додатків. Вона широко використовується як у невеликих проектах, так і в масштабних системах із мільйонами користувачів, таких як веб-сайти, інтернет-магазини, мобільні додатки та хмарні сервіси.

Однією з ключових переваг MySQL є її відкритий вихідний код, що дозволяє розробникам адаптувати систему під свої потреби. MySQL також підтримує інтеграцію з багатьма мовами програмування, такими як PHP, Python, Java, C++ та інші, що робить її універсальним рішенням для різних сфер застосування.

Додатково MySQL пропонує:

* Широкі можливості оптимізації продуктивності: наприклад, індексація таблиць, кешування запитів та розподіл навантаження за допомогою реплікації.
* Інструменти для резервного копіювання та відновлення: забезпечують надійність і безпеку даних у разі збоїв.
* Можливості масштабування: підходить як для роботи на одному сервері, так і в кластерних середовищах.

Особливо важливою є її активна спільнота користувачів і розробників, яка постійно розширює функціонал MySQL, додає нові інструменти та підтримує документацію.

В епоху цифрових технологій MySQL залишається універсальним інструментом для обробки великих обсягів даних, забезпечуючи гнучкість і ефективність роботи з базами даних. Сьогодні SQL є ключовим механізмом для зв’язку додаткового програмного забезпечення з базами даних. Він дозволяє виконувати широкий спектр запитів, які передбачають або отримання, або модифікацію даних із подальшим їх оновленням.Завдяки своїм можливостям і простоті у використанні, MySQL є оптимальним вибором для багатьох проектів — від персональних до корпоративних.

* 1. Побудова таблиць засобами MySQL

1. Створення бази даних Banking\_deposit\_system ([рис. 3.1](#Рисунок_3_1)).

Для створення таблиць використовувалася платформа MySQL Workbench. Щоб спростити роботу з даними та забезпечити зручність їх відображення, було обрано українське кодування тексту.

2. Створення таблиць та заповнення їх змістом

1) Таблиця «\_transaction або Транзакція» ([рис. 3.](#Рисунок_3_2)1-[3.2](#Рисунок_3_3)).

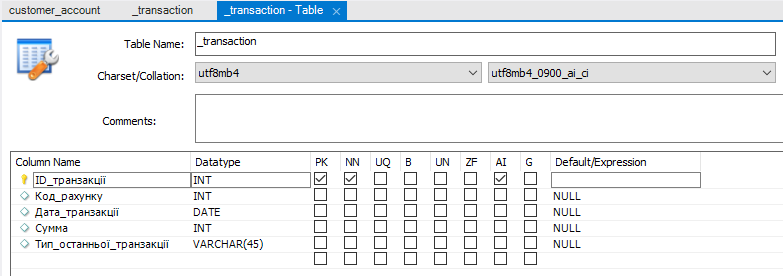


Рисунок 3.1 - Створення таблиці «\_transaction або Транзакція».

Ключовим полем у цій таблиці виступає поле «ID\_транзакції».

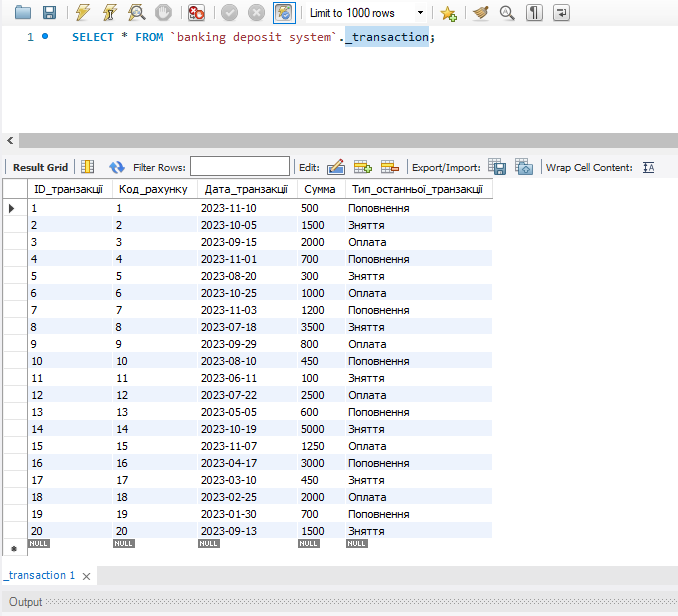


Рисунок 3.2 - Заповнення таблиці «\_transaction або Транзакція».

2) Таблиця «client або Кліент» ([рис 3.3](#Рисунок_3_5)-3.[4](#Рисунок_3_5)).

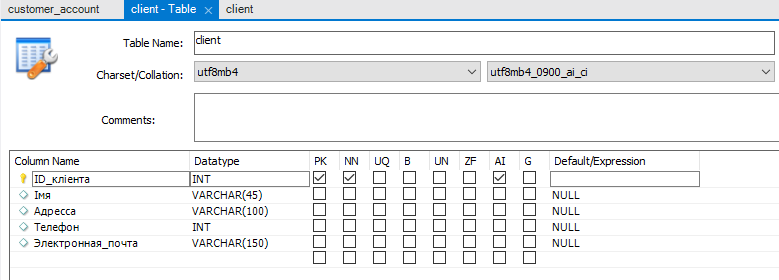


Рисунок 3.3 - Створення таблиці «client або Кліент».

Ключовим полем у цій таблиці виступає поле «ID\_клієнта».

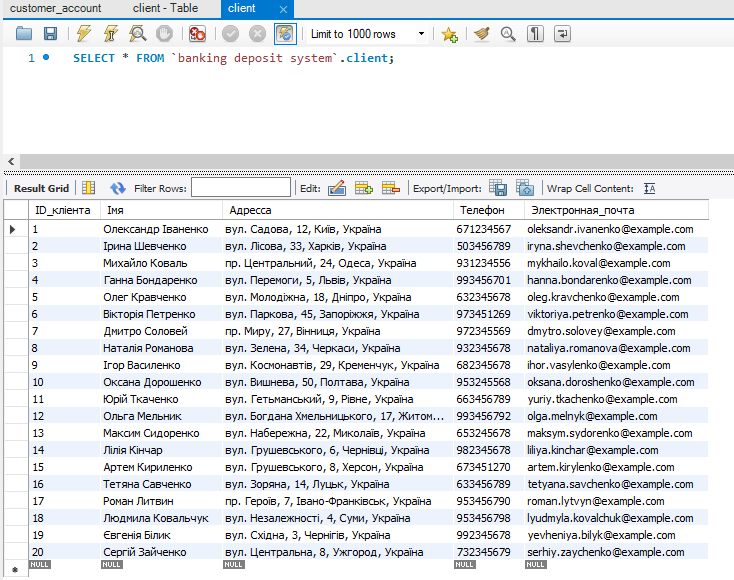


Рисунок 3.4 - Заповнення таблиці «client або Кліент».

3) Таблиця «contribution або Вклад» ([рис. 3.5](#Рисунок_3_6)-[3.6](#Рисунок_3_7)).

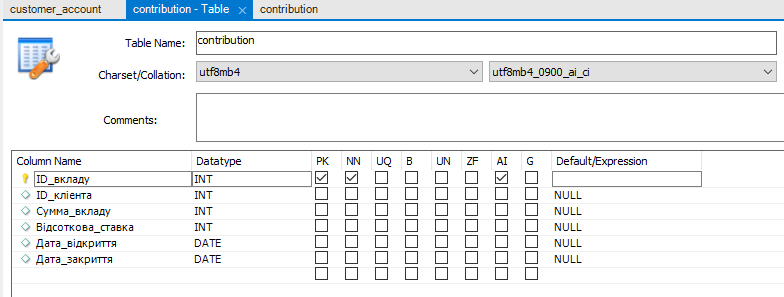


Рисунок 3.5 - Заповнення таблиці «contribution або Вклад».

Ключовим полем у цій таблиці виступає поле «ID\_вкладу». Поле «ID\_клієнта» бере значення із таблиці «Кліент».

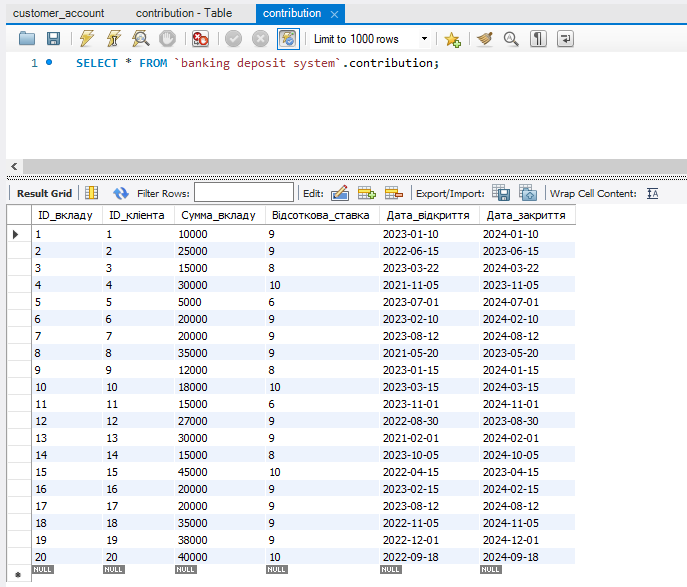


Рисунок 3.7 - Заповнення таблиці «contribution або Вклад».

4) Таблиця «credit або Кредит» ([рис. 3.7](#Рисунок_3_8)-[3.8](#Рисунок_3_9)).

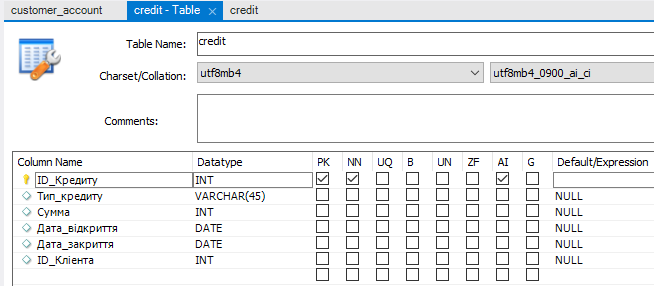


Рисунок 3.8 - Створення таблиці «credit або Кредит».

Ключовим полем у цій таблиці виступає поле «ID\_Кредиту». Поле «ID\_клієнта» бере значення із таблиці «Кліент».

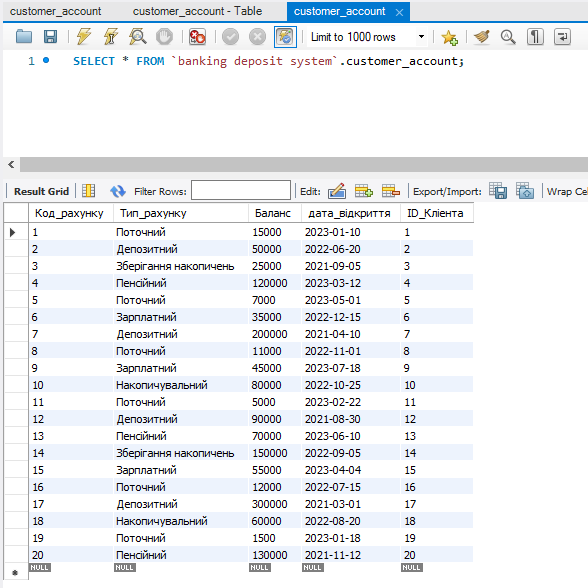


Рисунок 3.9 - Заповнення таблиці «customer\_account або Рахунок».

5) Таблиця «customer\_account або Рахунок» ([рис. 3.10](#Рисунок_3_10)-[3.11](#Рисунок_3_11)).

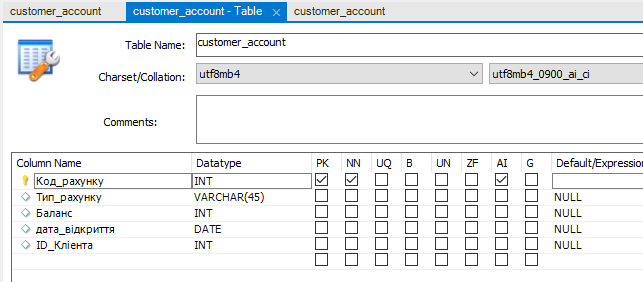


Рисунок 3.10 - Створення таблиці «customer\_account або Рахунок».

Ключовим полем у цій таблиці виступає поле «Код\_рахунку». Поле «ID\_клієнта» бере значення із таблиці «Кліент».

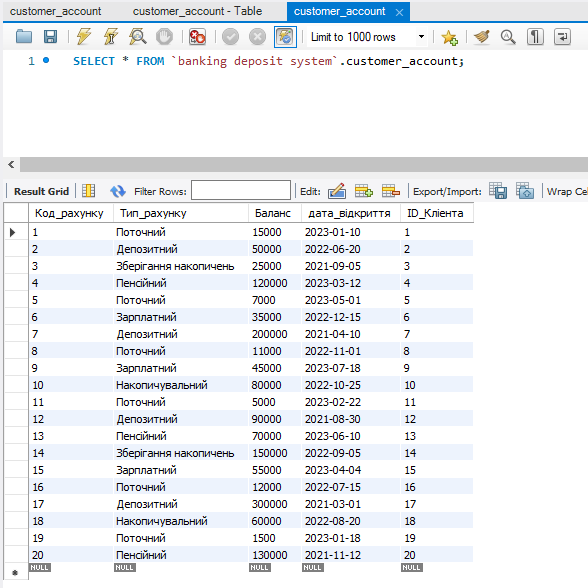


Рисунок 3.11 - Заповнення таблиці «customer\_account або Рахунок».

6) Таблиця «deposit\_type або Тип вкладу» ([рис. 3.11](#Рисунок_3_11)-[3.12](#Рисунок_3_12)).

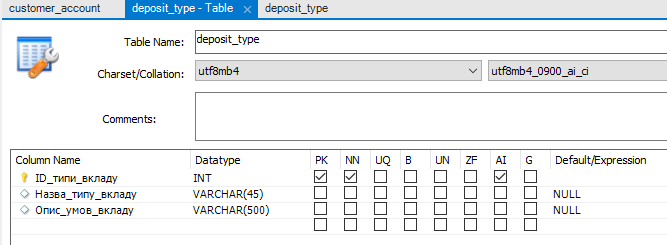


Рисунок 3.11 - Створення таблиці «deposit\_type або Тип вкладу».

Ключовим полем у цій таблиці виступає поле «ID\_типи\_вкладу».

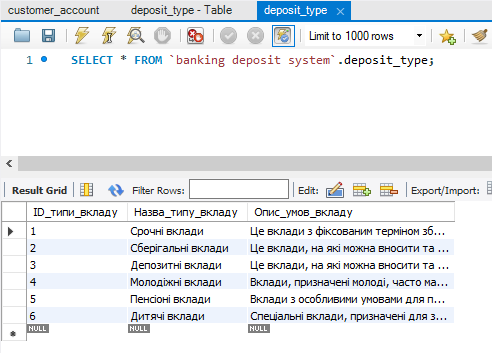


Рисунок 3.12 - Заповнення таблиці «deposit\_type або Тип вкладу».

На наступному рисунку зображено EER-діаграму «сутність-зв’язок» для бази даних «bamking deposit system» ([рис 3.13](#Рисунок_3_17)).

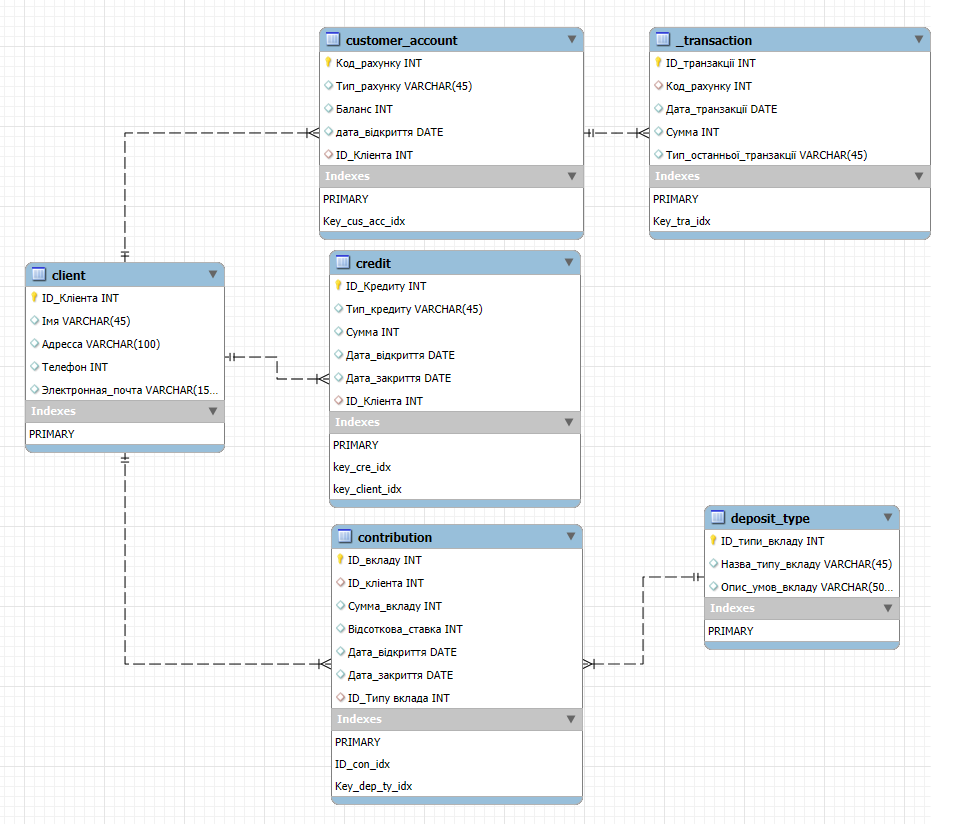


Рисунок 3.13 - EER-діаграма створеної бази даних

На цьому рисунку зображений список створених таблиць бази даних «bamking deposit system» ([рис.3.14](#Рисунок_3_18)).

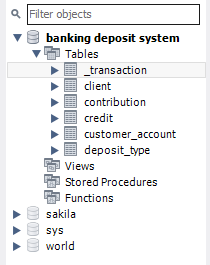


Рисунок 3.14 – Список створених таблиць бази даних «banking deposit system»

* 1. Використання запитів на мові SQL

Всі запити в SQL складаються з одиночної команди, яка має назву «SELECT». Структура цієї команди достатньо проста, але ви повинні розширювати її так, щоб виконати складні запити і обробку даних.

1. WHERE – це запит на вибірку інформації з таблиці.

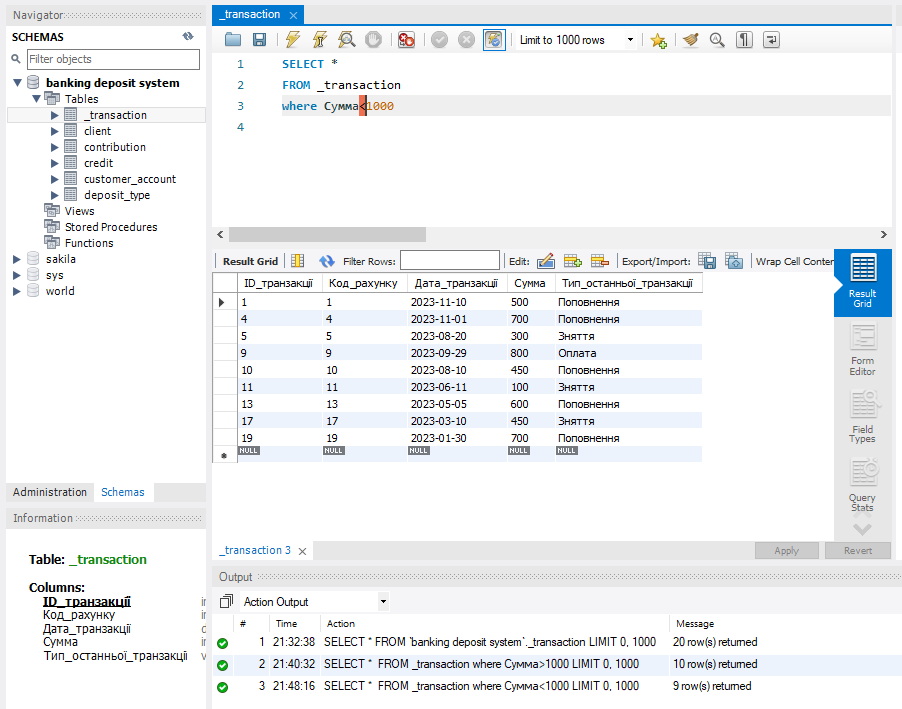


Рисунок 3.15 - Результат запиту для знаходження певної сумми.

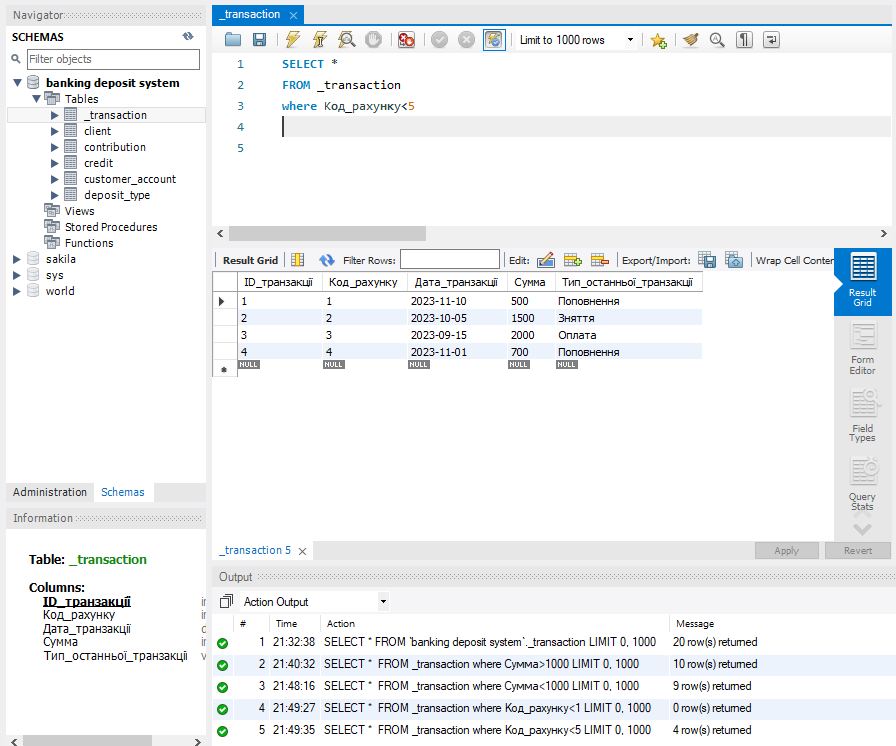


Рисунок 3.16 - Результат запиту для знаходження кліентів з Кодом рахунку до 5

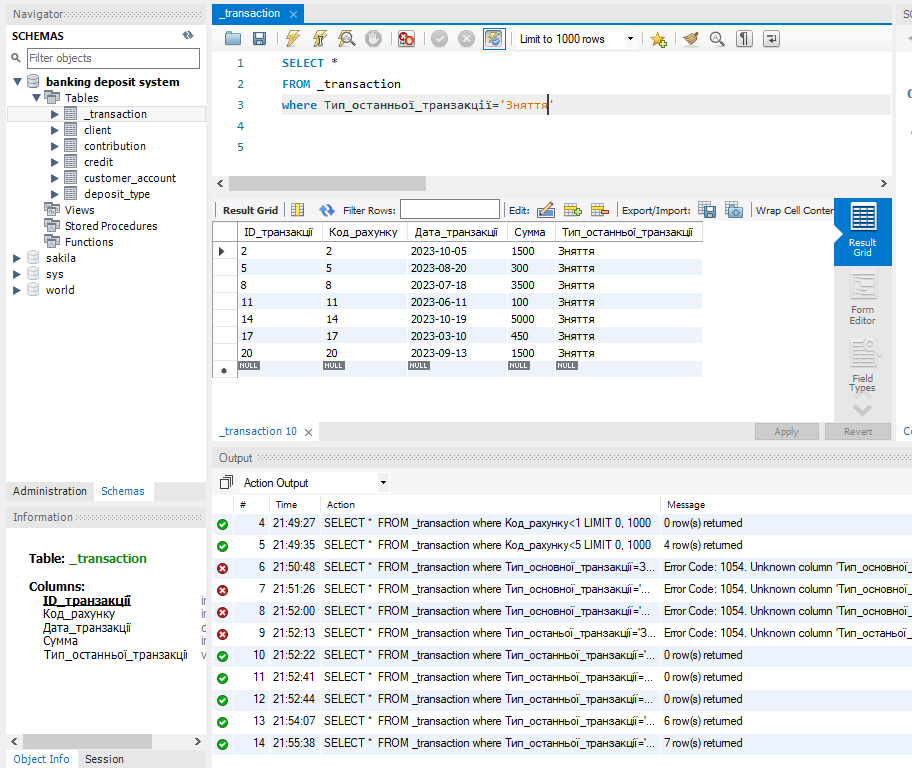


Рисунок 3.17 - Результат запиту по пошуку з таблиці Тип основної транзакції за запитом Зняття

1. Запит який за допомогою «JOIN» виводить ім'я, баланс і сумма; з таблиць «Клієнт», «Рахунок» і «Кредит». ([рис. 3.18](#Рисунок_3_23)).

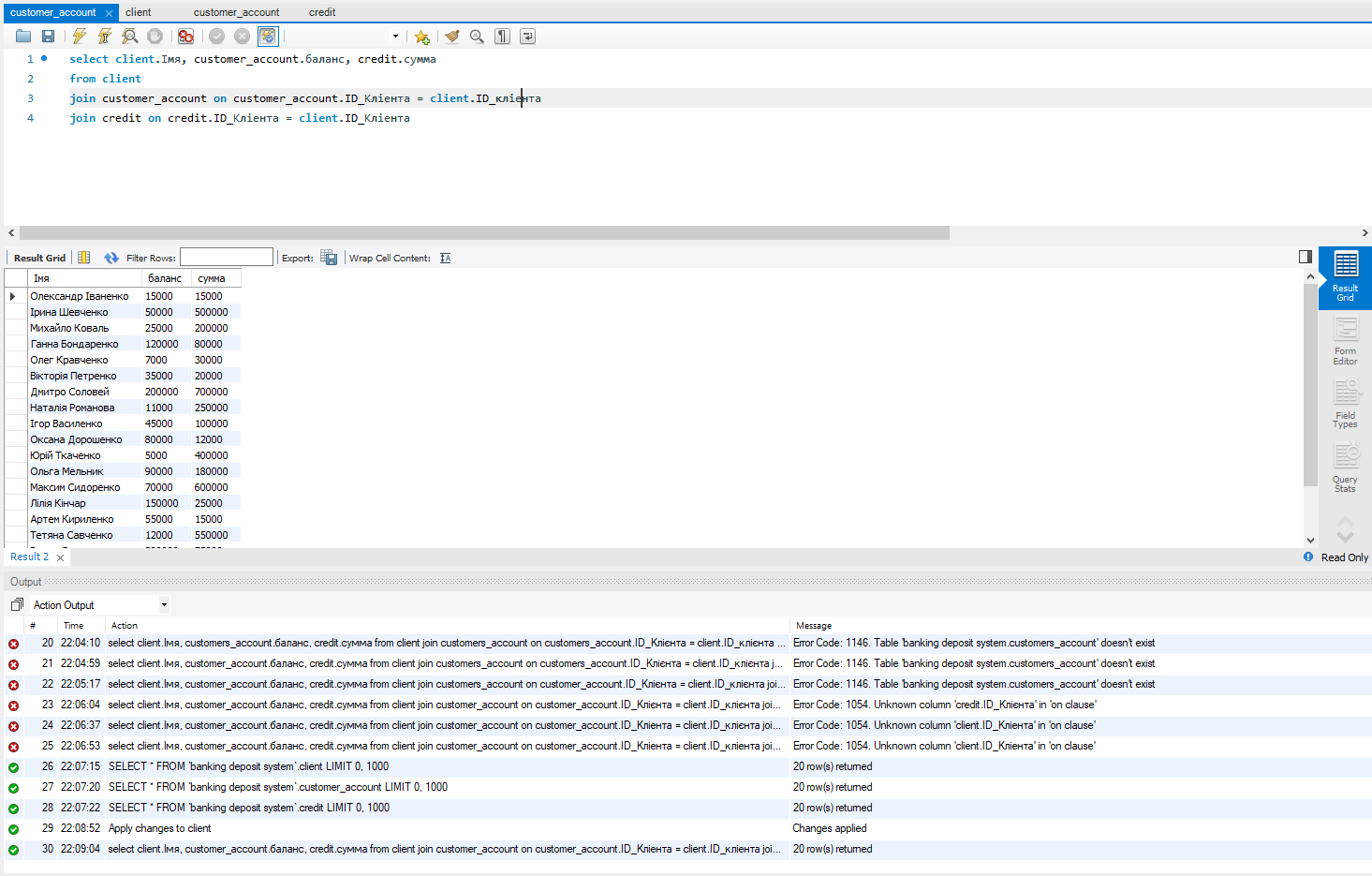


Рисунок 3.18 - Результат запиту по пошуку з таблиці Тип основної транзакції за запитом Оплата

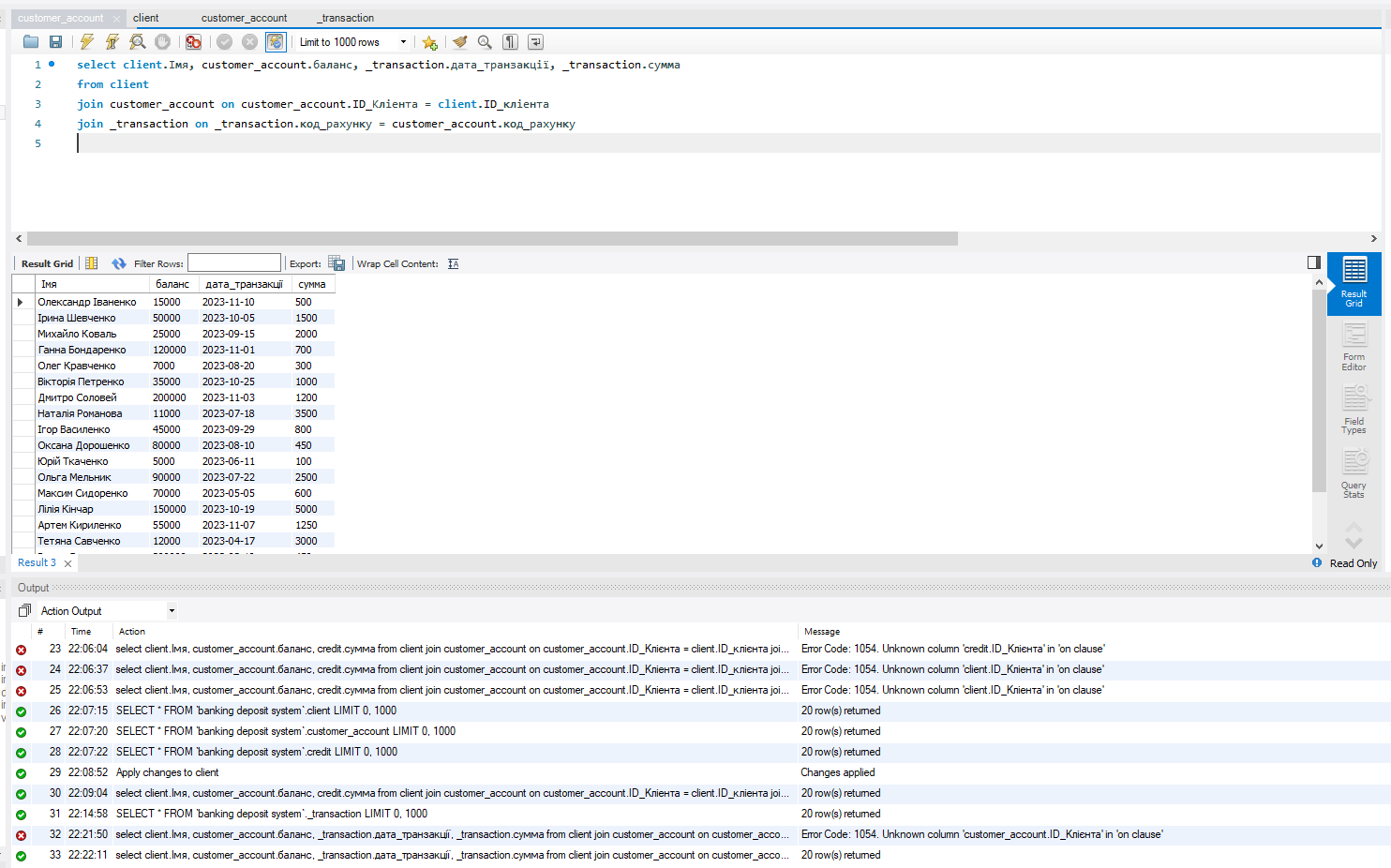


Рисунок 3.19 - Результат запиту по пошуку з таблиці Тип основної транзакції за запитом Оплата

# **ВИСНОВОК**

Ефективне зберігання та систематизація важливої інформації стали ключовими аспектами для швидкої та якісної обробки даних у сучасному світі. Практично кожна компанія в наш час потребує ведення статистики, створення або редагування списків з обширною кількістю даних та їх взаємозв'язків. І саме для цих потреб були розроблені бази даних - сукупності взаємопов'язаних даних, які зберігаються разом з мінімальною надмірністю для їх оптимального використання.

Створення баз даних, їх підтримка та надання доступу користувачам реалізовані за допомогою систем управління базами даних - спеціального програмного інструментарію.

У рамках цієї курсової роботи була розглянута та реалізована база даних для підприємства з а банківськими вкладами. Ця база допомагає відстежувати поточні рахунки, а також надає доступ до необхідної інформації щодо вкладів, клієнтів, транзакцій.

У результаті було створено шість таблиць, які містять інформацію про Клієнтів, Вклади, Рахунки, Кредити і Транзакції. Десять запитів (4 Accses а також 6 у MySQLw) полегшують пошук та відображення інформації у зручному для користувача форматі. Також розроблено дві форми та головну кнопкову форму, які спрощують управління базою даних.

Розроблений додаток відповідає всім вимогам предметної області, таблиці бази даних відповідають вимогам нормалізації, що забезпечує цілісність, не вкладеність і несуперечність інформації. Під час виконання курсового проекту отримано нові навички зі створення та наповнення таблиць, складання та оформлення форм, створення запитів та звітів, а також детальне ознайомлення з СУБД MySQL: робота з базою даних, таблицями, редагуванням та виконанням запитів.

# **Список використаної літератури**

* 1. Гладкий О.І., Павлов В.О. Основи баз даних і SQL. Навчальний посібник. – Львівська політехніка, 2017.
  2. Головко І.М., Кравченко Ю.А. MySQL для початківців: Практичний посібник. – Академія, 2021.
  3. Зелений В.В. Microsoft Access: Практичний курс. – Освіта України, 2019.
  4. Бемер С., Фратер Г. MS Access. — Київ: BHV, 2010 — 384с
  5. Олександр Ш., Леся Ш. Розробка баз даних в СУБД Microsoft Access -Кондор, 2018.
  6. Гурвиц, Г. Microsoft Access 2010. Розробка додатків на реальному прикладі/ Г. Гурвиц. — М.: БХВ-Петербург, 2017. — 496 c.
  7. Томас К., Каролін Б. Проектування, реалізація і супровід. Теорія і практика.– Вільямс, 2017.