

Міністерство освіти і науки України  
Український державний університет науки і технологій

Комп'ютерних технологій і систем  
(назва факультету)

Електронних обчислювальних машин  
(повна назва кафедри)

Пояснювальна записка  
до кваліфікаційної роботи  
перший (бакалаврський)  
(ступінь вищої освіти)

*Do Jaxa 5*  
*12.06.2022*

на тему: Підвищення ефективності роботи захищеної бази даних наукових розробок кафедри електронних обчислювальних машин за освітньою програмою Кібербезпека зі спеціальності: 125 Кібербезпека  
(шифр і назва спеціальності)

Виконав: студент групи КБ 19120

*[Signature]*

(підпис студента)

/ Анна Хрестян /

(Ім'я ПРІЗВИЩЕ)

Керівник:

*[Signature]*

(підпис)

/ доцент, Вікторія Пахомова /

(посада, Ім'я ПРІЗВИЩЕ)

Нормоконтролер:

*[Signature]*

(підпис)

/ старший викл., Володимир Дзюба /

(посада, Ім'я ПРІЗВИЩЕ)

Консультанти:

(назва розділу)

(підпис)

(посада, Ім'я ПРІЗВИЩЕ)

(назва розділу)

(підпис)

(посада, Ім'я ПРІЗВИЩЕ)

(назва розділу)

(підпис)

(посада, Ім'я ПРІЗВИЩЕ)

(назва розділу)

(підпис)

(посада, Ім'я ПРІЗВИЩЕ)

Засвідчую, що у цій роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент

*[Signature]*

(підпис)

Дніпро – 2022 рік

**Ministry of Education and Science of Ukraine**  
**Ukrainian State University of Science and Technologies**

Computer technology and systems  
(faculty)

Electronic computers  
(department)

**Explanatory Note**  
**to Bachelor's Thesis**  
**first (bachelor's)**  
(higher education degree)

on the topic: Increase efficiency of the protected data base of the science  
development of the electronic computing machines

according to educational curriculum Cybersecurity

in the Speciality: 125 Cybersecurity

(speciality and its code)

Done by the student of the group: CS 19120 / Anna Hrestyan /  
(name, surname)

Scientific Supervisor:  / docent, Viktoria Pakhomova /  
(position, name, surname)

Normative controller :  / senior lecturer, Vladimir Dziuba /  
(position, name, surname)

Supervisors

(Chapter title heading) / /  
(position, name, surname)

(Chapter title heading) / /  
(position, name, surname)

(Chapter title heading) / /  
(position, name, surname)

(Chapter title heading) / /  
(position, name, surname)

Dnipro – 2022

Міністерство освіти і науки України  
Український державний університет науки і технологій

Факультет: Комп'ютерних технологій і систем  
Кафедра: Електронних обчислювальних машин  
Рівень вищої освіти: Перший (бакалаврський)  
Освітня програма: Кібербезпека  
Спеціальність: 125 Кібербезпека

(шифр та назва)

ЗАТВЕРДЖУЮ  
Завідувач кафедри ЕОМ  
Ігор Жуковицький  
(підпис) (Ім'я ПРІЗВИЩЕ)  
Дата \_\_\_\_\_

**ЗАВДАННЯ**

на кваліфікаційну роботу перший (бакалаврський)  
(ступінь вищої освіти)

студенту Хрестян Анні Вячеславівні  
(Прізвище, Ім'я По батькові)

1. Тема роботи: Підвищення ефективності роботи захищеної бази даних наукових розробок кафедри електронних обчислювальних машин

Керівник роботи: Пахомова Вікторія Миколаївна, к.т.н., доцент кафедри ЕОМ  
(Прізвище, Ім'я, По батькові, науковий ступінь, вчене звання)

затверджені наказом від "07" 12 2021 р. № 67ст

2. Строк подання студентом 13.06.2022 р.  
роботи:

3. Вихідні дані до роботи: дані про наукові розробки, що розташовані в репозитарію університету; дані про професорсько-викладацький склад кафедри ЕОМ.

4. Зміст пояснювальної записки (перелік питань, які потрібно опрацювати):

4.1 Аналітична частина: огляд і аналіз предмета проектування.

4.2 Основна частина: постановка задачі; проектування БД за методом «Сутність-Зв'язок»; створення та обробка БД в Microsoft Access з використанням SQL;  
організація захисту БД(парольний захист, відображення та приховування об'єктів у вікні БД, шифрування та захист на рівні користувачів); підвищення ефективності створеної БД.


6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Завдання видав: (підпис консультанта, дата)	Завдання прийняв: (підпис студента, дата)

**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вступ	03.06.2022-05.06.2022	5%
2	Огляд і аналіз предмета проектування	25.04.2022-16.05.2022	20%
3	Постановка задачі	25.04.2022-05.05.2022	10%
4	Проектування БД за методом «Сутність-Зв'язок»	03.05.2022-12.05.2022	20%
5	Створення і обробка БД в Microsoft Access з використанням SQL	12.05.2022-26.05.2022	20%
6	Організація захисту БД та підвищення її ефективності	15.05.2022-30.05.2022	20%
7	Висновки та рекомендації	03.06.2022-05.06.2022	5%
8	Подання кваліфікаційної роботи до кафедри	13.06.2022	
9	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	23.06.2022	

Студент

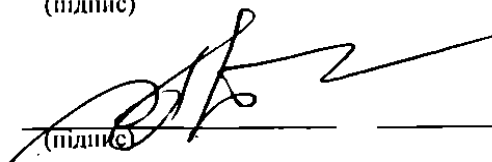


(підпис)

Анна Хрестян

(Ім'я ПРІЗВИЩЕ)

Керівник роботи



(підпис)

Вікторія Пахомова

(Ім'я ПРІЗВИЩЕ)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи бакалавра: 47 с., 46 рис., 1 табл., 4 додатки, 18 джерел.

Об'єкт розробки – база даних наукових розробок кафедри електронних обчислювальних машин (ЕОМ).

Мета кваліфікаційної роботи – підвищення ефективності роботи захищеної бази даних наукових розробок кафедри ЕОМ.

Методи дослідження – проектування бази даних за методами: «Сутність-Зв'язок» та нормальних форм; створення бази даних в Microsoft Access з використанням SQL; організація захисту бази даних; використання механізмів підвищення ефективності роботи бази даних.

Створено базу даних наукових розробок кафедри ЕОМ в Microsoft Access з використанням SQL. Виконано організацію захисту бази даних наступними засобами: парольний захист; відображення та приховування об'єктів у вікні бази даних; шифрування й захист на рівні користувачів; проведено підвищення ефективності роботи бази даних на основі наступних підходів: оптимізація самої бази даних та оптимізація операторів SQL.

Рекомендовано створену та захищену базу даних наукових розробок до використання адміністрації кафедри ЕОМ.

Запропонована нова лабораторна робота на основі розгляду механізмів підвищення ефективності бази даних здобувачам першого (бакалаврського ступеня) спеціальностей «Кібербезпека» та «Комп'ютерна інженерія» з дисципліни «Бази даних».

Ключові слова: БАЗА ДАНИХ, НАУКОВІ РОЗРОБКИ, СУТНІСТЬ-ЗВ'ЯЗОК, НОРМАЛЬНІ ФОРМИ, MICROSOFT ACCESS, SQL, ЗАХИСТ, ЕФЕКТИВНІСТЬ.

## ЗМІСТ

ВСТУП.....	43
1 ОГЛЯД І АНАЛІЗ ПРЕДМЕТА ПРОЕКТУВАННЯ.....	45
1.1 Методи проектування реляційної бази даних.....	45
1.2 Засоби програмної реалізації.....	50
1.3 Основні висновки.....	53
2 ПРОЕКТУВАННЯ, СТВОРЕННЯ ТА ОБРОБКИ ЗАХИЩЕНОЇ БАЗИ ДАНИХ.....	54
2.1 Постановка задачі.....	54
2.2 Проектування реляційної бази даних за методом «Сутність-Зв'язок».....	54
2.3 Створення та обробка бази даних в Microsoft Access з використанням мови SQL.....	59
2.4 Організація захисту бази даних.....	65
2.4.1 Парольний захист.....	67
2.4.2 Відображення та приховування об'єктів у вікні бази даних.....	69
2.4.3 Шифрування та захист на рівні користувачів.....	71
2.5 Підвищення ефективності створеної бази даних та її використання.....	74
2.6 Основні висновки.....	79
ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ.....	80
ПЕРЕЛІК ПОСИЛАНЬ.....	81
ДОДАТОК А ПОЧАТКОВЕ ВІДНОШЕННЯ (ФРАГМЕНТ). АКТ НА ВПРОВАДЖЕННЯ.....	<b>Помилка! Закладку не визначено.</b>
ДОДАТОК Б ПРОЕКТУВАННЯ МЕТОДОМ НОРМАЛЬНИХ ФОРМ.....	50
ДОДАТОК В ЗАПРОПОНОВАНА НОВА ЛАБОРАТОРНА РОБОТА.....	56
ДОДАТОК Г ТЕЗИ ДО НАУКОВОЇ КОНФЕРЕЦІЇ.....	61

## ВСТУП

На сьогоднішній день бази даних відіграють важливу роль у розвитку технологій, вони здатні зберігати великі об'єми інформації, тому вони повинні бути надійно захищені та мати високі показники ефективності в роботі, що підтверджує актуальність теми.

Мета кваліфікаційної роботи - підвищення ефективності роботи захищеної бази даних наукових розробок кафедри ЕОМ. Відповідно до мети поставлені наступні задачі:

1. Виконати огляд і аналіз предмета проектування.
2. Провести проектування бази даних наукових розробок кафедри ЕОМ за методом «Сутність – Зв'язок».
3. Створити в Microsoft Access засобами SQL базу даних та провести її захист.
4. Розглянути можливі підходи щодо підвищення ефективності створеної бази даних.

Існує багато моделей представлення бази даних, але саме реляційна база даних є найпоширенішим типом баз даних.

Вона може зв'язувати окремі записи з декількома таблицями, що дозволяє встановлювати між даними відношення типу «багато-до-багатьох» та «один-до-багатьох».

Завдяки гнучкості та проектному рішенню, спрямованому на збереження цілісності даних, реляційна база даних є основним способом управління даними та їх зберігання.

Реляційна база даних повинна задовільняти наступним умовам: кожен елемент таблиці – один елемент даних, дані у стовпцях повинні бути однорідними, кожен стовпець має унікальне ім'я, однакові рядки в таблиці відсутні, порядок рядків може бути довільним.

Для програмної реалізації бази даних, можна відносити такі додатки: Oracle, Paradox, dBase, Clipper, MS Access, Sybase, Informix, та ін.

Якщо в системі захисту бази даних є недоліки, то даним може бути нанесена шкода, яка може бути виражена у: порушенні цілісності даних, втраті важливої інформації, попаданні даних до зловмисника, та ін.

Продуктивність роботи бази даних напряму залежить від розміру проекту та складності зв'язку між таблицями.

Дана кваліфікаційна робота складається з вступу, двох розділів та висновків.

У першому розділі виконано: огляд предмета проектування, методи проектування реляційної бази даних, огляд її програмної реалізації.

Проведено порівняльний огляд додатків для програмного створення бази даних, таких як: Microsoft Access, MySQL Workbench, HeidiSQL, Navicat.

Для програмної реалізації обрано Microsoft Access.

У другому розділі сформована постановка задачі, виконано проектування за методом «Сутність – Зв'язок», створено базу даних в Microsoft Access з використанням SQL, виконано її захист наступними засобами: парольний захист, відображення та приховування об'єктів у вікні бази даних, шифрування й захист на рівні користувачів, а також розглянуті підходи щодо підвищення її ефективності: оптимізація самої бази даних та оптимізація операторів SQL.

Результати кваліфікаційної роботи подавались у вигляді тез доповіді до: Всеукраїнської науково технічної конференції студентів і молодих учених, що відбулась в Українському державному університеті науки та технологій (УДУНТ) у 2022 році.

## 1 ОГЛЯД І АНАЛІЗ ПРЕДМЕТА ПРОЕКТУВАННЯ

### 1.1 Методи проектування реляційної бази даних

Базою даних називають організовану структуру, яка призначена для зберігання, редагування та обробки взаємозалежної інформації, зазвичай великих обсягів.

Існують три класичні моделі бази даних: ієрархічна, мережева та реляційна.

Також з'явилися нові моделі, які стали активно впроваджуватися та розвиватися – постреляційна, багатомірна та об'єктно-орієнтована.

Ієрархічна модель являє собою таку базу даних в якій усі елементи запису впорядковані, тобто один елемент вважається основним, а інші підлеглими[1].

Модель має на увазі, що відношення «батько-нащадок» - це відношення «один-до-багатьох». Це означає, що у дочірнього елемента не може бути більше одного предка.

Дана моделі бере початок з 1950-го року вона стала першою системою керування базами даних для мейнфреймів. Прикладом є IMS – Information Management System фірми IBM (International Business Machines), яка характеризувалась наступними ознаками: записом є набір значень полів, записи одного типу групуються на типи записів, відношення «батько-нащадок» - це відношення виду «один-до-багатьох» між двома типами записів, ієрархічна база даних складається з декількох ієрархічних схем.

Основні переваги моделі: швидкість виконання операцій над даними та ефективно використання пам'яті комп'ютера.

Незважаючи на переваги має два суттєвих недоліки, а саме: велика громіздкість та складні логічні зв'язки, складність розуміння для звичайного користувача.

На практиці часто зустрічається зв'язок «багато-до-багатьох», який неможливо відобразити в даній моделі без дублювання даних, тому популярність цієї моделі невелика вона вважається незручною та застарілою[1,2].

Прикладом реалізації моделі є ієрархія вкладених папок та файлів операційної системи Windows[2].

Мережева модель представляє собою таку базу даних, де у батьківського елемента може бути кілька нащадків, а у дочірнього елемента – кілька предків.

Модель має ґратчасту структуру та записи в ній пов'язані списками з покажчиками.

Модель була розроблена Чарльзом Бахманом у 1960 році компанією General Electric та мала назву IDS – Integrate Data Store.

Основний елемент моделі даних – набір, який складається з типу «запис-власник», запис підпорядкованого рівня («запис-член»), може виконувати свою роль у кількох наборах в той же час запис старшого рівня («запис-власник») також може бути членом або власником в інших наборах. Зазвичай у наборі існує відношення «один-до-багатьох», але дозволено також відношення «один-до-одного».

Популярність мережевої моделі збіглася з популярністю ієрархічної моделі.

Певні дані набагато зручніше моделювати з декількома предками для одного дочірнього елемента.

Представники даної моделі: TurboIMAGE, IDMS, вбудована RDM, серверна RDM.

Можна виділити наступні переваги моделі: можливість ефективної реалізації за показниками витрат пам'яті та оперативності, а також більше можливостей утворення довільних зв'язків.

Також має певні недоліки, а саме: складність схеми бази даних побудованої на її основі та важкість розуміння для звичайного користувача.

Хоча модель дозволяла природніше моделювати відношення між елементами, але широкого використання на практиці так і не отримала, тому що компанія IBM вирішила не відмовлятися від ієрархічної моделі у розширеннях для своїх продуктів – IMS та DL/I, а також через деякий час її замінила реляційна модель бази даних, яка мала більш високорівневий та декларативний інтерфейс[1].

Прикладом реалізації мережевої модель бази даних є організація CODASYL, яка ґрунтується на математичній теорії множин[3].

У реляційній моделі бази даних інформація організована у вигляді прямокутних таблиць. Таблиці складаються з рядків (кортежей) та стовпців (атрибутів)[1]. Графічне зображення реляційної моделі представлено на рисунку 1.1[1].

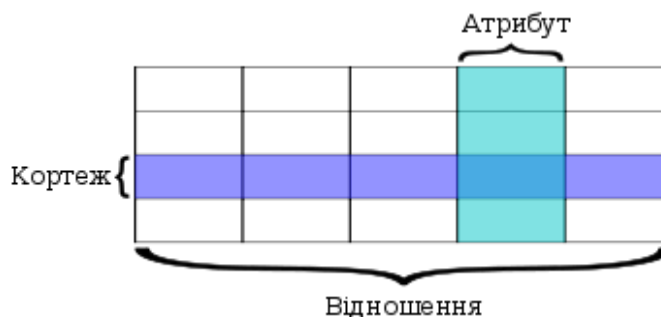


Рисунок 1.1 – Графічне представлення реляційної моделі

Вперше основу моделі було викладено у статті Єдгара Кодда у 1970 році, але найбільш поширене визначення належить Крістоферу Дейту, згідно якого реляційна модель повинна складатися з трьох основних частин – структурна, цілісна та маніпуляційна.

У структурній частині описуються об'єкти, які розглядаються моделлю.

У цілісній частині описуються обмеження спеціального виду, які повинні виконуватися для відносин у будь-яких реляційних базах даних.

У маніпуляційній частині описується два еквівалентних способи маніпулювання реляційними даними – реляційна алгебра та реляційне числення.

Реляційні таблиці повинні включати наступні властивості: усі значення повинні бути атомарними, кожен рядок є унікальним, порядок стовпців та рядків не важливий, кожному стовпцю належить своє унікальне ім'я.

Реляційна система керування базами даних (РСКБД) дозволяє визначити структурні аспекти даних, обробки відношень та їх цілісності.

Представники даної моделі: Oracle, Paradox, dBase, Clipper, MS Access, Sybase, Informix.

Модель має ряд переваг, а саме: вона проста та доступна для користувача, має єдину інформаційну конструкцію – таблицю, побудована на суворих правилах, які базуються на математичному апараті.

Також має два недоліки: при логічному проектуванні може утворюватися безліч таблиць, що призводить до складності розуміння структури даних, а також займає багато зовнішньої пам'яті.

Реляційна модель більш гнучка, ніж ієрархічна та простіша в управлінні, ніж мережева, тому вона не втрачає актуальність на сьогодні і використовується найчастіше[4].

Прикладом реляційної моделі бази даних є СУБД «Віра» в якій використовується мова програмування ПЛ/1, реалізована за допомогою макрозасобів цієї мови[5].

Існує два метода проектування бази даних:

1. Метод нормальних форм.
2. Метод «Сутність-Зв'язок».

Сенс першого методу полягає в переведенні відношень з першої нормальної форми в нормальні форми більш високого рівня, згідно визначених правил.

Приведення до кожної наступної нормальної форми обмежує визначений тип функціональних залежностей, ліквідує аномалії при виконанні операцій над відношенням бази даних, а також зберігає властивості попередніх нормальних форм.

Ціль першого етапу проектування – перехід відношення із першої нормальної форми до другої нормальної форми, ліквідування часткової функціональної залежності.

Якщо кожний атрибут відношення атомарний, то відношення знаходиться в першій нормальній формі.

Якщо відношення вже знаходиться в першій нормальній формі та кожен неключовий атрибут відношення функціонально повно залежить від первинного ключа, то відношення знаходиться в другій нормальній формі.

Ціль другого етапу проектування – перехід відношення із другої нормальної форми до третьої нормальної форми та ліквідування транзитивної функціональної залежності.

Якщо відношення вже знаходиться в другій нормальній формі та кожен його неключовий атрибут нетранзитивно залежить від первинного ключа, то відношення знаходиться в третій нормальній формі.

Перехід відношення в наступну нормальну форму здійснюється методом «декомпозиції без утрат».

Процес проектування можна вважати завершеним, якщо результуючі відношення знаходяться у БКНФ.

Якщо відношення все знаходиться у третій нормальній формі та в ньому відсутні будь-які залежності первинного ключа, а також ключових атрибутів від неключових атрибутів, то відношення знаходиться в посиленій третій нормальній формі або нормальній формі Бойса-Кодда (БКНФ)[6].

Прикладом бази даних в якій використовується метод нормальних форм є організаційно-інформаційна система керування вищим навчальним закладом «Курсор»[7].

Сенс другого методу полягає у графічному представленні сутностей предметної області, їх атрибутів та взаємозв'язків між ними.

Модель «Сутність-Зв'язок» вперше була представлена американським професором комп'ютерних наук Пітером Пін-Шен Ченом у 1976 році в університеті штату Луїзіана.

Процес проектування включає наступні етапи:

- 1) Виділення сутностей та зв'язків між ними;
- 2) Побудова діаграм ER-типу з урахуванням всіх сутностей та зв'язків між ними;
- 3) Формування набору попередніх відношень з вказівкою передбачуваного первинного ключа для кожного відношення та використання діаграм ER-типу;
- 4) Додавання неключових атрибутів у відношення;
- 5) Приведення попередніх відношень до нормальної форми Бойса-Кодда;

б) Перегляд діаграм ER-типу, якщо деякі відношення не приводяться до нормальної форми Бойса-Кодда або деякі атрибути не знаходять логічно обґрунтованого місця в попередніх відношеннях.

Процес проектування є ітераційним, тобто допускається повертання до попередніх етапів для перегляду раніше прийнятих рішень[6].

Прикладом є широке застосування діаграм ER-типу в Case -системах[8].

## **1.2 Засоби програмної реалізації**

Програми для роботи з базами даних дозволяють - зберігати, обробляти, редагувати та отримувати інформацію.

Існує велика кількість багатofункціональних програм для полегшення адміністрування баз даних, наведемо приклади:

1. MySQL Workbench – інструмент для візуального проектування, моделювання, створення та експлуатації баз даних, має ряд переваг:

1) Базу даних можна представити у графічному вигляді, а також редагувати дані в таблиці;

2) Простий механізм створення зв'язків між полями таблиць, серед яких реалізовано зв'язок «багато-до-багатьох» з можливістю створення таблиці зв'язків;

3) Наявність редактора SQL-запитів, який дає можливість при надсиланні на сервер, отримувати відповідь у табличному вигляді.

Основний недолік – складність в розумінні для звичайного користувача[9].

Інтерфейс програми MySQL Workbench представлений на рисунку 1.2[10].

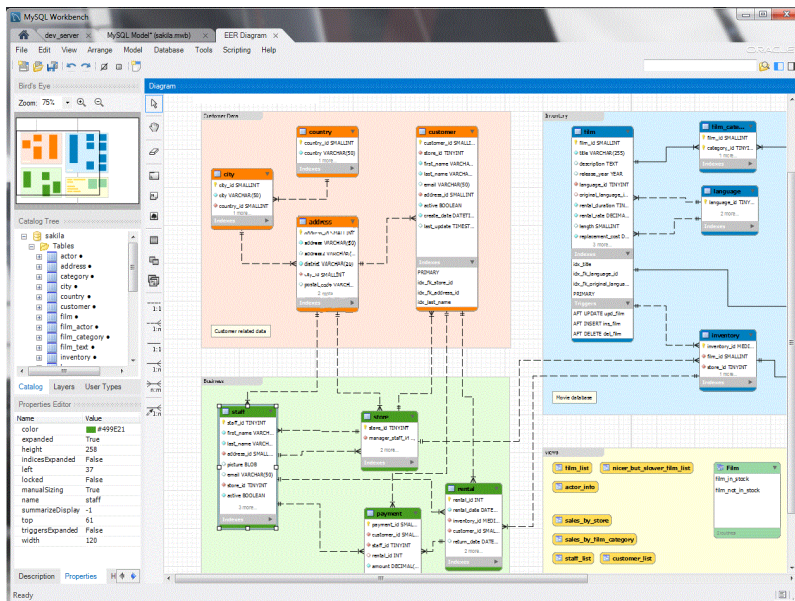


Рисунок 1.2 – Інтерфейс програми MySQL Workbench

2. Navicat – інструмент для розробки та адміністрування баз даних, який працює на будь-якому сервері MySQL, має ряд переваг:

- 1) Наявність візуального конструктору запитів;
- 2) Можливість створювати звіти;
- 3) Наявність можливості імпорту, експорту та резервного копіювання даних;
- 4) Широкий вибір інструментів для планування задач.

Основні недоліки – велика вартість та складність розуміння для звичайного користувача[9].

Інтерфейс програми Navicat представлений на рисунку 1.3[11].

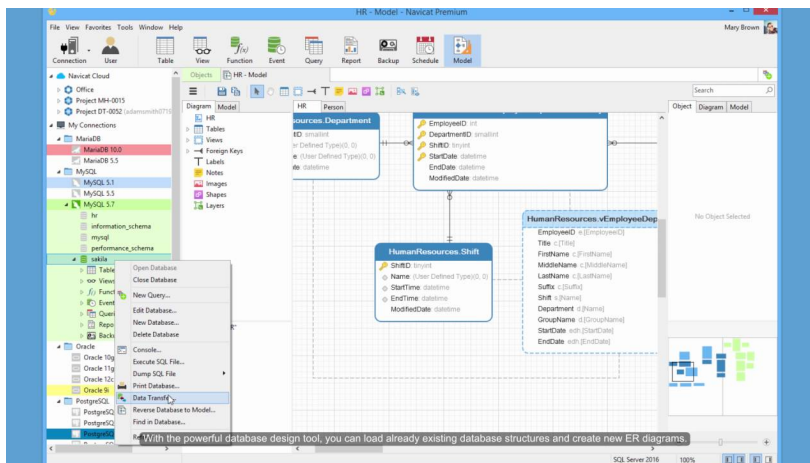


Рисунок 1.3 – Інтерфейс програми Navicat

3. HeidiSQL – безкоштовний інструмент управління базами даних, який дозволяє створювати та редагувати таблиці, надає можливість експорту даних, як SQL файл, так і в буфер обміну на інших серверах, має ряд переваг:

- 1) Можливість підключення до серверу за допомогою командного рядка;
- 2) Можливість пакетної оптимізації та відновлення таблиць;
- 3) Можливість редагування стовпців, індексів та зовнішніх ключів;
- 4) Синхронізація таблиці між різними базами даних.

Основний недолік – складний інтерфейс програми[9].

Інтерфейс програми HeidiSQL представлений на рисунку 1.4[12].

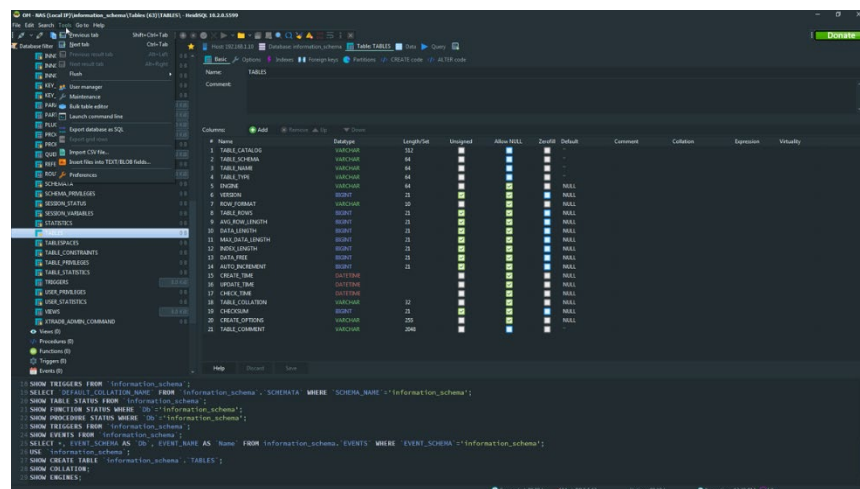


Рисунок 1.4 - Інтерфейс програми HeidiSQL

4. Microsoft Access – система управління базами даних, входить у склад професійної редакції Microsoft Office. Займає одне з провідних місць серед систем для проектування, створення та обробки баз даних, має ряд переваг:

1) Легкий та зрозумілий графічний інтерфейс, який дозволяє не тільки створювати базу даних, а й розробляти програми використовуючи вбудовані засоби;

2) Збереження всіх даних в одному файлі;

3) Постійно оновлюється виробником, підтримує багато мов;

4) Повністю сумісна з операційною системою Windows;

5) Орієнтована на користувача з різною професійною підготовкою, що виявляється в великій кількості «Майстрів» та розвиненій системі довідки;

- 6) Наявність можливості імпорту та експорту даних у різні формати;
- 7) Наявність розвинених вбудованих засобів розробки додатків;
- 8) Наявність вбудованої мови макрокоманд.

Основний недолік – не розповсюджується безкоштовно[13].

Інтерфейс програми Microsoft Access представлений на рисунку 1.5[14].

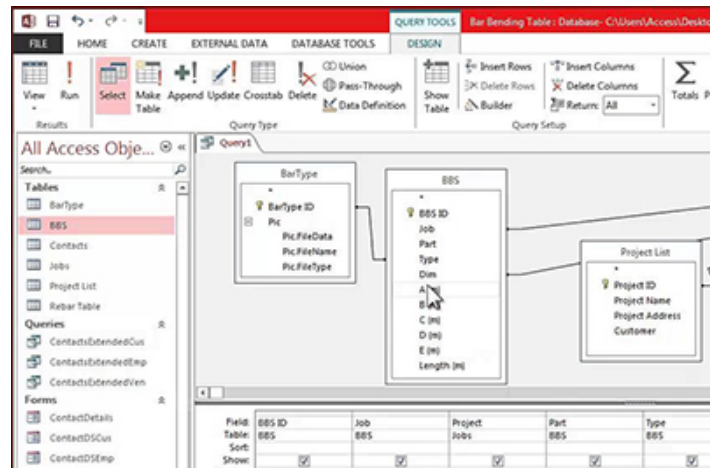


Рисунок 1.5 - Інтерфейс програми Microsoft Access

### 1.3 Основні висновки

1. Найбільше розповсюдження отримала реляційна модель бази даних, яка використовує математичний апарат відношень. Для подальшого виконання кваліфікаційної роботи обрано метод «Сутність – Зв'язок» в якому використовується графічний спосіб представлення сутностей та зв'язків між ними за допомогою ER-діаграм, задля зручності проектування та підвищення наочності.

2. Для програмної реалізації бази даних запропоновані такі програмні продукти: MySQL Workbench, Navicat, HeidiSQL, Microsoft Access. Обрано програму Microsoft Access, тому що більшість реляційних баз даних, окрім Access, складається з двох окремих компонентів – «back-end», де зберігаються дані та «front-end» - інтерфейс користувача для взаємодії з даними. Такий тип конструкції досить складний, Microsoft Access має більш зрозумілий інтерфейс для будь-якого користувача.

## 2 ПРОЕКТУВАННЯ, СТВОРЕННЯ ТА ОБРОБКИ ЗАХИЩЕНОЇ БАЗИ ДАНИХ

### 2.1 Постановка задачі

Вхідними даними для проектування бази даних є дані репозитарію Дніпровського інституту інфраструктури і транспорту (ДІІТ) з науковими розробками (Назва\_роботи, Вид\_роботи, Кількість\_сторінок, Рік\_видання, Місце\_видання) та даними про науково-педагогічний склад кафедри ЕОМ (ПІБ\_автора, Посада\_автора, Ступінь, Місце\_роботи, ORCID).

Основною задачею, що обмірковується в процесі проектування реляційної бази даних, являється задача нормалізації її відношень, використовуючи графічний метод «сутність-зв'язок», який заснований на використанні діаграм ER-екземплярів та діаграм ER-типу. Захист та підвищення ефективності бази даних, можливе на основі використання засобів оптимальності, з одного боку оптимальності бази даних, з іншого боку – оптимальності операторів SQL.

### 2.2 Проектування реляційної бази даних за методом «Сутність-Зв'язок»

Складено початкове відношення «БД\_ІІТ\_ЕОМ», яке представлено в Додатку А. Назва відношення означає: база даних наукових розробок кафедри ЕОМ, воно знаходиться в першій нормальній формі, тому що всі його атрибути атомарні.

Пояснення щодо атрибутів відношення:

- 1) ПІБ\_автора – прізвище, ім'я, по-батькові автора наукової роботи;
- 2) Місце\_роботи – назва навчального закладу в якому працює автор;
- 3) Посада\_автора – назва посади, яку займає автор;
- 4) Ступінь – ступінь, яку отримав автор наукової розробки;
- 5) Електронна\_пошта – електронна адреса автора;
- 6) Назва\_роботи – назва роботи наукової праці;
- 7) Вид\_роботи – назва виду роботи наукової праці.
- 8) ORCID – код, який ідентифікує автора наукової роботи;
- 9) ISSN – код, який ідентифікує друковане чи цифрове видання;
- 10) Місце\_видання – місце, де надрукована наукова робота;

- 11) Рік\_видання – рік у якому видана наукова розробка;
- 12) Кількість\_сторінок – загальна кількість сторінок в роботі;
- 13) Країна\_видання – назва країни в якій знаходиться місце видання роботи.

У даному початковому відношенні ступінь дорівнює 13, а потужність 160.

Первинний ключ відношення складений, складається з трьох атрибутів (ПІБ\_автора, Назва\_роботи, Рік\_видання).

На першому етапі проектування необхідно виявити сутності та зв'язки між ними.

Виділено наступні сутності (ключі сутності підкреслені):

- АВТОР (ПІБ\_автора);
- РОЗРОБКА (Назва\_роботи, Рік\_видання);
- ВИДАННЯ (Місце\_видання);
- ПОСАДА (Посада\_автора).

Виділено зв'язки між сутностями:

- АВТОР ПИШЕ РОЗРОБКУ;
- АВТОР ЗАЙМАЄ ПОСАДУ;
- РОЗРОБКА ДРУКУЄТЬСЯ ВИДАННЯМ.

На другому етапі проектування необхідно побудувати діаграми ER-типу з урахуванням всіх сутностей та зв'язків між ними.

Діаграма ER-типу відношення «БД\_НР\_ЕОМ» представлена на рисунку 2.1.



Рисунок 2.1 - Діаграма ER-типу відношення «БД\_НР\_ЕОМ»

Зв'язок ПИШЕ має ступінь «багато-до-багатьох», тому що автор може писати декілька наукових розробок, а також при цьому в одній науковій розробці може бути декілька авторів.

Обидві сутності в даному зв'язку мають обов'язковий клас приналежності, в припущенні, що не має авторів, які не пишуть наукові розробки, а також немає наукових розробок в яких немає автора.

Зв'язок ЗАЙМАЄ має ступінь «багато-до-одного», тому що кожен автор займає певну посаду (професор, доцент, старший викладач, асистент, студент), а також однакові посади можуть займати декілька авторів.

Сутність АВТОР має обов'язковий клас приналежності, оскільки у кожного автора обов'язково повинна бути посада. Сутність ПОСАДА має необов'язковий клас приналежності, наприклад, відсутність автора з посадою асистент в наукових розробках.

Зв'язок ДРУКУЄТЬСЯ має ступінь «багато-до-одного», тому що кожна наукова розробка друкується певним виданням, та одне й те саме видання може друкувати декілька наукових розробок.

Сутність РОЗРОБКА має обов'язковий клас приналежності, оскільки кожна наукова розробка повинна бути випущена виданням. Сутність ВИДАННЯ має необов'язковий клас приналежності, оскільки можливі такі назви видавництва в яких не випущено розробок кафедри «Електронних обчислювальних машин».

На третьому етапі формуємо набір попередніх відношень на основі правил з первинними ключами для кожного відношення, використовуючи діаграми ER-типу.

1. Зв'язок АВТОР ПИШЕ РОЗРОБКУ задовольняє умовам правила шість, відповідно до якого отримуємо три відношення:

- АВТОР (ПІБ\_автора, Посада\_автора,
- РОЗРОБКА (Назва\_роботи, Місце\_видання,
- ПИШЕ (ПІБ\_автора, Назва\_роботи, Рік\_видання,

Графічна інтерпретація зв'язку АВТОР ПИШЕ РОЗРОБКУ за правилом шість представлена на рисунку 2.2.

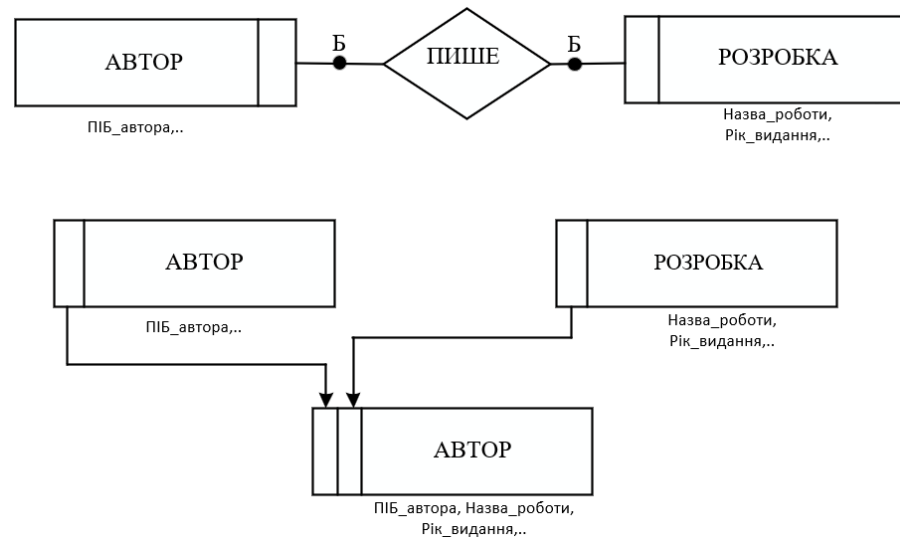


Рисунок 2.2 – Графічне представлення зв'язку АВТОР ПІШЕ РОЗРОБКУ за правилом шість

2. Зв'язок АВТОР ЗАЙМАЄ ПОСАДУ задовольняє умовам правила чотири, відповідно до якого отримуємо два відношення:

- АВТОР (ПІБ\_автора, Посада\_автора,
- ПОСАДА (Посада\_автора,

Графічна інтерпретація зв'язку АВТОР ЗАЙМАЄ ПОСАДУ за правилом чотири представлена на рисунку 2.3.

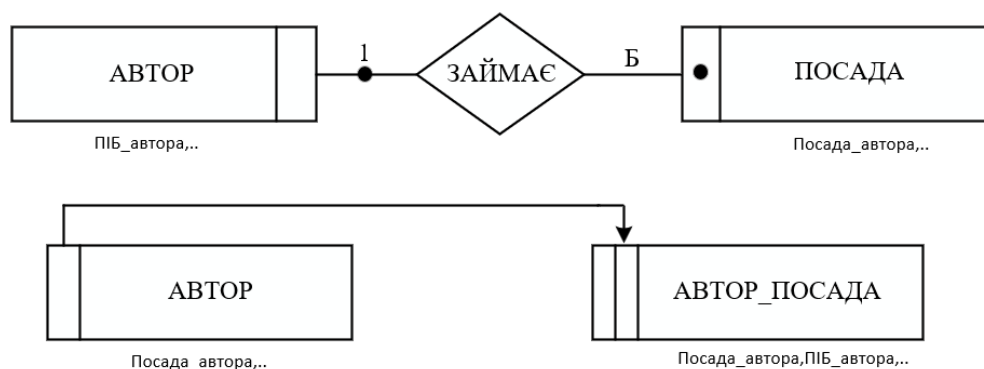


Рисунок 2.3 – Графічне представлення зв'язку АВТОР ЗАЙМАЄ ПОСАДУ за правилом чотири

3. Зв'язок РОЗРОБКА ДРУКУЄТЬСЯ ВИДАННЯМ задовольняє умовам правила чотири, відповідно до якого отримуємо два відношення:

- РОЗРОБКА (Назва\_роботи, Місце\_видання,
- ВИДАННЯ (Місце\_видання,

Графічна інтерпретація зв'язку РОЗРОБКА ДРУКУЄТЬСЯ ВИДАННЯМ за правилом чотири представлена на рисунку 2.4.

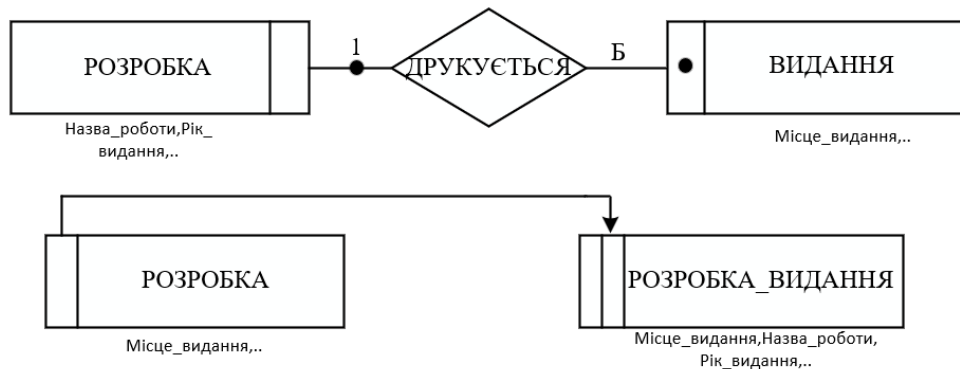


Рисунок 2.4 – Графічне представлення зв'язку РОЗРОБКА ДРУКУЄТЬСЯ ВИДАННЯМ за правилом чотири

На четвертому етапі необхідно додати неключові атрибути, які не були обрані в якості ключових раніше, та призначення їх одному з відношень з тією умовою, щоб відношення відповідали вимогам нормальної форми Бойса-Кодда.

Після додавання неключових атрибутів відношення мають наступний вигляд:

1. АВТОР(ПІБ\_автора, Посада\_автора, ORCID, Місце\_роботи, Електронна\_пошта);
2. РОЗРОБКА(Назва\_роботи, Місце\_видання, ISSN, Кількість\_сторінок);
3. ПИШЕ(ПІБ\_автора, Назва\_роботи, Рік\_видання, Вид\_роботи);
4. ПОСАДА(Посада\_автора, Ступінь);
5. ВИДАННЯ(Місце\_видання, Країна\_видання).

Всі результуючі відношення представлені у Додатку А.

Після визначення всіх відношень необхідно їх перевірити на відповідність вимогам БКНФ. Усі вихідні відношення відповідають вимогам БКНФ.

Таким чином, отримано ті ж відношення, що і при проектуванні бази даних методом нормальних форм. Проектування методом нормальних форм представлено у Додатку Б.

Відношення АВТОР співпадає з відношенням R4.

Відношення РОЗРОБКА співпадає з відношенням R6.

Відношення ПИШЕ співпадає з відношенням R1.

Відношення ПОСАДА співпадає з відношенням R5.

Відношення ВИДАННЯ співпадає з відношенням R7.

Структура спроектованої бази даних за методом «сутність-зв'язок» представлена на рисунку 2.5.

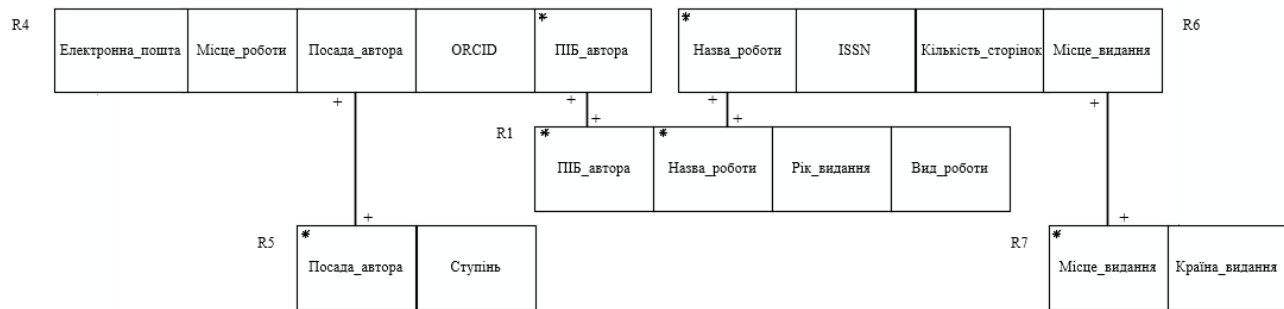


Рисунок 2.5 - Структура спроектованої бази даних за методом «сутність-зв'язок»

### 2.3 Створення та обробка бази даних в Microsoft Access з використанням мови SQL

За допомогою запитів SQL створено п'ять таблиць відношень, а саме: АВТОР, РОЗРОБКА, ПИШЕ, ПОСАДА, ВИДАННЯ.

Відношенню АВТОР відповідає таблиця з назвою АВТОР.

Відношенню РОЗРОБКА відповідає таблиця з назвою НАУКОВА\_РОЗРОБКА.

Відношенню ПИШЕ відповідає таблиця з назвою БД\_НР\_ЕОМ, що означає БАЗА\_ДАНИХ\_НАУКОВИХ\_РОЗРОБОК\_КАФЕДРИ\_ЕОМ.

Відношенню ПОСАДА відповідає таблиця з назвою ПОСАДА.

Відношенню ВИДАННЯ відповідає таблиця з назвою ВИДАННЯ.

Запити реалізовані за допомогою «Конструктору запитів», який знаходиться на вкладці «Створення» в групі «Запити».

Код створення таблиці АВТОР, представлено на рисунку 2.6.

```

CREATE TABLE АВТОР
(
ПІБ_автора TEXT(35) PRIMARY KEY,
ORCID TEXT(40),
Посада_автора TEXT(30),
Місце_роботи TEXT(70),
Електронна_пошта TEXT(50)
);

```

Рисунок 2.6 – Код створення таблиці АВТОР

У таблиці АВТОР всі атрибути мають тип даних – короткий текст.

Код створення таблиці НАУКОВА\_РОЗРОБКА, представлено на рисунку 2.7.

```

CREATE TABLE НАУКОВА_РОЗРОБКА
(
Назва_роботи MEMO PRIMARY KEY,
ISSN TEXT(70),
Кількість_сторінок INT,
Місце_видання TEXT(70)
);

```

Рисунок 2.7 – Код створення таблиці НАУКОВА\_РОЗРОБКА

У таблиці НАУКОВА\_РОЗРОБКА атрибут «Назва\_роботи» має тип даних – довгий текст, атрибути «ISSN» та «Місце\_видання» мають тип даних – короткий текст, а атрибут «Кількість\_сторінок» має числовий тип даних.

Код створення таблиці БД\_НР\_ЕОМ, представлено на рисунку 2.8.

```

CREATE TABLE БД_НР_ЕОМ
(
ПІБ_автора TEXT(70) PRIMARY KEY,
Назва_роботи MEMO PRIMARY KEY,
Рік_видання INT PRIMARY KEY,
Вид_роботи TEXT(70)
);

```

Рисунок 2.8 – Код створення таблиці БД\_НР\_ЕОМ

У таблиці БД\_НР\_ЕОМ атрибут «Назва\_роботи» має тип даних – довгий текст, атрибути «ПІБ\_автора» та «Вид\_роботи» мають тип даних – короткий текст, а атрибут «Рік\_видання» має числовий тип даних.

Код створення таблиці ПОСАДА, представлено на рисунку 2.9.

```
CREATE TABLE ПОСАДА
(
Посада_автора TEXT(30) PRIMARY KEY,
Ступінь TEXT(70)
);
```

Рисунок 2.9 – Код створення таблиці ПОСАДА

У таблиці ПОСАДА всі атрибути мають тип даних – короткий текст.

Код створення таблиці ВИДАННЯ, представлено на рисунку 2.10.

```
CREATE TABLE ВИДАННЯ
(
Місце_видання TEXT(70) PRIMARY KEY,
Країна_видання TEXT(70)
);
```

Рисунок 2.10 – Код створення таблиці ВИДАННЯ

У таблиці ВИДАННЯ всі атрибути мають тип даних – короткий текст.

Після створення основних таблиць відношень, створенні таблиці такої ж структури, які заповненні даними в діалоговому режимі.

За допомогою оператора INSERT відбувалось копіювання з поточних таблиць в загальні таблиці відношення.

Код перенесення даних з таблиці АВТОР\_копія в таблицю АВТОР, представлено на рисунку 2.11.

```
INSERT INTO АВТОР
SELECT *
FROM АВТОР_копія;
```

Рисунок 2.11 – Код перенесення даних в таблицю АВТОР

Код перенесення даних з таблиці НАУКОВА\_РОЗРОБКА\_копія в таблицю НАУКОВА\_РОЗРОБКА, представлено на рисунку 2.12.

```
INSERT INTO НАУКОВА_РОЗРОБКА  
SELECT *  
FROM НАУКОВА_РОЗРОБКА_копія;
```

Рисунок 2.12 – Код перенесення даних в таблицю НАУКОВА\_РОЗРОБКА

Код перенесення даних з таблиці БД\_НР\_ЕОМ\_копія в таблицю БД\_НР\_ЕОМ, представлено на рисунку 2.13.

```
INSERT INTO БД_НР_ЕОМ  
SELECT *  
FROM БД_НР_ЕОМ_копія;
```

Рисунок 2.13 – Код перенесення даних в таблицю БД\_НР\_ЕОМ

Код перенесення даних з таблиці ПОСАДА\_копія в таблицю ПОСАДА, представлено на рисунку 2.14.

```
INSERT INTO ПОСАДА  
SELECT *  
FROM ПОСАДА_копія;
```

Рисунок 2.14 – Код перенесення даних в таблицю ПОСАДА

Код перенесення даних з таблиці ВИДАННЯ\_копія в таблицю ВИДАННЯ, представлено на рисунку 2.15.

```
INSERT INTO ВИДАННЯ  
SELECT *  
FROM ВИДАННЯ_копія;
```

Рисунок 2.15 – Код перенесення даних в таблицю ВИДАННЯ

За допомогою вкладки «Робота з базами даних» в групі «Відношення», побудовано схему даних.

Схема даних представлена на рисунку 2.16.

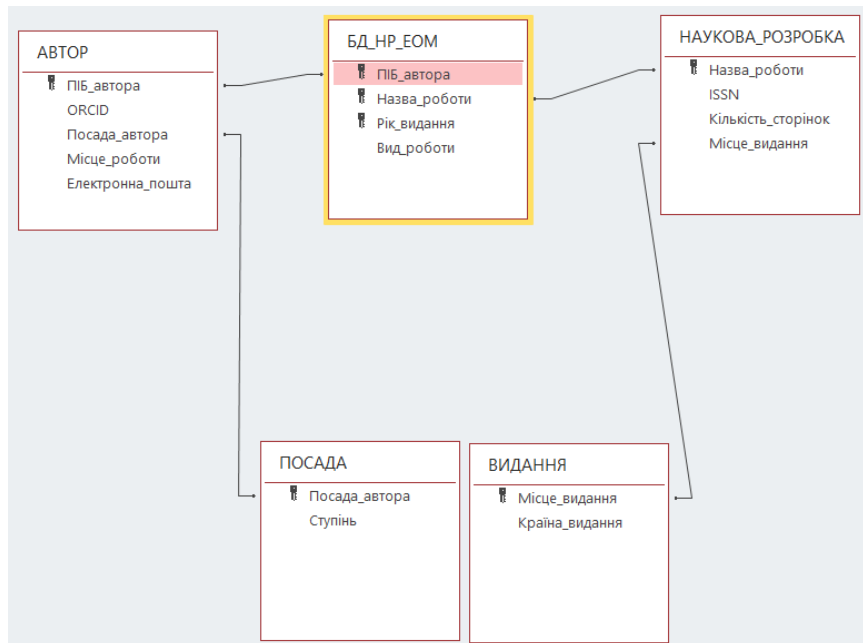


Рисунок 2.16 – Графічне представлення схеми даних

Результат зв'язування таблиць на прикладі таблиці АВТОР представлено на рисунку 2.17.

ПІБ_автора	ORCID	Посада_автора	Місце_роботи	Електронна_пошта
Антонов Владислав Миколайович	-	Студент	УДУНТ	-
Архипов Олександр Євгенович	-	Студент	УДУНТ	-
Бондар Борис Євгенович	0000-0002-3591-4772	Студент	УДУНТ	-
Бондар Євген Борисович	0000-0001-6040-913X	Студент	УДУНТ	-
Бондарева Валентина Сергіївна	0000-0002-4016-1656	Асистент	УДУНТ	-
Годун Єлизавета Дмитрівна	-	Студент	УДУНТ	-
Гринько Катерина Петрова	-	Студент	УДУНТ	-
Грошовенко Олексій Петрович	-	Студент	УДУНТ	-
Дзюба Володимир Володимирович	-	Старший викладач	УДУНТ	*****
Дмитрієв Святослав Юрійович	0000-0002-9512-4158	Студент	УДУНТ	-
Доманська Галина Анатоліївна	0000-0002-5746-299X	Доцент	УДУНТ	*****

Назва роботи	Рік видання	Вид роботи	Щезливість δ
Формування ІКТ-компетентності фахівців спеціальності «Кібербезпека» з використанням дослідницького підходу	2020	Стаття	
Вибір енергозберігаючих режимів роботи електроттягових мереж на основі нечіткого опису їх станів	2009	Стаття	
Інтелектуалізація управління режимами роботи систем тягового електропостачання	2010	Стаття	

Рисунок 2.17 – Таблиця АВТОР

У таблиці АВТОР п'ять атрибутів та п'ятьдесят один кортеж.

Результат зв'язування таблиць на прикладі таблиці НАУКОВА\_РОЗРОБКА представлено на рисунку 2.18.

НАУКОВА_РОЗРОБКА				
Назва роботи	ISSN	Кількість сторінок	Місце видання	ЦД
Ідентифікація поїздів в інформаційних системах залізничного транспорту	1562-9945	9	ДНМАУ	
Автоматизована ідентифікація рухомих одиниць	1681-4886	6	УкрДУЗТ	
Процедура ідентифікації поїздів з використанням інформації АСК ВП УЗ	1681-4886	6	УкрДУЗТ	
<b>Вивчення вебометрик</b>	1562-9945	6	ДНМАУ	
ПІБ автора	Рік видання	Вид роботи	Щелкните здесь	
Бондар Борис Євгенович	2015	Стаття		
Косолапов Анатолій Аркадійович	2015	Стаття		
Лобода Дмитро Геннадійович	2015	Стаття		

Рисунок 2.18 – Таблиця НАУКОВА\_РОЗРОБКА

У таблиці НАУКОВА\_РОЗРОБКА чотири атрибути та дев'яносто чотири кортежа.

Результат зв'язування таблиць на прикладі таблиці БД\_НР\_ЕОМ представлено на рисунку 2.19.

БД_НР_ЕОМ				
ПІБ автора	Назва роботи	Рік видання	Вид роботи	
Егоров Олег Йосипович	Імітаційна модель ідентифікації рухомих одиниць та поїзда в цілому	2015	Стаття	
Егоров Олег Йосипович	Підвищення достовірності ідентифікації рухомих одиниць на станціях сортування залізничного транспорту	2013	Автореферат	
Егоров Олег Йосипович	Удосконалення процедури визначення кількості та осності рухомих одиниць	2015	Стаття	
Егоров Богдан Вікторович	Інноваційні підходи до розвитку техніки і технологій	2015	Монографія	
Жуковичкий Ігор Володимирович	Ідентифікація поїздів в інформаційних системах залізничного транспорту	2015	Стаття	
Жуковичкий Ігор Володимирович	Автоматизована ідентифікація рухомих одиниць	2012	Стаття	

Рисунок 2.19 – Таблиця БД\_НР\_ЕОМ

У таблиці БД\_НР\_ЕОМ чотири атрибути та сто шістьдесят один кортеж.

Результат зв'язування таблиць на прикладі таблиці ПОСАДА представлено на рисунку 2.20.

Посада_автора	Ступінь	Щелкните для добавления	ПІБ_автора	ORCID	Місце роботи	Електронна пошта
Асистент	-					
Доцент	Кандидат технічних наук					
Доманська Галина Анатоліївна			0000-0002-5746-299X	УДУНТ	*****	
Егоров Олег Йосипович			0000-0002-8260-9463	УДУНТ	*****	
Остапеш Денис Олександрович			0000-0003-1778-7770	УДУНТ	*****	
Пахомова Вікторія Миколаївна			0000-0002-0022-099X	УДУНТ	*****	
Пахомова Вікторія Миколаївна			0000-0001-8346-0405	УДУНТ	*****	
Устенко Андрій Борисович			0000-0002-8677-4781	УДУНТ	*****	
Професор	Доктор технічних наук					
Старший викладач	-					
Студент	-					

Рисунок 2.20 – Таблиця ПОСАДА

У таблиці ПОСАДА два атрибути та п'ять кортежей.

Результат зв'язування таблиць на прикладі таблиці ВИДАННЯ представлено на рисунку 2.21.

Місце видання	Країна видання	Щелкните для добавления	Назва роботи	ISSN	Кількість сторінок	Щелкните д
«ISE&E» & Sworld	Сполучені Штати Америки					
Формування ІКТ-компетентності фахівців спеціальності «Кибербезпека» з використанням дослідницького підходу			2567-5273	6		
Доманська Галина Анатоліївна			2020	Стаття		
Пахомова Вікторія Миколаївна			2020	Стаття		
Формування компетентностей у здобувачів ступеня «бакалавр» при дистанційному навчанні з дисципліни «Локальні мережі»			2567-5273	7		

Рисунок 2.21 – Таблиця ВИДАННЯ

У таблиці ВИДАННЯ два атрибути та двадцять кортежей.

## 2.4 Організація захисту бази даних

У Microsoft Access існують наступні способи захисту бази даних: парольний захист, відображення та приховування об'єктів у вікні бази даних, шифрування й захист на рівні користувачів.

Парольний захист бази даних є простим та часто достатнім засобом забезпечення захисту бази даних для обмеження доступу несанкціонованим користувачем.

Знаючи пароль бази даних, будь-який користувач може її відкрити, а також виконати всі необхідні операції з нею. Встановлення пароля може бути неможливе, якщо для бази даних установлений захист на рівні користувача та заборона на парольний захист. Парольний захист може використовуватися додатково для захисту на рівні користувача, в цьому випадку встановлювати пароль може користувач, який володіє правами адміністратора бази даних.

Спосіб захисту об'єктів у базі даних від сторонніх користувачів є приховування об'єктів у вікні бази даних.

Спосіб захисту бази даних – шифрування, при кодуванні бази даних її файл стискається та стає недоступним для читання за допомогою службових програм або текстових редакторів.

Захист на рівні користувачів використовується, коли з однією базою даних працює кілька користувачів або груп користувачів, які мають різні права доступу до об'єктів бази даних[6].

В даній роботі використовувалась версія Microsoft Access 2019, яка має ряд переваг для організації захисту бази даних в порівнянні з попередніми версіями Microsoft Access.

Нові функції Microsoft Access 2019 для підвищення безпеки даних:

1. Можливість зберігати дані на сервері баз даних, який керує безпекою користувачів, наприклад Microsoft SQL Server, а також за допомогою Microsoft Access можна створювати запити, форми та звіти пов'язуючи їх із даними на сервері;

2. Більш стійкий алгоритм шифрування бази даних у форматі ACCDB з використанням пароля, а також у процесі шифрування відбувається перемішування даних у таблицях, що робить неможливим їх несанкціонований перегляд;

3. Новий підклас макрокоманд, які виконуються при відключеній базі даних, вони також включають можливість обробки помилок;

4. Підтримка програм шифрування сторонніх компаній;

5. Нові способи підписання та розповсюдження файлів баз даних, якщо вилучити базу даних з підписаного пакету в надійне розташування, вона відкриється без виведення панелі повідомлень. Якщо база даних з підписаного пакету вилучається в ненадійне розташування, але наявний надійний сертифікат пакету, а також дійсний підпис, то база даних відкриється без відображення панелі повідомлень[15].

Особливості функції Microsoft Access 2010 для забезпечення безпеки даних:

1. Можливість перегляду даних без увімкнення змісту бази даних;

2. Зручність використання, якщо файли бази даних знаходяться у надійному розташуванні, то вони відкриватимуться і виконуватимуться без видачі попереджень та запитів про включення вимкненого змісту[16].

Нові функції Microsoft Access 2019 є більш актуальними та забезпечують вищий рівень безпеки даних.

#### **2.4.1 Парольний захист**

Для організації парольного захисту бази даних, спочатку необхідно закрити її, потім за допомогою провідника відкрити базу даних в режимі «Монопольно».

Відкриття бази даних представлено на рисунку 2.22.

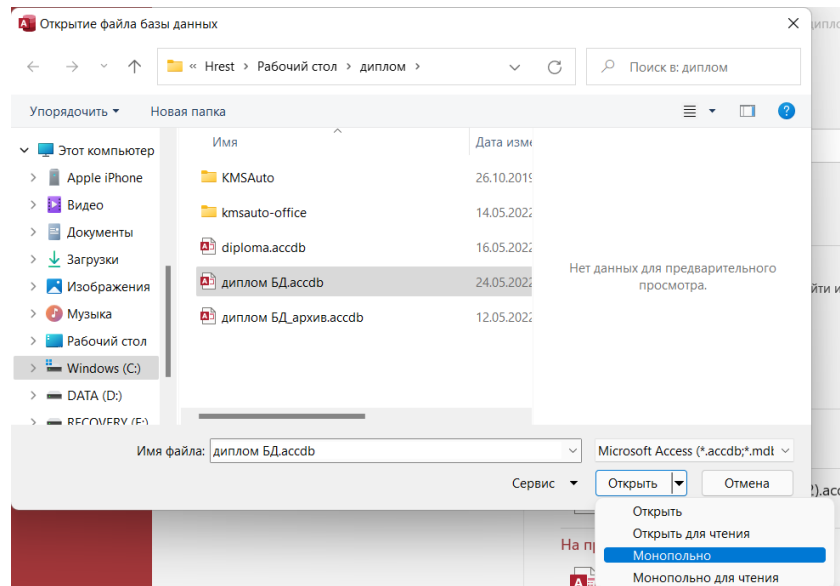


Рисунок 2.22 – Відкриття бази даних

Для встановлення пароля заходимо «Файл» - «Відомості» - «Зашифрувати паролем».

Процес встановлення пароля представлено на рисунку 2.23.

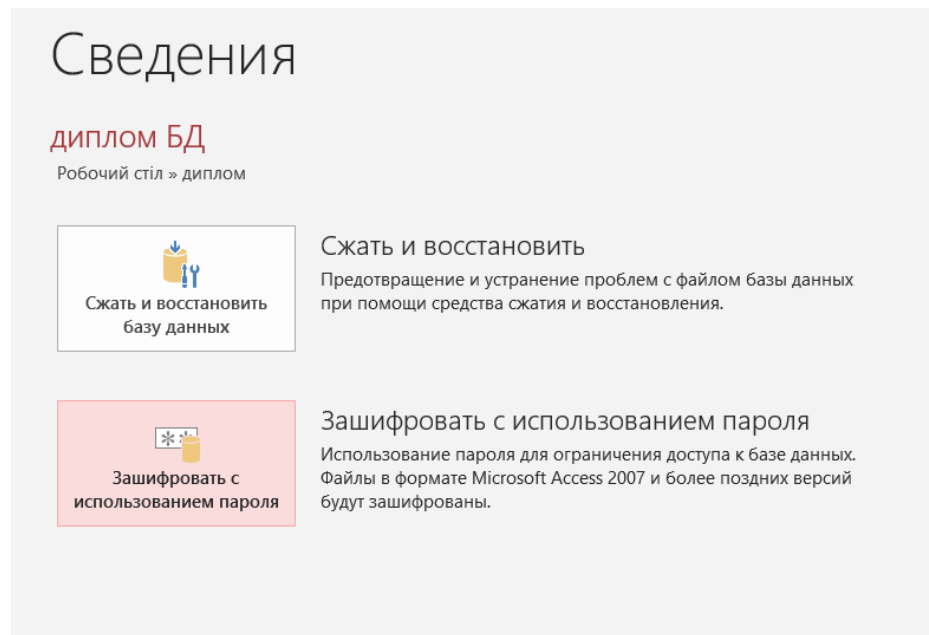


Рисунок 2.23 – Встановлення пароля

Задання пароля представлено на рисунку 2.24.

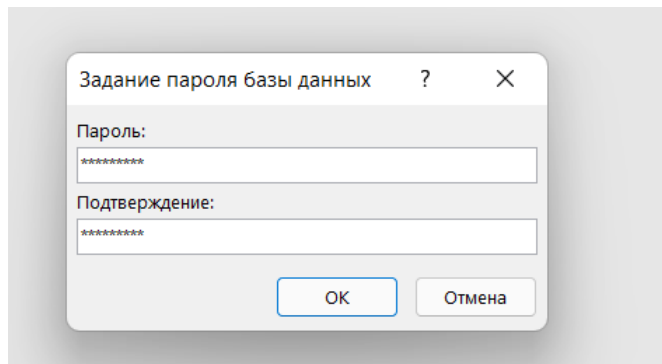


Рисунок 2.24 – Задання пароля

Тепер, при відкритті бази даних треба обов'язково вводити пароль.

Вікно вводу пароля при спробі відкриття бази даних представлено на рисунку 2.25.

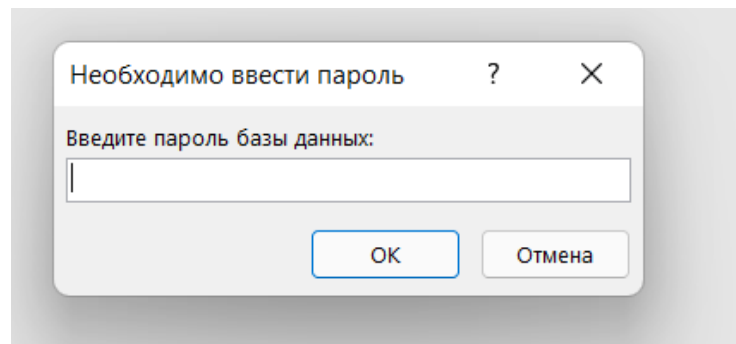


Рисунок 2.25 – Вікно вводу пароля

#### 2.4.2 Відображення та приховування об'єктів у вікні бази даних

Для того, щоб приховати таблицю в базі даних, необхідно натиснути на неї правою кнопкою миші, в контекстному меню обрати «Властивості» та поставити галочку в «Атрибути» - скриті.

Приховування таблиці АВТОР представлено на рисунку 2.26.

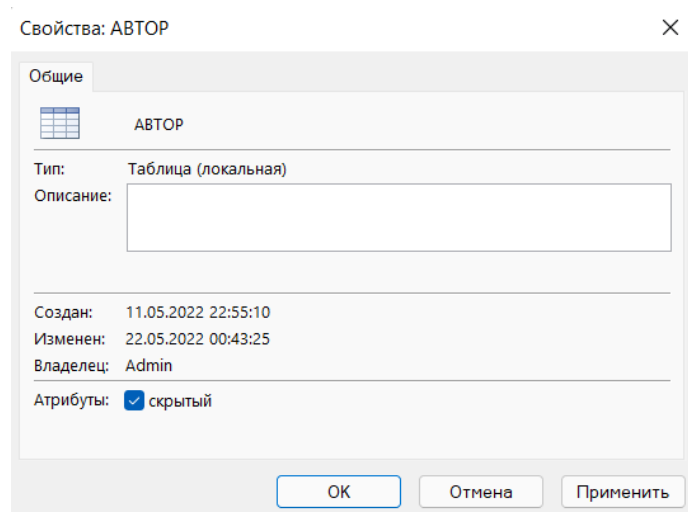


Рисунок 2.26 - Приховування таблиці АВТОР

Аналогічні дії були проведені над таблицями – НАУКОВА\_РОЗРОБКА, БД\_НР\_ЕОМ, ВИДАННЯ, ПОСАДА.

Для того, щоб таблиці знову стали видимими, необхідно зайти в «Параметри» - «Поточна база даних» - «Параметри навігації» та поставити галочку в пункті «Відображати скриті об'єкти».

Настройка відображення таблиць представлена на рисунку 2.27.

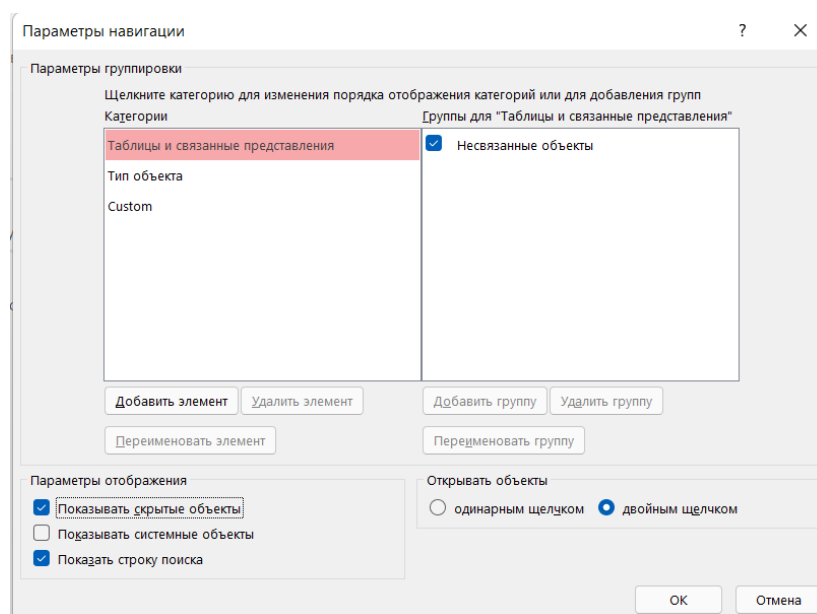


Рисунок 2.27 - Настройка відображення таблиць

В результаті чого з'являються тусклі іконки таблиць.

Відображення скритих таблиць представлено на рисунку 2.28.

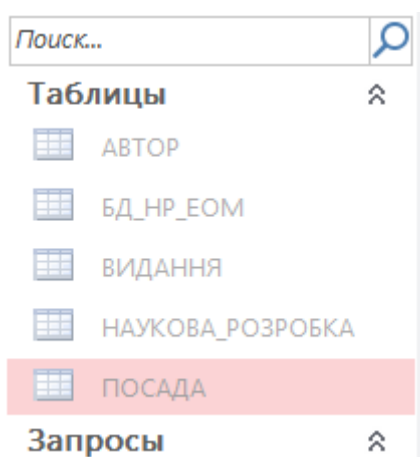


Рисунок 2.28 - Відображення скритих таблиць

Далі необхідно натиснути правою кнопкою миші на таблицю, обрати в контекстному меню «Властивості» та зняти галочку в «Атрибути» - скриті.

Відображення таблиці АВТОР представлено на рисунку 2.29.

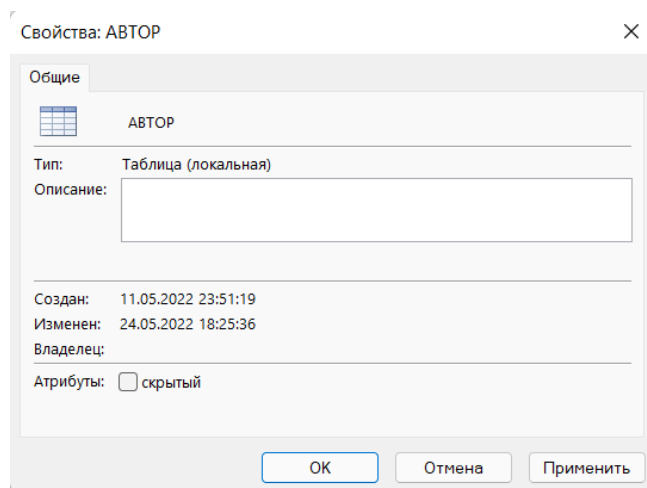


Рисунок 2.29 - Відображення таблиці АВТОР

Аналогічні дії були проведені над таблицями – НАУКОВА\_РОЗРОБКА, БД\_НР\_ЕОМ, ВИДАННЯ, ПОСАДА.

### 2.4.3 Шифрування та захист на рівні користувачів

Захист на рівні користувача можливий лише в версії Access 2007 та нижче.

Як сказано на офіційному сайті: «Безпека на рівні користувачів недоступна для баз даних, створених у Access 2007 або пізніших версіях (ACCDB-файли). Крім того, при перетворенні MDB-файлу на новий формат (ACCDB-файл) Access відхиляє параметри безпеки на рівні користувача.»[16].

Тому базу даних додано до центру керування безпекою, для цього необхідно зайти у вкладку «Параметри» та обрати «Центр керування безпекою».

Вкладка параметри представлена на рисунку 2.30.

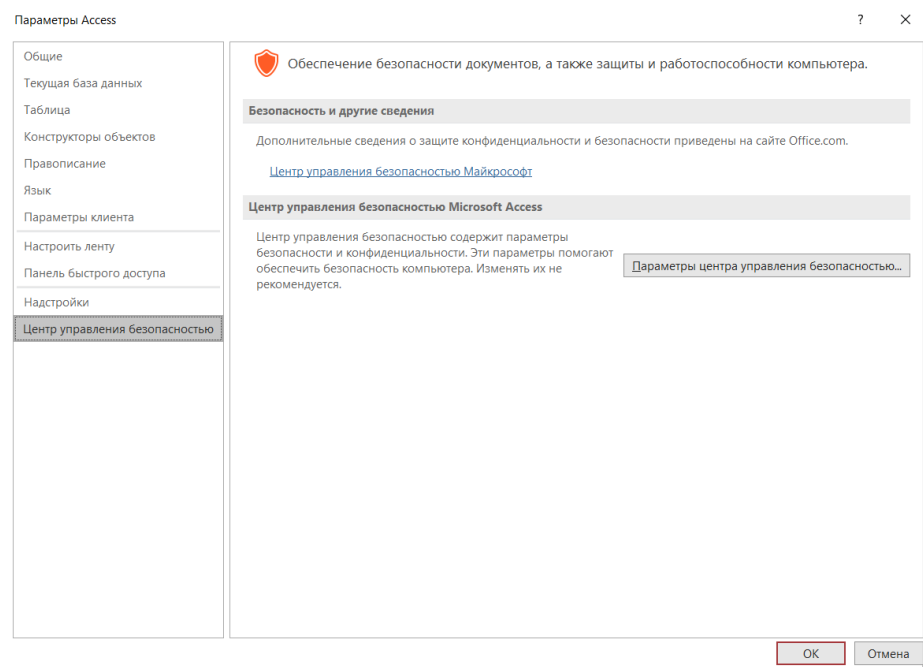


Рисунок 2.30 – Вкладка «Параметри»

Обираємо пункт «Параметри центра керування безпекою» - «Надійні розташування».

Вкладка «Надійні розташування» представлена на рисунку 2.31.

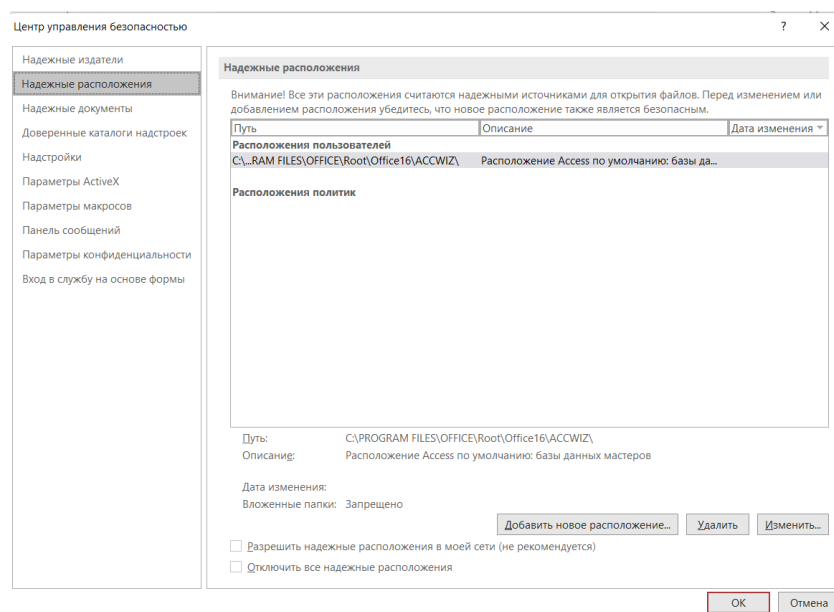


Рисунок 2.31 - Вкладка «Надійні розташування»

Необхідно додати папку в якій буде знаходитися база даних.

Додавання папки представлено на рисунку 2.32.

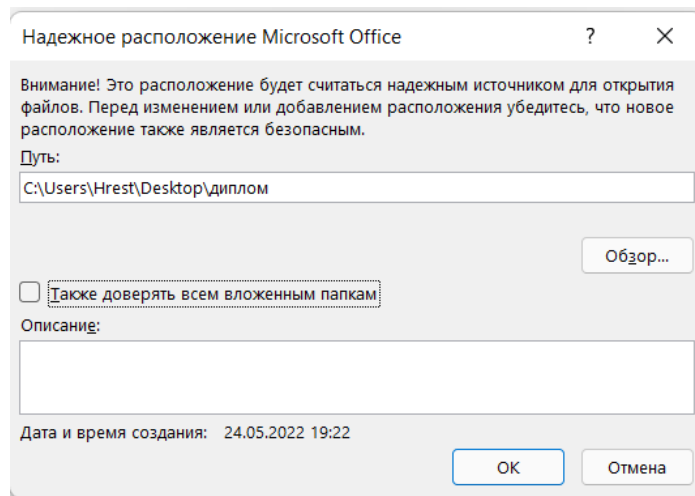


Рисунок 2.32 – Додавання папки «Диплом»

Після цього необхідно упаковати та підписати базу даних, для цього треба перейти в вкладку «Файл» - «Зберегти як» - «Упакувати та підписати».

Процес упаковування та підписання бази даних представлено на рисунку 2.33.

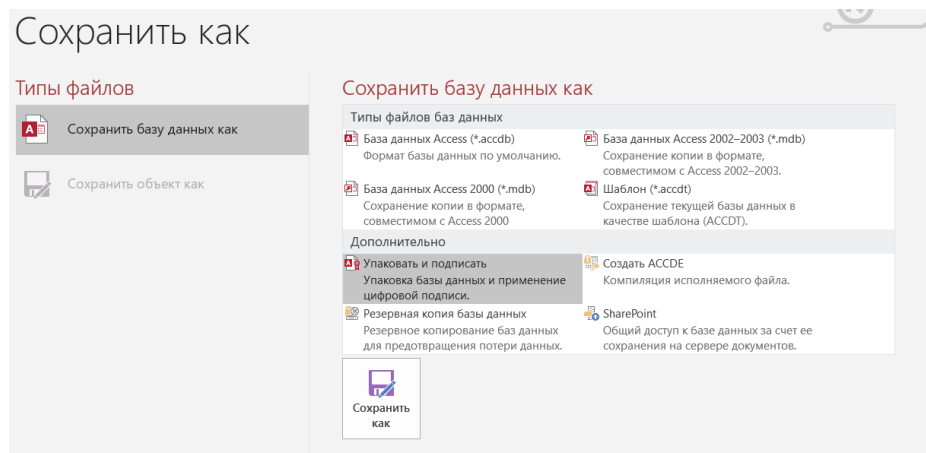


Рисунок 2.33 - Процесс упаковування та підписання бази даних

Далі необхідно обрати сертифікат, обрано сертифікат створений 01.01.2022, дійсний до 01.01.2028.

Сертифікат представлено на рисунку 2.34.

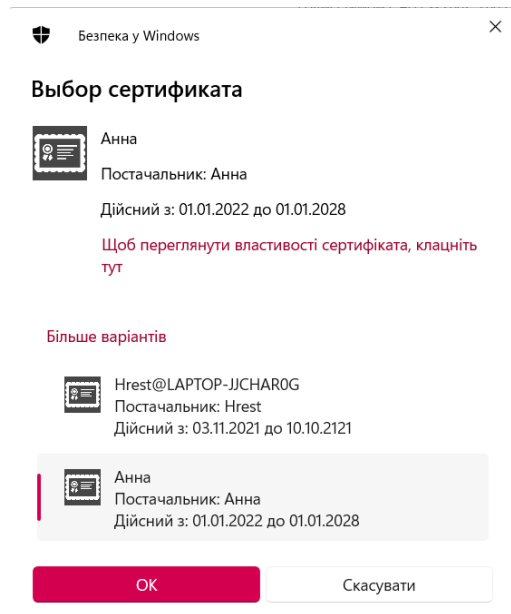


Рисунок 2.34 – Вибір сертифіката

Після цього необхідно створити підписаний пакет Microsoft Access. Створення підписаного пакету представлено на рисунку 2.35.

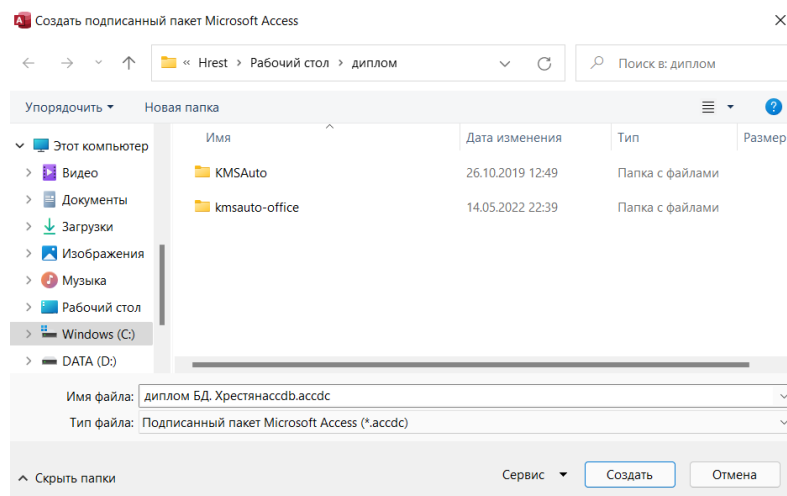


Рисунок 2.35 - Створення підписаного пакету

Тепер упакована та підписана база даних знаходиться у папці «Диплом».

## 2.5 Підвищення ефективності створеної бази даних та її використання

Для досягнення оптимальної швидкості роботи з базою даних необхідно, як оптимізувати спроектовану базу даних (це зроблено при нормалізації відношень), так і оптимізувати використані оператори SQL для доступу до бази даних. З одного боку, погано оптимізована база даних потребує багато зайвих зусиль з оптимізації операторів SQL, з іншого боку, навпаки, гарна оптимізація

бази даних не допоможе, якщо використані оператори SQL, елементи яких йдуть в невірному порядку.

Проведено дослідження часу виконання програмних конструкцій у наступних випадках:

- 1) Порядок використання таблиць у виразі FROM;
- 2) Використання умови обмеження у виразі WHERE;
- 3) Послідовність розташування умови обмежування та умови зв'язування у виразі WHERE.

Для відстеження часу виконання запиту використовувалась функція TIME();

Результат виконання першої оптимальної програмної конструкції представлено на рисунку 2.36.

ПІБ_автора	ORCID	Посада_автора	Ступінь	Місце_роботи	Електронна_пошта	Назва_роботи	Рік_видення	Вид_роботи
Косолапов Анатолій Аркадійович	0000-0001-8878-568X	Професор	Доктор технічних наук	УДУНТ	*****	Логіко-лінгвістична система управління сповільнювачем прицільної гальмівної позиції на сортувальній гірці	2012	Стаття
Косолапов Анатолій Аркадійович	0000-0001-8878-568X	Професор	Доктор технічних наук	УДУНТ	*****	Методика оцінки характеристик систем керування реального часу в процесі проектування	2015	Стаття
Косолапов Анатолій Аркадійович	0000-0001-8878-568X	Професор	Доктор технічних наук	УДУНТ	*****	Науковий підхід до створення інтелектуальних транспортних систем: теорія і практика	2014	Стаття
Косолапов Анатолій Аркадійович	0000-0001-8878-568X	Професор	Доктор технічних наук	УДУНТ	*****	ф-транзакція як основна модель для оцінки інформаційно-часових характеристик сервіс-орієнтованих комп'ютерних систем	2014	Стаття

Рисунок 2.36 - Результат виконання першої оптимальної програмної конструкції

Результат виконання другої оптимальної програмної конструкції представлено на рисунку 2.37.

Посада_автора	ПІБ_автора	ORCID	Електронна_пошта	Місце_роботи
Старший викладач	Дзюба Володимир Володимирович	-	*****	УДУНТ

Рисунок 2.37 - Результат виконання другої оптимальної програмної конструкції

Результат виконання третьої оптимальної програмної конструкції представлено на рисунку 2.38.

Посада_автора	ПІБ_автора
Старший викладач	Дзюба Володимир Володимирович
*	

Рисунок 2.38 - Результат виконання третьої оптимальної програмної конструкції

Результати часу виконання програмних конструкцій наведено у таблиці 2.1.

Таблиця 2.1 - Результати часу виконання програмних конструкцій

Програмна конструкція	Час виконання програмної конструкції, с.	Оптимальна програмна конструкція	Час виконання оптимальної програмної конструкції, с.
SELECT* FROM БД_НР_ЕОМ, НАУКОВА_РОЗРОБКА, АВТОР, ВИДАННЯ, ПОСАДА;	2	SELECT* FROM ПОСАДА, ВИДАННЯ, АВТОР, НАУКОВА_РОЗРОБКА, БД_НР_ЕОМ;	1
SELECT Посада_автора, ПІБ_автора, ORCID, Електронна_пошта, Місце_роботи FROM АВТОР WHERE ПІБ_автора = 'Дзюба Володимир Володимирович';	2	SELECT Посада_автора, ПІБ_автора, ORCID, Електронна_пошта, Місце_роботи FROM АВТОР WHERE Посада_автора = 'Старший викладач';	1
SELECT Посада_автора, ПІБ_автора FROM АВТОР WHERE ПІБ_автора = 'Дзюба Володимир Володимирович' AND Посада_автора = 'Старший викладач';	2	SELECT Посада_автора, ПІБ_автора FROM АВТОР WHERE Посада_автора = 'Старший викладач' AND ПІБ_автора = 'Дзюба Володимир Володимирович';	1

З таблиці видно, що при виконанні оптимальних конструкцій у наступних випадках: розташування невеликих таблиць спочатку переліку у виразі FROM; використання найбільш обмеженої умови у виразі WHERE; розташування найбільш обмеженої умови перед умовою зв'язування у виразі WHERE, час виконання конструкцій зменшується в два рази.

Оскільки створена база даних БД\_НР\_ЕОМ зберігається на локальному жорсткому диску, а не в мережі, тому обрані наступні механізми підвищення

ефективності роботи: відключення autocorrect імен, налаштування автоматичного стиску та відновлення бази даних, відключення параметрів autocorrect.

1. Функція autocorrect імен забезпечує працездатність об'єктів бази даних при перейменуванні інших об'єктів від яких вона залежить.

Оскільки створена база даних є стабільною, а її об'єкти не будуть перейменовані, то можна відключити цю функцію, задля підвищення ефективності роботи бази даних.

Для цього необхідно: відкрити базу даних, на вкладці «Файл» обрати пункт «Параметри», далі обрати «Параметри Access», обрати категорію «Поточна база даних» в пункті «Параметри автозавіря імен» необхідно прибрати всі прапорці[17].

Налаштування функції autocorrect імен представлено на рисунку 2.39.

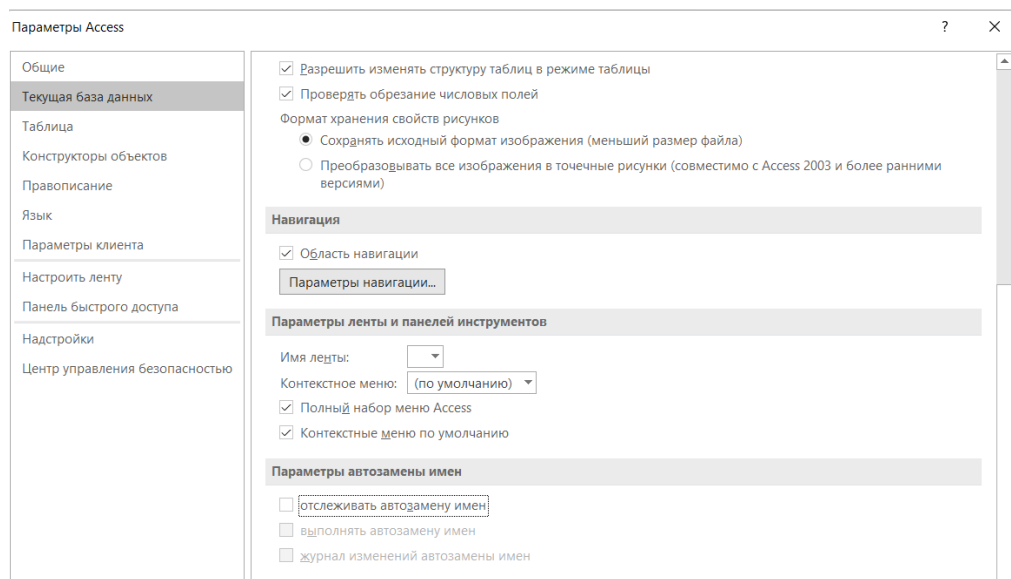


Рисунок 2.39 - Налаштування функції autocorrect імен

2. З часом продуктивність бази даних може сповільнюватись через те, що залишається виділене місце для видалених або тимчасових об'єктів. Команда «Стискання та відновлення» видаляє це місце та допомагає швидше та ефективніше працювати з базою даних.

Для налаштування команди необхідно: відкрити базу даних, на вкладці «Файл» обрати пункт «Параметри» в діалоговому вікні «Параметри Access»

обрати категорію «Поточна база даних» та обрати в меню «Параметри додатків» параметр «Стискання при закритті» [17].

Налаштування команди «Стискання та відновлення» представлено на рисунку 2.40.

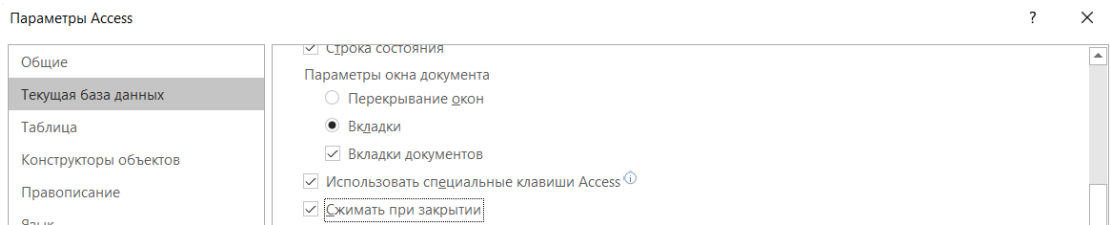


Рисунок 2.40 - Налаштування команди «Стискання та відновлення»

3. Параметр autocorrect за замовчуванням виправляє орфографію по мірі її написання, щоб підвищити ефективність роботи бази даних потрібно вимкнути її.

Для цього необхідно: відкрити базу даних, на вкладці «Файл» обрати пункт «Параметри», в діалоговому вікні параметрів обрати функцію «Перевірка», натиснути на параметр «Параметри autocorrect» та прибрати прапорці.

Чим більше параметрів функції autocorrect відключено, тим вища продуктивність роботи бази даних[17].

Налаштування параметрів функції autocorrect представлено на рисунку 2.41.

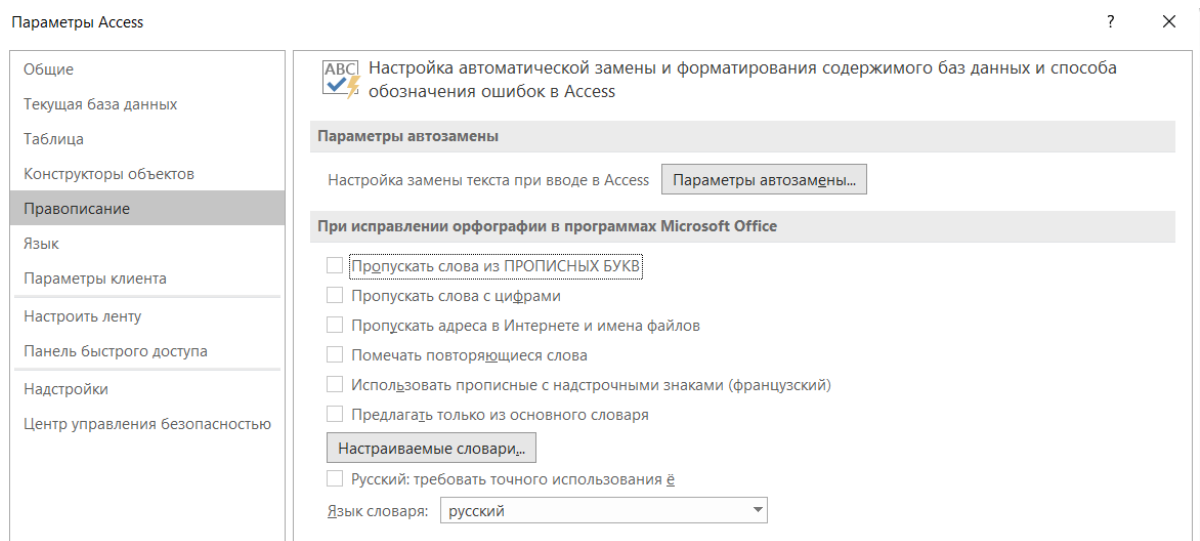


Рисунок 2.41 - Налаштування параметрів функції autocorrect

На основі методів підвищення ефективності роботи бази даних в Microsoft Access було створено лабораторну роботу, яка представлена у Додатку В.

## **2.6 Основні висновки**

У другому розділі кваліфікаційної роботи розглянуті питання проектування, створення та підвищення ефективності роботи захищеної бази даних.

1. Результатом проектування стала база даних за методом «Сутність-Зв'язок». Виділено наступні сутності: АВТОР, РОЗРОБКА, ВИДАННЯ, ПОСАДА та зв'язки між ними - АВТОР ПИШЕ РОЗРОБКУ, АВТОР ЗАЙМАЄ ПОСАДУ, РОЗРОБКА ДРУКУЄТЬСЯ ВИДАННЯМ. У результаті проектування отримані відношення R1, R4, R5, R6 та R7, які співпали з результатами проектування за методом нормальних форм, що свідчить про правильність проектування. Тому спроектована база даних готова до програмного створення.

2. У результаті створення бази даних у програмі Microsoft Access з використанням мови SQL створено п'ять таблиць відношень, а саме: АВТОР, НАУКОВА\_РОЗРОБКА, БД\_НР\_ЕОМ, ПОСАДА, ВИДАННЯ, а також складена схема зв'язку між таблицями.

3. Після програмної реалізації бази даних розглянуті наступні організації захисту бази даних: парольний захист, відображення та приховування об'єктів у вікні бази даних, шифрування й захист на рівні користувачів.

4. Проведено дослідження часу виконання програмних конструкцій у наступних випадках: порядок використання таблиць у виразі FROM; використання умови обмеження у виразі WHERE; послідовність розташування умови обмежування та умови зв'язування у виразі WHERE. Визначено, що час виконання зменшується у два рази при запису відповідних оптимальних конструкцій. Крім того, підвищення ефективності роботи бази даних відбувається при використанні наступних механізмів: відключення autocorrect імен, налаштування автоматичного стиску та відновлення бази даних, відключення параметрів autocorrect.

## ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

У даній кваліфікаційній роботі реалізовано створення та підвищення ефективності роботи захищеної бази даних наукових розробок кафедри ЕОМ.

1. Виконаний огляд та аналіз предмета проектування. Серед існуючих додатків обрано Microsoft Access у зв'язку з тим, що більшість реляційних баз даних, окрім Access, складається з двох окремих компонентів – «back-end», де зберігаються дані та «front-end» - інтерфейс користувача для взаємодії з даними. Такий тип конструкції досить складний, Microsoft Access має більш зрозумілий інтерфейс для будь-якого користувача.

2. Спроектовано базу даних наукових розробок за методами: нормальних форм та «Сутність – Зв'язок». На основі даних репозитарія УДУНТ та даних про науково-педагогічний склад кафедри ЕОМ. Результати проектування за різними методами співпали, що свідчить про правильність проектування.

3. Створено у програмі Microsoft Access з використанням SQL базу даних наукових розробок кафедри ЕОМ, яка містить наступні таблиці: БД\_НР\_ЕОМ, НАУКОВА\_РОЗРОБКА, АВТОР, ВИДАННЯ та ПОСАДА.

4. Виконано захист створеної бази даних з використанням наступних механізмів: парольний захист, відображення та приховування об'єктів у вікні бази даних, шифрування й захист на рівні користувачів.

5. Проведено дослідження часу виконання програмних конструкцій у наступних випадках: порядок використання таблиць у виразі FROM; використання умови обмеження у виразі WHERE; послідовність розташування умови обмежування та умови зв'язування у виразі WHERE. Визначено, що час виконання зменшується у два рази при запису відповідних оптимальних конструкцій.

6. Запропоновано нову лабораторну роботу з дисципліни «Бази даних» здобувачам першого (бакалаврського) рівня спеціальності «Кібербезпека».

## ПЕРЕЛІК ПОСИЛАНЬ

1. Моделі баз даних – шпаргалка для початківців [Електронний ресурс]. – Режим доступу: <https://www.internet-technologies.ru/articles/modeli-baz-dannyh-sistemy-upravleniya-bazami-dannyh.html> ;
2. Ієрархічні бази даних [Електронний ресурс]. – Режим доступу: [https://spravochnick.ru/bazy\\_dannyh/ierarhicheskie\\_bazy\\_dannyh/](https://spravochnick.ru/bazy_dannyh/ierarhicheskie_bazy_dannyh/) ;
3. CODASYL Data Model [Електронний ресурс]. – Режим доступу: <https://silo.tips/download/network-codasyl-data-model> ;
4. Моделі бази даних. Типи моделей баз даних [Електронний ресурс]. – Режим доступу: <https://bitserv.ru/uk/modeli-bazy-dannyh-tipy-modelei-baz-dannyh/> ;
5. Автоматизоване проектування силових трансформаторів – банки даних [Електронний ресурс]. – Режим доступу: <https://leg.co.ua/arhiv/podstancii/avtomatizirovannoe-proektirovanie-silovyh-transformatorov-20.html> ;
6. Дистанційний курс з дисципліни «Бази даних», укладач доцент Пахомова В.М. [Електронний ресурс]. – Режим доступу: <https://lider.diit.edu.ua/mod/lesson/view.php?id=13730> ;
7. Організаційно-інформаційна система управління [Електронний ресурс]. – Режим доступу: <http://diit.edu.ua/university/structure/subdivisions/ioc/robota-z-prohramoyu-asu-kursor> ;
8. Дистанційний курс з дисципліни «Бази даних», укладач доцент Оскаренко В.С. [Електронний ресурс]. – Режим доступу: [http://bseu.by/it/tohod/lekcii4\\_3.htm](http://bseu.by/it/tohod/lekcii4_3.htm) ;
9. Десять найкращих інструментів для розробки та адміністрування MySQL [Електронний ресурс]. – Режим доступу: <https://habr.com/ru/post/142385/> ;
10. MySQL Workbench [Електронний ресурс]. – Режим доступу: <https://www.mysql.com/de/products/workbench/> ;
11. Navicat [Електронний ресурс]. – Режим доступу: <https://www.capterra.com.de/software/154292/navicat-premium> ;

12. HeidiSQL [Електронний ресурс]. – Режим доступу: <https://www.embarcadero.com/ru/case-study/heidisql-case-study> ;
13. MS Access : основні можливості, переваги та недоліки [Електронний ресурс]. – Режим доступу: [https://life-prog.ru/1\\_27156\\_tema-subd-ms-access-osnovnievozmozhnosti-dostoinstva-i-nedostatki.html](https://life-prog.ru/1_27156_tema-subd-ms-access-osnovnievozmozhnosti-dostoinstva-i-nedostatki.html) ;
14. Was sind Microsoft Access Project Dateien [Електронний ресурс]. – Режим доступу: <https://www.exefiles.com/de/extensions/file-types/microsoft-access-project> .
15. Безпека в Microsoft Access [Електронний ресурс]. – Режим доступу: <https://support.microsoft.com/ru-ru/office/безопасность-в-access-2010-cae6d764-0318-4622-955f-68d9f186d6ca> .
16. Безпека на рівні користувача в Microsoft Access [Електронний ресурс]. – Режим доступу: <https://support.microsoft.com/ru-ru/office/что-произошло-с-безопасностью-на-уровне-пользователя-69b362cd-503f-4e8a-a735-fb923ffb9fa3>.
17. Ефективність роботи бази даних в Microsoft Access [Електронний ресурс]. – Режим доступу: <https://support.microsoft.com/ru-ru/office/как-ускорить-работу-access-f6827763-bb5c-4f48-8457-7a14addab6be#bm3>.
18. Хрестян А.В., керівник Пахомова В.Н., Підвищення ефективності роботи захищеної бази даних наукових розробок кафедри обчислювальних машин, Молода академія 2022. Збірка тез доповідей. Всеукраїнська науково-технічна конференція студентів і молодих учених, Дніпро: УДУНТ, 2022, с. 146.