

Міністерство освіти і науки України

Національний університет водного господарства
та природокористування

Навчально-науковий інститут агроєкології та землеустрою

Кафедра геодезії та картографії

05-04-158М

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних і самостійних робіт з навчальної
дисципліни **«Проектування та управління базами
геопросторових даних» (частина 1)** для здобувачів вищої
освіти першого (бакалаврського) рівня за освітньо-професійною
програмою «Геодезія та землеустрій» спеціальності
193 «Геодезія та землеустрій» усіх форм навчання

Рекомендовано науково-методичною
радою з якості ННІАЗ
Протокол №13 від 18 лютого 2025 р.

Рівне – 2025

Методичні вказівки до виконання лабораторних і самостійних робіт з навчальної дисципліни «Проектування та управління базами геопросторових даних» (частина 1) для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Геодезія та землеустрій» спеціальності 193 «Геодезія та землеустрій» усіх форм навчання. [Електронне видання] / Янчук О. Є., Прокопчук А. В. – Рівне : НУВГП, 2025. – 53 с.

Укладачі:

Янчук О. Є., канд. техн. наук, доцент кафедри геодезії та картографії;

Прокопчук А. В., ст.викл. кафедри геодезії та картографії.

Відповідальний за випуск:

Янчук Р. М., к.т.н., доцент, завідувач кафедри геодезії та картографії.

Керівник групи забезпечення спеціальності:

Янчук Р. М., к.т.н., доцент, завідувач кафедри геодезії та картографії.

Попередня версія методичних вказівок 05-04-38

© О. Є. Янчук,
А. В. Прокопчук, 2025
© Національний університет
водного господарства та
природокористування, 2025

ЗМІСТ

ПЕРЕДМОВА	4
ЛАБОРАТОРНІ РОБОТИ.....	5
Лабораторна робота № 1 Знайомство з Microsoft Access. Створення таблиць. Налаштування полів	5
Лабораторна робота № 2 Створення та налаштування форм MS Access. Автоматизація роботи бази даних	20
Лабораторна робота № 3 Запити та звіти	37
Лабораторна робота № 4 Розробка схеми бази даних у DbDesigner.....	48
РЕКОМЕНДОВАНА ЛІТЕРАТУРА ТА РЕСУРСИ.....	53

ПЕРЕДМОВА

Методичні вказівки складено відповідно до програми навчальної дисципліни «Проектування та управління базами геопросторових даних» та призначено для здобувачів вищої освіти першого (бакалаврського) рівня спеціальності 193 «Геодезія та землеустрій» усіх форм навчання.

Головним завданням курсу є навчання студентів навичкам проектування, організації функціонування і використання баз даних геоінформаційних систем, а також впровадження таких систем в виробництво і органи державного управління та у всі можливі галузі народного господарства.

У даній частині методичних вказівок викладені основні положення, порядок виконання та розробки бази даних у програмному середовищі Microsoft Office Access.

У результаті виконання даних лабораторних робіт студенти повинні навчитися виконувати інфологічне проектування структури бази даних та працювати в пакеті Microsoft Office в середовищі Microsoft Access.

ЛАБОРАТОРНІ РОБОТИ

Лабораторна робота № 1

Знайомство з Microsoft Access. Створення таблиць.

Налаштування полів

Мета: навчитись створювати таблиці у Microsoft Access та налаштовувати властивості полів.

Завдання: запроектувати структуру бази даних (мінімум з трьох таблиць) для опису заданої предметної області та реалізувати її у Microsoft Access.

В якості вихідних даних студенти обирають із запропонованого нижче переліку предметну область. Самостійно шукають інформацію на обрану тематику та проєктують структуру майбутньої бази даних. База даних має містити мінімум три таблиці та обов'язково містити інформацію про місцезположення об'єктів для подальшого геокодування.

Орієнтовна тематика:

1. Створення бази даних для забезпечення діяльності установи, організації, підприємства.
2. Створення бази даних обслуговуючої інфраструктури адміністративно-територіальної одиниці (готелі, ресторани, магазини, автозаправні станції тощо).
3. Створення бази даних системи водо-, газо-, електропостачання адміністративно-територіальної одиниці.
4. Створення бази даних кадастру земельних або інших природних ресурсів.
5. Створення бази даних пам'яток архітектури.
6. Створення бази даних туристичних маршрутів.
7. Створення бази даних закладів освіти адміністративно-територіальної одиниці.
8. Створення бази даних закладів охорони здоров'я адміністративно-територіальної одиниці.
9. Створення бази даних спортивних закладів адміністративно-територіальної одиниці.
10. Створення бази даних обліку залізничних перевезень.
11. Створення бази даних для обліку поштових відправлень.

1) Проектування таблиць бази даних

Розглянемо основні визначення, які використовуються при створенні баз даних.

Предметна область – може стосуватись будь-якого типу організації (наприклад, банк, університет, лікарня, завод тощо).

Необхідно розрізняти повну предметну область (велике підприємство з виробництва автомобілів, ЕОМ, хімічної продукції або управління земельних ресурсів) та організаційну одиницю цієї предметної області. Організаційна одиниця може представляти свою предметну область (наприклад, цех з виробництва процесорів або відділ реєстрації землекористувачів).

Для предметних областей потрібна інформація про персонал, заробітну плату, товар, землекористувачів, сервітути, історії хвороб, тобто відомості про людей, місця, предмети, події та поняття.

Об'єктом може бути людина, предмет, явище, місце, подія або поняття про які записані дані.

Тип об'єкта – набір однорідних предметів, явищ або ідей, які виступають як ціле. Екземпляр об'єкта стосується конкретного об'єкта, наприклад, об'єкт МІСТО, а екземпляр – Київ, Львів і т.д.

Кожен об'єкт характеризується низкою основних атрибутів. Наприклад, будинок характеризується габаритами, кольором, часом експлуатації, розміром присадибної ділянки. Земельна ділянка – кадастровим номером, адресою, формою власності, власником, розміром, призначенням і т.д. Атрибут часто називають **елементом даних**.

Значення даних представляють дані, які знаходяться в кожному елементі даних.

Множина **кортежів (рядочків)** утворює таблицю.

Правильне проектування будь-якої бази даних починається на листку паперу. Не дивлячись на можливості автоматизованого проектування, деякі задачі легше вирішити на папері.

1. Оскільки причиною створення бази даних є отримання конкретних кінцевих результатів, то необхідно коротко описати

звіти, списки або інші документи, які передбачається отримувати за допомогою системи. Не потрібно відразу створювати ідеальну та повну базу даних. На початку необхідно включити до неї все необхідне, тому що потім завжди можна повернутись до бази та доповнити її.

2. На іншому листку паперу накидати декілька варіантів звітів, списків та інших документів, які перераховані в п. 1. На цьому етапі головним є розуміння – яку інформацію потрібно отримати вкінці. Необхідно передбачити, який вигляд матимуть найнеобхідніші звіти, та помітити з якими даними вони працюватимуть.

3. Для кожного поля у списку необхідно вибрати ім'я та тип, визначити розмір (кількість символів) текстових полів.

Вибираються поля, які можна розмістити в одній таблиці, наприклад, ім'я, адреса, ідентифікаційний код, паспортні дані, телефон фізичної особи – у таблицю **Фізична особа**; кадастровий номер ділянки, площа, цільове призначення, адреса, форма власності – у таблицю **Земельна ділянка**.

Деякі елементи даних мають важливі властивості. Володіючи значенням, яке може набувати такий елемент даних об'єкта, ми можемо ідентифікувати значення, яке набувають інші елементи даних цього ж об'єкта.

Наприклад, знаючи кадастровий номер земельної ділянки "1234567890123456789", ми можемо встановити, що ця земельна ділянка розміщена на вул. В. Чорновола, 2 та має власника – ЗАТ "Бедрик".

Ключ – мінімальний набір атрибутів, за значеннями яких можна однозначно знайти необхідний елемент даних.

Мінімальність означає, що виключення із набору будь-якого атрибуту не дозволяє однозначно ідентифікувати об'єкт за тими що залишилися.

Розрізняють **первинні** й **зовнішні** ключі.

Первинний ключ (ключове поле) – поле, що однозначно ідентифікує запис.

Зовнішній ключ – поле підпорядкованої таблиці, необхідне для зв'язку між таблицями.

1. Головним критерієм вибору первинного ключа є його унікальність. Значення, які зберігаються в такому полі не повинні повторюватись! Під час проектування таблиці потрібно знайти унікальні поля. Такими полями можуть бути – кадастровий номер, ідентифікаційний номер фізичної особи, номер автомобіля або будь-яке інше поле, яке містить дані, що не зустрічаються більше як в одному записі.

2. Якщо унікального поля в запроєктованій таблиці немає, тоді в таблицю рекомендується ввести поле типу **Лічильник**.

3. Таблиця може мати тільки один унікальний ідентифікатор - **первинний ключ (primary key)** і декілька **вторинних (зовнішніх) ключів (foreign key)**, що посилаються на первинні ключі інших таблиць.

4. Якщо ви створюєте таблицю без первинного ключа, Access нагадає про це і запитає про необхідність його створення. Якщо відповідь буде стверджувальною, то програма створить поле **ID**, з типом даних **Лічильник**.

У реляційних таблицях дані можуть бути розташовані в двох і більше таблицях. В Access для зв'язування таблиць використовується зв'язуюче поле, яке називається **зовнішнім ключем**. Наприклад, в одній таблиці розміщена інформація про громадян, їх адреси, ідентифікаційний код, а в інших – інформація про земельні ділянки, які їм належать або про нерухомість, яка їм належить. Інформація про належність нерухомості громадянину зв'язується з його даними завдяки зв'язуючому полю, яким в даному випадку може бути ідентифікаційний код.

При створенні зв'язків між таблицями необхідно дотримуватись таких правил:

1. Як правило, зв'язуючим полем являється первинний ключ однієї таблиці, тоді як в іншій таблиці це поле являється звичайним.

2. Зв'язуючі поля повинні містити дані одного типу!
Зв'язуються тільки однотипні поля!

Коли зв'язуються між собою дві таблиці, утворюється один із чотирьох можливих зв'язків:

1. **Один до одного.** За такого зв'язку один запис першої таблиці зв'язується тільки з одним записом іншої таблиці.

Такий зв'язок є найпростішим, проте зустрічається рідко. Таблиці, які мають зв'язок один до одного, завжди можна зкомпонувати в одну, що зазвичай і виконується.

2. **Один до багатьох.** Один запис першої таблиці зв'язаний з багатьма записами в іншій.

Наприклад, громадянин володіє трьома земельними ділянками на праві приватної власності, тому, один запис про громадянина зв'язується з трьома записами в таблиці про земельні ділянки.

3. **Багато до одного.** Багато записів першої таблиці зв'язані з одним записом другої таблиці.

Наприклад, два громадянина володіють земельною ділянкою на праві спільної власності, тому, два записи про громадян зв'язуються з одним записом в таблиці про земельні ділянки.

4. **Багато до багатьох.** У такому випадку багато записів однієї таблиці зв'язані з іншими даними в іншій. Фактично такий тип зв'язку – складний і заплутаний, у базі даних Access його створювати заборонено. На практиці для реалізації такого зв'язку використовуються **зв'язуючі таблиці**.

Наприклад, існує дві таблиці, одна з яких називається **Земельна ділянка**, а інша – **Фізична особа**. Таблиці **Фізична особа** може відповідати декілька записів із таблиці **Земельна ділянка**, оскільки громадянин може володіти багатьма земельними ділянками одночасно. Таблиці **Земельна ділянка** може відповідати багато записів із таблиці **Фізична особа**, оскільки земельна ділянка може належати декільком громадянам на праві спільної власності. Наведений приклад являє собою класичний приклад зв'язку **багато до багатьох**.

Для створення такого типу зв'язку необхідно створити **зв'язуючу таблицю**, яка буде містити первинні ключі першої та другої таблиці. Наприклад, первинним ключем таблиці **Земельна ділянка** є поле Кадастровий номер, а первинним

ключем таблиці **Фізична особа** – поле Ідентифікаційний номер. Для створення зв'язку необхідно створити третю таблицю, звичайними (не ключовими) полями в якій будуть Ідентифікаційний номер та Кадастровий номер.

2) Створення таблиць у MS Access


Під час першого запуску Access відображається сторінка **Початок роботи з Microsoft Office Access** (рис.1.1).



Рис. 1.1. Вікно Початок роботи з Microsoft Office Access


На цій сторінці можна розпочати створення нової бази даних, відкрити наявну базу даних або переглянути вміст веб-сайту Microsoft Office Online.

Створення пустої бази даних

На сторінці **Початок роботи з Microsoft Office Access** у розділі **Нова пуста база даних** потрібно обрати пункт **Пуста база даних**. Далі, праворуч, в області **Пуста база даних** ввести ім'я файлу в полі **Ім'я файлу**. Якщо не вказати розширення файлу, додаток Access додасть його автоматично. Щоб змінити розташування файлу, вказане за замовчуванням, необхідно вказати елемент **Пошук розташування для бази даних** 

(поруч із полем **Ім'я файлу**), перейти до нового розташування та натиснути кнопку **ОК**.

Натиснути кнопку **Створити**. Додаток Access створює базу даних із пустою таблицею з іменем «Таблиця1», а потім відкриває цю таблицю в поданні таблиці. Курсор розташовано в першій пустій клітинці у стовпці **Додати нове поле**.

Якщо наразі не потрібно вводити дані в «Таблиці1», то потрібно натиснути кнопку **Закрити** . Якщо до таблиці внесено будь-які зміни, буде запропоновано зберегти їх. Необхідно обрати кнопку **Так**, щоб зберегти зміни або кнопку **Ні**, щоб скасувати їх або натиснути кнопку **Скасувати**, щоб залишити таблицю відкритою.

Слід звернути увагу – якщо закрити таблицю без збереження, її буде видалено, навіть якщо вона містить дані.

Вигляд вікна програми

На рис. 1.2 відображено загальний вигляд інтерфейсу програми Microsoft Access з відкритою базою даних.

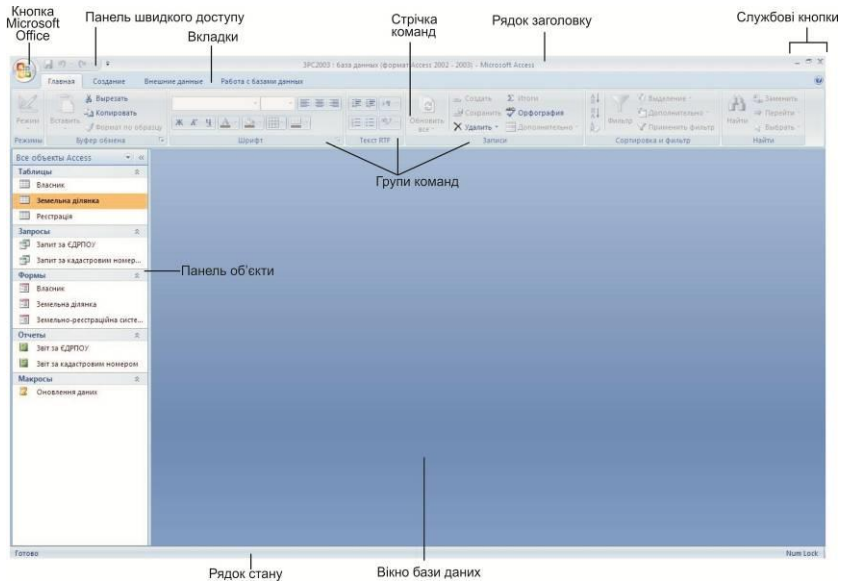


Рис. 1.2. Вікно Microsoft Access

Під панельлю швидкого доступу розташована стрічка для швидкого пошуку потрібних у роботі команд, яка організована у логічні групи за вкладками. Така організація дозволяє отримати доступ до функцій, які найчастіше використовуються, за допомогою клацка миші.

Вигляд вікна таблиці

В створеній базі даних дані зберігаються в таблицях – тематичних списках рядків і стовпців. Таблиця містить дані про певну область. Кожен запис у таблиці містить дані про один елемент. Запис складається з полів. Запис також зазвичай називається рядком, а поле — стовпцем.

База даних може містити багато таблиць, кожна з яких містить дані про окрему область. Кожна таблиця може містити багато полів різних типів, зокрема текстові поля, числові поля, поля дат і зображення.

У верхній частині розташоване ім'я таблиці (рис. 1.3). Відразу під ним вказані імена полів, а в рядочках розміщено записи таблиці. У нижній частині вікна знаходяться кнопки переходу між записами.

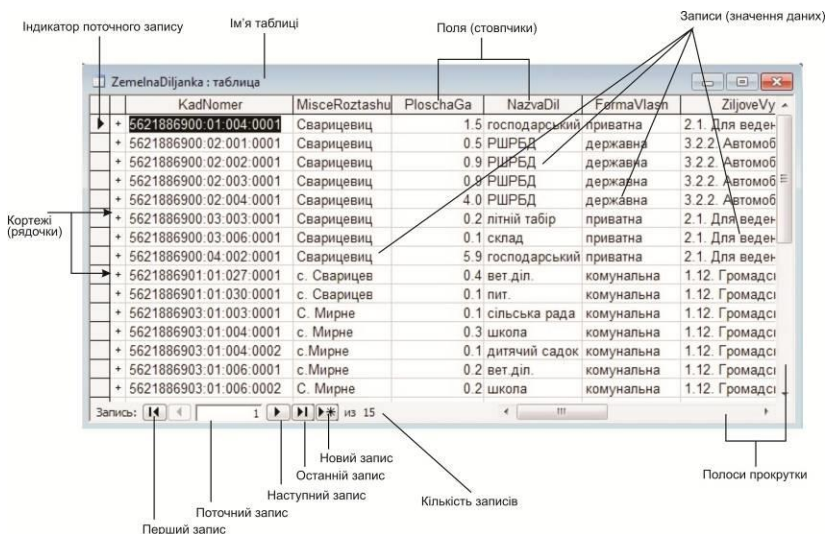


Рис. 1.3. Вікно таблиці у Microsoft Access

Порядок створення нової таблиці у базі даних

У відкритій базі даних, на вкладці **Створити** у групі **Таблиці** клацніть по елементу **Таблиця** (рис. 1.4), внаслідок чого нову таблицю буде додано до бази даних та відкрито в режимі таблиці.

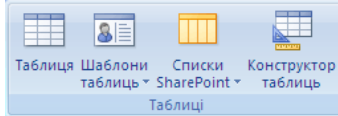


Рис. 1.4. Елемент Таблиця групи Таблиці на вкладці Створити

Під час створення таблиць у Access, рекомендовано користуватись **Конструктором таблиць**, а не **Майстром таблиць**, оскільки він передбачає далеко не всі можливості та функції, які нам необхідні.

Створюючи таблиці у режимі конструктора ви можете контролювати весь процес.

Для створення таблиці у режимі конструктора необхідно виконати наступне:

1. На вкладці **Створити** у групі команд **Таблиці** вибрати команду **Конструктор таблиць** (рис. 1.4). Access виведе на екран пусту форму для проектування таблиці, аналогічну, зображеній на рис. 1.5.

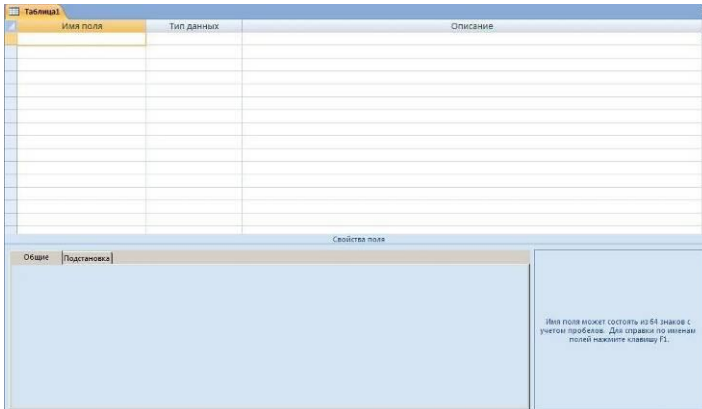


Рис. 1.5. Конструктор таблиці

2. Ввести ім'я поля та, щоб перемістити курсор далі, натиснути клавішу <Tab>. Курсор переміститься в стовпчик **Тип даних**. *Рекомендується використовувати для назв полів лише латинські літери, щоб полегшити подальшу інтеграцію з СУБД PostgreSQL.*

3. У випадяючому списку **Тип даних** клацнути на потрібному типі поля і натиснути клавішу <Tab>. Курсор переміститься в стовпчик **Опис**.

4. У полі **Опис** ввести чітке, зрозуміле пояснення змісту поля. Під час роботи з формами для внесення даних, пояснення, введене в поле **Опис**, буде відображатись у рядку стану, що слугуватиме підказкою.

5. Повторювати пп. 3-5 доти, поки всі поля не займуть свої місця.

6. Для збереження нової таблиці на диску виконати команду **Зберегти**.

7. У діалоговому вікні **Збереження** ввести ім'я, яке потрібно присвоїти таблиці, і натиснути клавішу <Enter>.

Access виведе на екран діалогове вікно з повідомленням “Первинний ключ не визначено” та запропонує створити ключове поле. Це повідомлення означає, що таблиця не буде впорядкована автоматично. Оскільки, не рекомендується створювати таблиці без ключових полів, у даному діалоговому вікні слід клацнути на кнопці **Так**.

Для створення поля, яке буде виконувати функції **первинного ключа**, необхідно виконати наступні кроки:

1. Відкрити таблицю в режимі конструктора.
2. Клацнути правою кнопкою миші на рядку з назвою поля, яке вибрано в якості первинного ключа.
3. З'явиться контексне меню. Вибрати **Ключове поле**. Якщо на кнопці маркера з'явиться зображення маленького ключика – це означає, що первинний ключ створено.

Вибір типу даних для поля таблиці

В режимі **Конструктор** для полів таблиці можна обрати типи даних у відповідному випадяючому списку. При виборі типу даних необхідно враховувати наступне:

- які значення повинні відображатися в полі. Не можна, наприклад, зберігати текст у полі, що має числовий тип даних і недоцільно зберігати числові дані в текстовому вигляді;
- скільки місця необхідно для збереження значень у полі;
- які операції повинні проводитися зі значеннями в полі. Наприклад, підсумовувати значення можна в числових полях і в полях, які мають грошовий формат, а в текстових – не можна; чи потрібне сортування або індексування поля;
- яким чином повинні бути відсортовані значення в полі. Числа в текстових полях сортуються як рядкові значення (1, 10, 100, 2, 20, 200 і т.д.), а не як числові значення. Для сортування дат і часу варто використовувати поле типу Дата/Час.

У нижченаведеній таблиці 1.1 описано основні типи даних для полів в MS Access.

Таблиця 1.1

Типи даних у Microsoft Access

Тип даних	Зміст поля	Розмір
Текстовий	Текст, або числа, що не вимагають виконання розрахунків, наприклад, номера телефонів	Число символів від заданого розміру до 255 символів
Поле МЕМО	Довгий текст або комбінація тексту та чисел	До 65535 символів
Числовий	Числові дані, що використовуються для виконання розрахунків	1,2,4,8 або 16 байт
Дата/час	Дата і час, які стосуються років з 100 по 9999 включно	8 байт
Грошовий	Грошові значення і числові дані, які використовуються в математичних розрахунках, виконуються з точністю до 15 знаків у цілій та до 4 у дробовій частині	8 байт
Лічильник	Унікальні послідовно збільшувані (на 1) або випадкові числа, автоматично вводяться при створенні нового запису в таблиці	4 байт


Логічний	Логічні значення, а також поля, які можуть містити одне з двох можливих значень (True/False, так/ні)	1 біт
Поле об'єкта OLE	Об'єкт (наприклад, таблиця Excel, документ Word, рисунок, звукозапис або інші дані), зв'язані або упроваджені в таблицю Access	До 1 Гб
Гіпер-посилання	Рядок, який складається з букв та цифр, та являє собою адресу гіперпосилання. Може складатись з трьох частин: текст, адреса, додаткова адреса URL	До 2048 символів

3) Створення зв'язків між таблицями

Створення зв'язків строго формальна процедура, тому їх потрібно створювати після створення таблиць і перед створенням форм, запитів, модулів чи програм.

Послідовність створення зв'язків у Access:

1. На вкладці **Робота з базами даних** у групі команд **Показати або сховати** оберіть команду **Схема даних**. З'явиться вікно **Схема даних**, яке в даний момент швидше за все не буде заповненим.

2. В контекстному меню оберіть команду **Додати таблицю** або клацніть на кнопці **Відобразити таблицю** , яка розміщена на панелі інструментів. На екрані з'явиться діалогове вікно **Додати таблицю** та список таблиць поточного файлу бази даних (рис. 1.6).

3. Послідовно клацніть на таблицях, для яких необхідно створити зв'язок, при цьому так само послідовно натискати кнопку **Додати**.

4. Помістіть курсор миші на поле, з якого необхідно розпочати зв'язок, натисніть та утримуйте ліву клавішу миші. Перемістіть вказівник миші від одного зв'язуючого поля до іншого.

5. Для створення надійного зв'язку між двома таблицями встановить галочку в полі опції **Забезпечення цілісності даних** (в Access неможливо створити зв'язок із неіснуючим об'єктом)

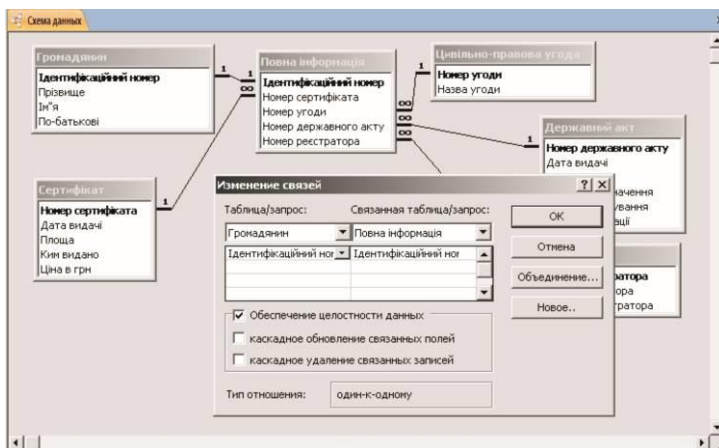


Рис. 1.6. Зв'язування таблиць

4) Створення випадаючих списків та налаштування масок введення

Для зручності введення даних у Access є можливість налаштувати деякі додаткові функції – випадаючі списки, значення за замовчуванням, маски введення тощо.

Випадаючий список – елемент керування на сторінці доступу до даних, клацання на якому призводить до розкриття списку із переліком значень, які може набувати даний елемент даних. Джерелом рядків для випадаючого списку може слугувати набір фіксованих значень, що вводиться при створенні елемента керування або джерело записів (таблиця, запит). Перший випадок зручний, якщо список змінюється рідко, а другий – при частому оновленні списку.

Наприклад, необхідно створити випадаючий список у полі Форма власності. Для створення випадаючого списку в Конструкторі таблиць потрібно обрати відповідне поле та перейти на вкладку **Підстановка** у властивостях поля. Вибрати тип елемента керування – **Список**, тип джерела рядків – **Список**

значень, джерело рядків – ввести необхідний набір значень розділяючи їх ";", тобто – "приватна;комунальна;державна" (рис. 1.7).

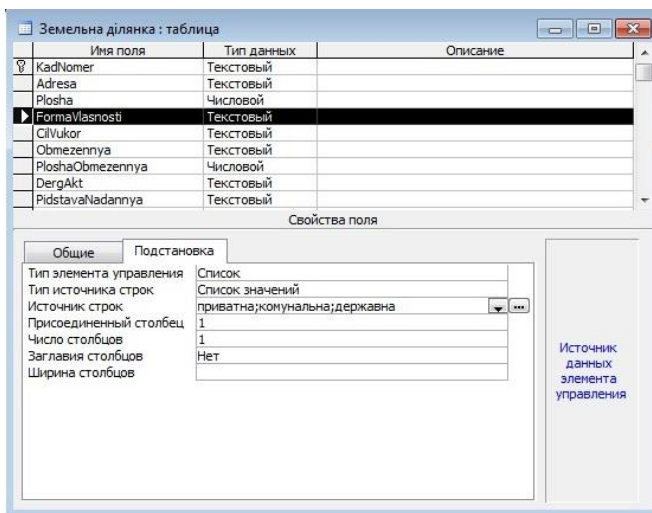


Рис. 1.7. Налаштування випадаючого списку

Якщо якийсь зі значень елемента даних повторюється часто, щоб щоразу не вводити його, можна налаштувати значення за замовчуванням. Для цього обираємо потрібне поле та у властивостях поля на вкладці **Загальні** задаємо потрібне значення у властивості **Значення за замовчуванням**.

Маска введення – елемент керування, який дозволяє контролювати введення даних у таблицю. Маску використовують у випадку, якщо вводяться дані, що мають певну визначену структуру – відома кількість символів у визначених позиціях.

Наприклад, поле Код ЄДРПОУ повинно містити 8 символів, які є обов'язковими для введення, тому для нього можна задати маску "00000000" (символ "0" означає, що в дану позицію обов'язково повинна бути введена цифра).

Для поля Серія, номер паспорта зручною є маска – ">LL000000" (символ ">" означає, що літери праворуч від нього

будуть відображатися у верхньому реєстрі; "L" означає, що в дану позицію обов'язково повинна бути введена літера).

Ще одним цікавим прикладом, є маска для введення номера телефону – "!(999)" "999\-99\-99" (символ "!" означає, що значення у цьому полі будуть виводитися по правому краю поля – це зручно коли, наприклад, міжміський код телефону вводиться не обов'язково або номер телефону може містити або 7 або 6 цифр; символ "\" означає, що наступний після нього символ, буде відображатися в полі як постійний символ, тобто у даному прикладі в полі введення даних будуть відображатися відкрита дужка та дефіси; символ "" означає, що всі значення, які взято в лапки буде виведено у вигляді постійних символів, у даному прикладі це закрита дужка та пробіл після неї; символ "9" – означає, що в дану позицію може бути введена цифра або пробіл, тобто цифра не є обов'язковою).

Також корисними при створенні масок є символи: "?" – означає, що в дану позицію може бути введена літера або пробіл, тобто літера не є обов'язковою; "A" – означає, що в дану позицію обов'язково повинна бути введена літера або цифра; "a" – означає, що в дану позицію може бути введена літера, або цифра, або пробіл, тобто введення даних не є обов'язковим.

Завдання для самостійної роботи. Ознайомтеся з усіма доступними налаштуваннями у **Властивостях поля** на вкладках **Загальні** та **Підстановка**.

Запитання для контролю:

1. Як створити нову базу даних у MS Access?
2. Як створити таблицю у режимі конструктора?
3. Які типи даних доступні для полів у MS Access?
4. Що таке предметна область?
5. Що таке первинний ключ?
6. Який ключ обрати у випадку, якщо у таблиці немає поля з унікальними значеннями?
7. Як зв'язати між собою дві таблиці?
8. Як налаштувати маску введення для поля?
9. Як налаштувати випадаючий список для значень поля?

Лабораторна робота № 2
Створення та налаштування форм MS Access.
Автоматизація роботи бази даних

Мета: навчитись створювати форми у Microsoft Access та автоматизувати роботу бази даних.

Завдання: створити форми для зручного введення інформації у базу даних, заповнити дані. Створити макроси та налаштувати параметри завантаження бази даних.

1) Способи створення форм

Форма – це аналог вітрини в магазині, яка дає змогу легко переглядати та вибирати потрібні товари. Форми це об'єкти, за допомогою яких користувачі можуть додавати, редагувати або відображати дані, що зберігаються в базі даних Microsoft Access. Якщо очікується, що базою даних користуватиметься кілька користувачів, належним чином спроектовані форми вкрай важливі для ефективності та точності вводу даних.

Форми зазвичай використовують для наступних цілей:

- для введення нових даних в таблицю, їх редагування і видалення;
- для вибору даних, їх попереднього перегляду і для друку звіту;
- для відкриття інших форм або звітів.

Форми можна створювати двома способами, за допомогою: Конструктора або Майстра (рис. 2.1). Як правило, спочатку форми створюють за допомогою майстра, а потім редагують в конструкторі, тому що на відміну від майстра він має ширший спектр можливостей по форматуванню форм і надання їм бажаного розробником зовнішнього вигляду.

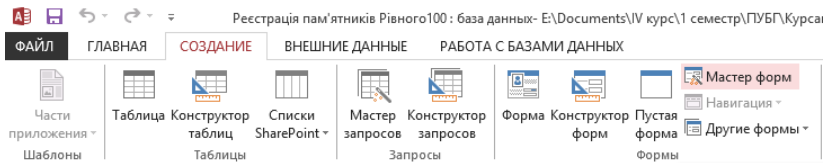


Рис. 2.1. Закладка Створити у групі Форми

Щоб створити форму за допомогою Майстра форм потрібно: на вкладці **Створити** у групі **Форми** натиснути кнопку **Майстер форм**.

Далі, потрібно обрати таблицю і поля з таблиці, за якими буде створено форму.

На останній сторінці майстра натиснути кнопку **Готово**.

У майстрі форм можна отримати різноманітні результати, залежно від вибраних параметрів. Тому рекомендовано запускати майстер кілька разів, щоразу експериментуючи з різними параметрами.

Конструктор форм служить для подальшого налаштування розділів форми. Для того щоб викликати конструктор форм потрібно перейти на закладку **Створити** і в групі **Форми** вибрати **Конструктор форм** (рис. 2.1). У даному випадку з'явиться вікно конструктора форм. Готову форму можна відкрити в режимі конструктора натиснувши на потрібну форму правою кнопкою миші та обрати потрібну дію (рис. 2.2).

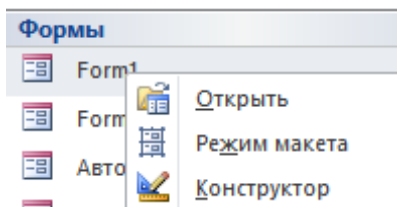


Рис. 2.2. Відкриття форми в режимі Конструктора

У режимі конструктора структура форми відображається більш деталізовано (рис. 2.3).

Вона розділена на три окремі розділи: заголовок форми, область даних і примітка форми. На рисунку заповнена тільки область даних, це пов'язано з тим, що майстер форм створює тільки основний розділ, а допоміжні залишає порожніми.

Заголовок форми може містити елементи, необхідні для зовнішнього оформлення або просто для прикраси форми. Вони будуть присутніми на формі завжди, незалежно від того, створюється новий запис або редагується вже існуючий. Це може бути напис, малюнок, логотип фірми або інший графічний елемент.

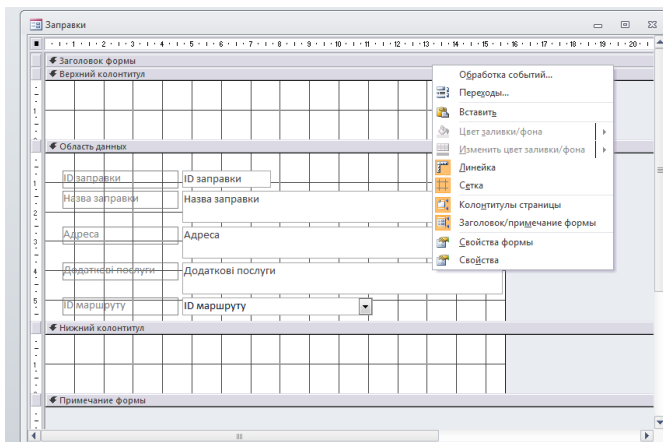


Рис. 2.3. Структура форми в режиме конструктора

Область даних - це основний розділ будь-якої форми. Без нього немає форми, оскільки він містить елементи управління. Тут здійснюються усі динамічні процеси, що відбуваються при роботі з формою. У нашому випадку там будуть тільки два типи елементів управління - пов'язане поле і приєднаний напис.

У «пов'язане поле» здійснюється введення даних, які синхронно поступають в однойменне поле відповідної таблиці, на основі якої створена форма. Поле форми і поле таблиці пов'язані між собою.

Приєднаний напис – це статичний елемент, що містить текстове пояснення. Текст напису зараз співпадає з назвою пов'язаного поля, але це не обов'язково і легко може бути змінено.

Примітка форми – схожа за своїм призначенням на заголовок, але знаходиться в нижній частині форми. Дуже часто містить інструкцію по заповненню форми або пам'ятку з вказівкою необхідної довідкової інформації.

Фоновий малюнок, що знаходиться під елементами управління і розграфлений сіткою, показує реальні розміри робочої області форми – ті, в яких вона виводиться на екран. При наведенні курсора на межі областей, покажчик міняє свою форму і будь-яка з меж у цей момент може бути переміщена

шляхом перетягування. Розташування елементів теж можна змінити, обравши елемент і не відпускаючи кнопку миші перетягнувши його на нове місце. Приєднаний напис при цьому теж автоматично переміщуватиметься.

Окрім вказаних розділів у форми є верхній і нижній колонтитули. Зазвичай вони порожні і не відображаються. Вивести їх на екран, можна встановивши покажчик на лінії з назвою розділу і клацнувши правою кнопкою миші. У додатковому контекстному меню оберіть пункт **Колонтитули**. Аналогічно виконується їх приховування.

Для додавання полів з основної таблиці або запиту до макета форми, можна використати область **Список полів**. Щоб відобразити область **Список полів**, потрібно на вкладці **Конструктор** у групі **Знаряддя** натиснути кнопку **Додати наявні поля** або сполучення клавіш **ALT+F8**. Потім перетягнути поля безпосередньо з області **Список полів** до форми.

Для додавання одного поля потрібно клацнути його двічі або перетягнути з області **Список полів** до розділу форми, у якому воно має відображатися. Для одночасного додавання кількох полів – вибрати потрібні поля, утримуючи натиснутою клавішу **CTRL**. Потім перетягнути вибрані поля до форми.

2) Панель елементів Керування

При відкритті форм в режимі конструктора потрібно перейти на закладку конструктор, на котрій міститься панель елементів (рис. 2.4).

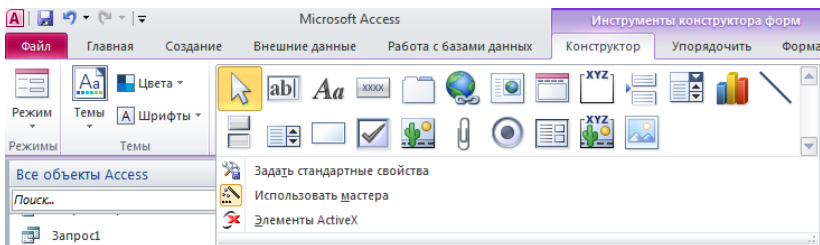


Рис. 2.4. Панель елементів керування форм













За допомогою кнопок панелі елементів керування до форми можна додавати різні об'єкти, типи яких перераховані в табл. 2.1. Елементи керування форм і звітів подібні між собою, тому така ж панель елементів є й у конструкторі звітів.

Таблиця 2.1

Елементи керування форм і звітів

Значок	Назва	Призначення
	Вибір об'єктів	Використовується для вибору, переміщення, зміни розмірів об'єктів
	Майстер	Вмикання й вимикання Майстра елементів керування
	Напис (Label)	Додання напису. Написи створюються автоматично разом з елементами типу текстове поле, поле зі списком і т.п. Вони пояснюють зміст поля введення. Додаткові написи можуть використовуватися для заголовків і пояснень
	Поле (Text Box)	Створення полів (наприклад, розрахункових). Використовується для введення й відображення інформації полів таблиць і запитів, а також для виведення результатів обчислень
	Група перемикачів (Option Group)	Група перемикачів, що посилається на деяке поле таблиці. Кожен перемикач групи відповідає певному цілочисловому значенню цього поля
	Вимикач (Toggle Button)	Дозволяє вводити інформацію типу так/ні. Величині «так» або «істина» відповідає втоплене положення вимикача .
	Перемикач (Option Button)	Елемент групи перемикачів, у якій може бути відзначений тільки один з них (його кружок виглядає «зачорненим») .
	Прапорець (Check Box)	Призначений для подання полів типу так/ні. Відзначений квадратик відповідає величині «так», а порожній — «ні»

продовження таблиці 2.1

	Поле зі списком (Combo Box)	Створення комбінованого списку. Дозволяє як безпосереднє введення значення в поле, так і його вибір в списку визначених значень
	Список (List Box)	Створення звичайного списку. Дозволяє вибирати дані з набору пунктів і не дозволяє безпосереднє введення значень
	Кнопка (Command Button)	Додання кнопки для виконання команд. Клацання на такій кнопці виконує команду, за допомогою якої можна перейти в іншу форму, переміститися по записах тощо
	Малюнок	Вставлення малюнку будь-якого графічного формату з рамкою
	Вільна рамка об'єкта (Unbound Object Frame)	Вставлення рамки, яка містить об'єкт OLE, не зв'язаний ні з яким джерелом даних .
	Приєднана рамка об'єкта (Bound Object Frame)	Вставлення рамки, яка містить об'єкт OLE, пов'язаний з полем даних таблиці або запиту
	Розрив сторінки (Page Break)	Вставлення розриву сторінки
	Набір вкладок (Tab Control)	Вставлення форми, що складається з кількох вкладок
	Підлегла форма/звіт (Subform/Subreport)	Вставлення у форму (або звіт) області з іншої форми (або звіту), зв'язаної з головною
	Лінія (Line)	Додавання прямої лінії
	Прямокутник (Rectangle)	Малювання прозорого або непрозорого прямокутника з кольоровою границею будь-якої товщини
	Інші елементи (More Controls)	Відкриття меню з елементами керування, які не ввійшли в панель елементів керування

3) Властивості об'єктів форми та редагування елементів форми

Для перегляду властивостей об'єктів форми необхідно виконати наступні дії: **Конструктор-Сервіс-Сторінка властивостей** (рис. 2.5). У даному вікні властивостей є можливість налаштовувати, змінювати параметри елементів форми.

Вікно «Властивості» містить п'ять вкладок, що включають окремі набори параметрів:

Макет – властивості для управління зовнішнім виглядом вибраного елемента;

Дані – властивості для управління даними, що відображаються вибраним елементом;

Події – властивості для управління кодом, який виконується вибраним елементом;

Інші – властивості, що не відносяться до трьох вказаних класів властивостей;

Все – всі властивості вибраного елемента в одному великому списку.

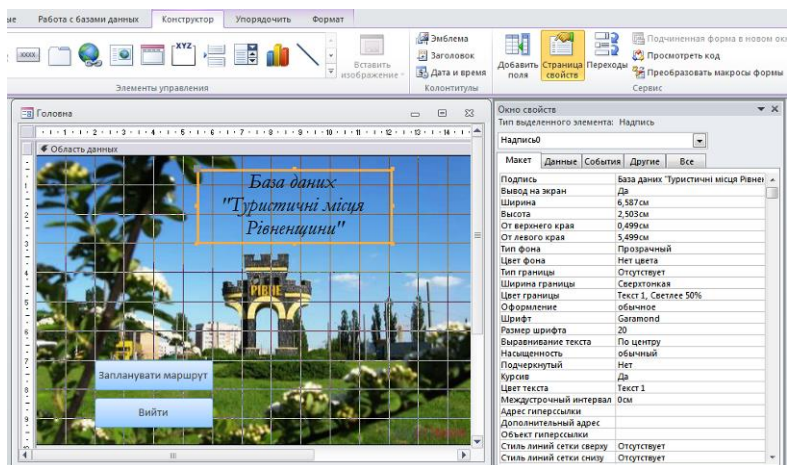


Рис. 2.5. Вікно властивостей елементів форми

Створення кнопок керування

Для прискороеного переходу між записами, формами, звітами, роботи з ними (друк, перегляд та інше) використовують кнопки. Для того щоб їх створити потрібно на вкладці **Конструктор** у групі **Форми** натиснути **Створити Кнопку** та виконати необхідні налаштування.

Розробка складних підпорядкованих форм

Створити форму, що дозволяє переглядати і редагувати записи тільки однієї таблиці, досить просто. Але на практиці потрібні форми, що дозволяють виконувати одночасне введення або перегляд даних, що зберігаються у двох або більше пов'язаних таблицях. Наприклад, при виведенні інформації про певний маршрут, відразу розміщувати інформацію про заправки, які розміщуються вздовж цього маршруту. Такі дані реально зберігаються у двох різних таблицях, пов'язаних відношенням "один-до-багатьох". Але можна створити форму, яка дозволить вводити, редагувати і переглядати всі ці дані одночасно. У таких випадках для відображення записів основної та підпорядкованої таблиць можливе використання багатотабличної ієрархічної форми (рис. 2.6).

Туристичні місця Рівненщини
Як доїхати з Рівного

Відкрити звіт

Назва об'єкта: Рівненський краєзнавчий музей

Станція: Чайка

Маршрут: АС "Чайка"-Театральна площа

Час від'їзду: Початок руху о 6:00, інтервал руху 3-4 хв

Тривалість маршруту: 17 хв

Вартість дитячого квитка (грн): 2

Вартість дорослого квитка (грн): 2

Вартість пільгового квитка (грн): 2

Заправки

Назва заправки	Адреса	Додаткові послуги
ОККО	м. Рівне, вул. В.Чорновола, 92	На заправці знаходиться міні-м
ОККО	м. Рівне, вул. Гагаріна, 18	На заправці знаходиться міні-м
ОККО	м. Рівне, вул. Дубенська, 74	На заправці знаходиться міні-м
ОККО	м. Рівне, вул. Київська, 84	На заправці знаходиться міні-м

Рис. 2.6. Приклад ієрархічної форми


Ієрархічною називається форма, що містить поля з головної і однієї або декількох підпорядкованих форм.

Підпорядковані форми використовуються для відображення даних з таблиць з відношенням "один-до-багатьох". Найбільш простим способом створення ієрархічної форми є використання Майстра форм.

На рис. 2.6 форма складається з двох частин. У верхній половині вікна форми містяться поля таблиці «Як доїхати з Рівного», тобто таблиці з боку зв'язку "один". У нижній половині знаходиться підпорядкована форма, що містить поля з таблиці «Заправки» (таблиця з боку зв'язку "багато").

Для створення такої форми потрібно:


- перейти на вкладку **Форми** у вікні бази даних, натиснути кнопку **Створити** і вибрати режим створення **Майстер форм**;
- у вікні **Нова форма** як джерело записів форми вказати таблицю «Заправки» (в ієрархічних формах базовою завжди є таблиця з боку "багато");
- у наступному вікні Майстра форм визначити головну і підпорядковану таблиці. Тут в якості головної слід вибрати таблицю «Заправки». У правій частині даного вікна діалогу відображаються поля таблиць: у верхній частині поля головної, а в нижній – поля підлеглої таблиці. Опція **Пов'язані форми** призначена для представлення даних без використання підпорядкованої форми;
- далі слід вибрати зовнішній вигляд підпорядкованої форми: табличний або стрічковий ;
- на наступних кроках Майстра форм вибрати стиль, задати імена форм, а також вказати режим, в якому буде відкрита форма після завершення її створення.

Крім того, можна створити підпорядковану форму на вже існуючій, скориставшись кнопкою панелі елементів керування **Підпорядкована форма**  та виконавши необхідні налаштування.

Випадаючі списки у формах


Для швидшого введення чи перегляду інформації можна створити у формі випадаюче **Поле зі списком**.

Якщо таке поле створюється на основі вже створеного поля таблиці з випадającym списком, то достатньо у формі клацнути правою кнопкою миші на потрібному полі та вибрати з контекстного меню **Перетворити елемент/У поле зі списком**.

Для створення нового поля у формі слід скористатися кнопкою панелі елементів керування **Поле зі списком**  та виконати необхідні налаштування.

Побудова діаграм у формах

Діаграми використовуються для наглядного представлення інформації з бази даних. В Access діаграми як окремі об'єкти не існують, вони лише можуть бути представлені як елемент форми чи звіту. Для того щоб створити діаграму потрібно відкрити пусту форму в режимі конструктора.

Вибрати інструмент **Діаграма**  на панелі елементів управління, щоб перемістити діаграму в форму. Далі з'явиться вікно створення діаграми, в якому потрібно обрати таблицю чи запит для створення діаграми і натиснути на кнопці **Далі**.

У другому діалоговому вікні майстра необхідно вибрати потрібні поля, за якими створюватиметься діаграма, і натиснути **Далі**.

У третьому діалоговому вікні вказати тип діаграми і натиснути **Далі**.

У наступному вікні вказується розмітка діаграми, тобто перетягуються кнопки полів, щоб визначити відповідність осей полям таблиці або запиту та виконати групування даних для відображення. На будь-якому етапі можна клацнути мишкою на кнопці **Взірець**, щоб подивитися як буде виглядати діаграма.

В останньому діалоговому вікні Майстра діаграм необхідно ввести заголовок діаграми. У цьому ж вікні є можливість для відтворення легенди. Можна вибрати відкриття форми в режимі форми або в режимі конструктора. Зробити вибір і клацнути мишкою на кнопці **Готово** для створення форми з діаграмою.

Оскільки можливості Майстра діаграм обмежені, для оформлення та редагування діаграм краще використовувати


MS Graph, запуск якого здійснюється подвійним клацанням миші на діаграмі у формі, відкритої в режимі Конструктора.

Кожен елемент діаграми має певний набір параметрів, значення яких встановлюються у відповідному вікні редагування, що відкривається подвійним клацанням миші на необхідному елементі.

Відредагувати текст легенди або самі дані можна через таблицю даних, яка також відображається в режимі MS Graph.

4) Створення макросів

Для автоматизації часто виконуваних задач корисно застосовувати макроси. Одним із найпростіших прикладів макросу є кнопка у формі, яка виконує дві дії – закриває поточну форму та відкриває іншу.

Макросом  називають набір з однієї або більш макрокоманд, що виконують певні операції. Макрокоманда - це інструкція, що визначає виконувани дії.

Є два шляхи запуску макросів. Безпосередній запуск макросів. Можна вручну запустити макрос з вікна бази даних, з конструктора, або з меню Робота з базами даних\Макрос\Виконати макрос. Непрямої запуск макросів. Можна вказати макрос як дію пункту меню, кнопки, обробника події. Також макрос можна запустити з іншого макросу.

Для створення та редагування макросів використовується **Конструктор макросів**. Щоб відкрити конструктор макросів необхідно перейти на вкладку **Створити** у групі **Додатково** та обрати елемент **Макрос**. Якщо ця команда недоступна, клацніть по стрілці поруч із кнопкою **Модуль** або **Модуль класу** та оберіть пункт **Макрос**. Після цього з'явиться конструктор макросів (рис. 2.7).

У вікні конструктора макросів формується список дій, які потрібно виконати під час їх запуску. В процесі першого запуску конструктора відображаються поля **Дія**, **Аргументи** і **Примітка**.

У полі **Дія** вводиться назва потрібної дії, яку виконуватиме макрос. У полі **Аргументи** відображаються

аргументи для кожної дії макросу, якщо вони потрібні. Праворуч відображається поле **Примітка**, де може міститися короткий опис.

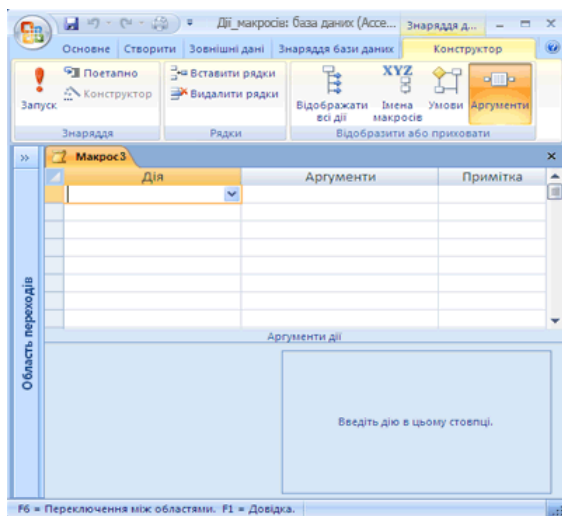


Рис. 2.7. Конструктор макросів

Знизу у розділі **Аргументи дії**, в лівій частині вводяться й редагуються аргументи для кожної дії макросу, якщо вони потрібні (рис. 2.8). Праворуч відображається вікно опису, де надається короткий опис кожної дії або аргумента. Потрібно клацнути дію або аргумент дії, щоб прочитати опис у вікні.

Можна також використовувати команди на вкладці **Конструктор** конструктора макросів, для створення, перевірки й запуску макросу.

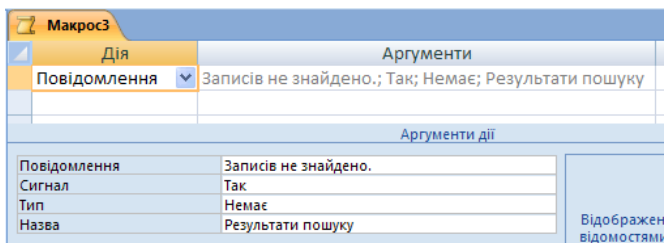



Рис. 2.8. Розділ Аргументи дії

Створення ізольованого макросу

1. На вкладці **Створити** у групі **Додатково** клацніть по елементу **Макрос**. Якщо ця команда недоступна, натисніть на стрілку поруч із кнопкою **Модуль** або **Модуль класу** та оберіть пункт **Макрос** . З'явиться конструктор макросів.

2. Додавання дії до макросу:

- у конструкторі макросів оберіть першу пусту клітинку у стовпці **Дія**;
- задайте назву потрібної дії або клацніть по стрілці для відображення списку доступних дій, після чого виберіть потрібну дію;
- у розділі **Аргументи дії** (у нижній частині конструктора макросів) за потреби задайте аргументи дії (рис. 2.8);

Під час введення аргументів в області **Аргументи дії** вони відображаються у стовпці **Аргументи** в списку дій. Проте у стовпці **Аргументи** аргументи тільки відображаються, вводити їх у цьому стовпці не можна.

Для відображення короткого опису кожного аргумента потрібно обрати поле аргумента в області **Аргументи дії** та прочитати опис праворуч від аргумента.


- за потреби можна ввести примітку до дії у стовпці **Примітка**.

3. Щоб додати до макросу наступну дію, потрібно перейти до іншого рядка дії, після чого повторити крок 2.

У разі запуску макросу, дії у програмі Access виконуються в тому порядку, в якому їх наведено у списку дій макросу.

Створення групи макросів

Для групування кількох пов'язаних макросів в один об'єкт, можна створити групу макросів. Для цього на вкладці **Створити** у групі **Додатково** потрібно обрати елемент **Макрос**. З'явиться конструктор макросів, в якому можна налаштувати потрібні стовпці та розділи макросу:

1. На вкладці **Конструктор** у групі **Відобразити або приховати** потрібно натиснути кнопку **Імена макросів** , за умови якщо її ще не натиснуто (рис. 2.9). У конструкторі макросів з'явиться стовпець **Ім'я макросу**.

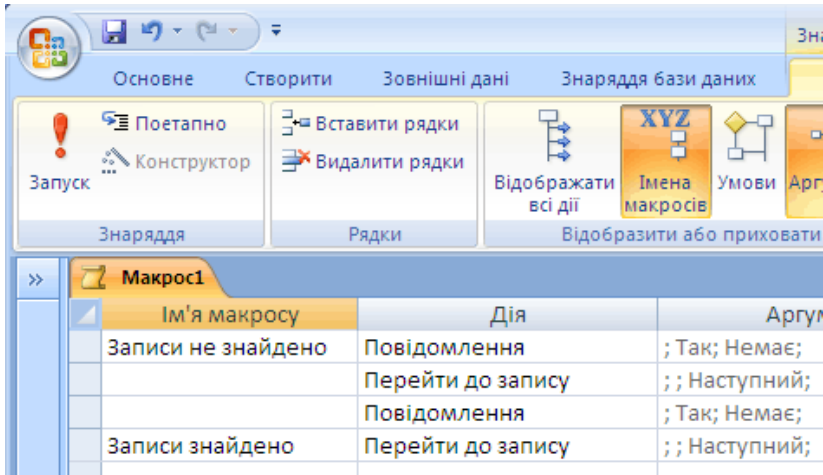


Рис. 2.9. Вікно створення групи макросів

Імена макросів є обов'язковими у групі, вони дозволяють розрізнити окремі макроси. Ім'я макросу відображається в одному рядку з першою дією макросу. Стовець імені макросу залишається пустим для можливості додавання подальших дій до цього макросу. Макрос закінчується, коли досягнуто ім'я наступного макросу.

2. У стовпці **Ім'я макросу** введіть ім'я першого макросу із групи макросів.

3. Додайте дії для виконання в першому макросі.

4. Перейдіть до наступного пустого рядка та введіть ім'я наступного макросу в стовпці **Ім'я макросу**.

5. Додайте дії для виконання в цьому макросі.

6. Повторіть кроки 4 і 5 для кожного макросу у групі.



Створення вбудованого макросу

Вбудовані макроси відрізняються від ізольованих макросів тим, що вони зберігаються у властивостях подій форм, звітів або елементів керування. Вони не відображаються як об'єкти в розділі **Макроси** в області переходів. Це спрощує процес керування базою даних, оскільки не потрібно відстежувати окремі об'єкти макросів, які містять макроси для форми або звіту. Окрім того, вбудовані макроси додаються до

форми або звіту під час кожного копіювання, імпорту або експорту.

Наприклад, якщо потрібно скасувати відображення звіту, коли він не містить даних, можна вбудувати макрос до властивості події звіту **Відсутність даних**. Можна скористатися дією **Повідомлення** для відображення повідомлення та використати дію **Скасувати подію**, щоб скасувати відображення звіту замість відображення пустої сторінки.


Порядок створення вбудованого макросу наступний:

1. Відкрийте форму чи звіт в режимі **Конструктор**  або **Подання розмічування** .

2. Якщо вікно властивостей не відображено, натисніть клавішу **F4**.

3. Оберіть елемент керування або розділ, який містить властивість події, до якої потрібно вбудувати макрос.

4. На аркуші властивостей перейдіть до вкладки **Подія**.

5. Клацніть властивість події, до якої потрібно вбудувати макрос, і натиснути кнопку .


6. У діалоговому вікні **Вибір конструктора** оберіть пункт **Конструктор макросів**, після чого натисніть кнопку **ОК**.


7. У конструкторі макросів додайте необхідні дії та аргументи.

8. Коли макрос готовий, натисніть кнопку **Зберегти** та кнопку **Закрити**.

Макрос запускатиметься щоразу під час виникнення події.

Змінення макросу

Додавання рядка дії. Оберіть рядок макроса, вище якого потрібно додати дію, та натисніть кнопку **Вставити рядки** .

Видалення рядка дії. Оберіть рядок макроса, який потрібно видалити, після чого натисніть кнопку **Видалити рядки** .


Переміщення рядка дії. Оберіть рядок, клацнувши маркер виділення рядка ліворуч від імені дії, та перетягніть рядок у потрібне місце.

Можна додавати, видаляти або перемішувати кілька рядків, вибравши групу рядків й виконуючи потрібні дії. Для вибору групи рядків оберіть заголовок першого рядка, який потрібно вибрати, натиснути і утримувати клавішу SHIFT, а потім клацніть по заголовку останнього рядка, який потрібно вибрати (заголовок рядка — це затінене поле ліворуч від кожного рядка дії.)

5) Параметри завантаження БД

Для зручного використання створеної БД у режимі користувача, необхідно задати відповідні параметри запуску, які відповідають за такі властивості як:

- наявність керуючих кнопок;
- назва додатку;
- вигляд БД (вікна, закладки);
- параметри відкривання БД (запуск основної форми).

Відповідні налаштування задаються на вкладці **Поточна база даних** (рис. 2.10) вікна **Параметри Access** .

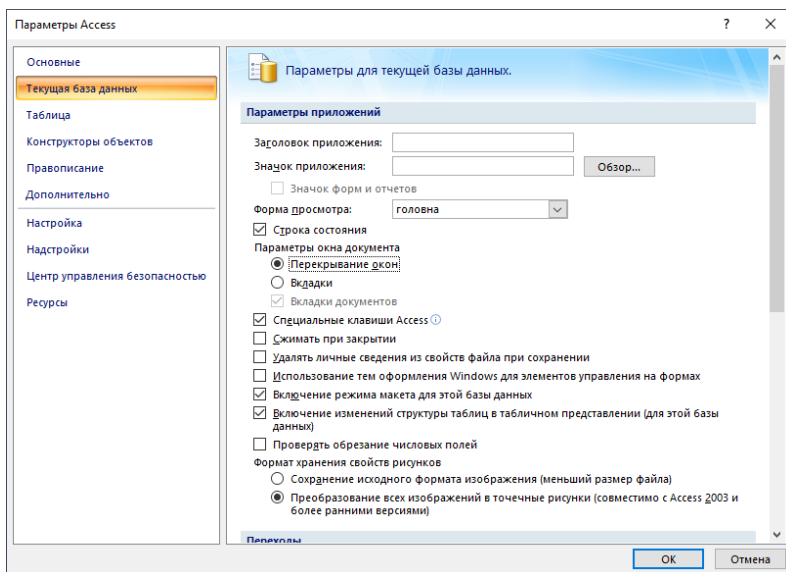


Рис. 2.10. Налаштування параметрів запуску

Налаштування пароля

При використанні пароля для шифрування бази даних всі дані стають нечитабельними в інших програмах, і для того щоб використовувати цю базу даних, користувачі повинні вводити пароль. Порядок шифрування з використанням пароля бази даних наступний:

1. Відкрийте базу даних, яку потрібно зашифрувати, в монопольному режимі:

1.1 На вкладці **Файл** натисніть кнопку **Відкрити**.

1.2 У діалоговому вікні **Відкрити** оберіть файл для відкриття та виділіть його.

1.3 Клацніть стрілку поруч із кнопкою **Відкрити** та оберіть параметр **Монопольно**.

2. На вкладці **Файл** натисніть кнопку **Відомості** та оберіть пункт **Зашифрувати паролем**.

3. Введіть пароль у полі **Пароль**, а потім повторіть його в полі **Перевірити**

4. Натисніть кнопку **ОК**.

Видалення пароля

1. На вкладці **Файл** натисніть кнопку **Відомості** та оберіть пункт **Розшифрувати базу даних**.

2. Відкриється діалогове вікно **Видалити пароль бази даних**.

3. Введіть пароль у полі **Пароль** і натиснути кнопку **ОК**.

Завдання для самостійної роботи. Ознайомтеся з функціями основних елементів керування формами.

Запитання для контролю:

1. Як створити форму у режимі майстра?
2. Як створити форму у режимі конструктора?
3. Як додати керуючі кнопки у форму?
4. Як створити випадаючий список у формі?
5. Як створити підпорядковану форму?
6. Що таке макрос?
7. Як створити макрос?

Лабораторна робота № 3

Запити та звіти

Мета: навчитись створювати запити та звіти у Microsoft Access.

Завдання: створити декілька запитів різного типу (на вибірку, з параметром...). Створити та налаштувати декілька звітів.

1) Конструктор запитів

В процесі роботи з таблицями можна в будь-який момент вибрати з бази даних необхідну інформацію за допомогою запитів.

Запит - це звернення до БД для пошуку або зміни в базі даних інформації, котра відповідає заданим критеріям.

За допомогою Access можуть бути створені такі типи запитів: запити на вибірку, запити на зміну, перехресні запити, запити з параметром.

Одним з найбільш поширених запитів є *запит на вибірку*, який здійснює відбір даних з однієї або декількох таблиць за заданими користувачем критеріями, що не призводить до змін в самій БД.

Конструктор – створює запит на основі порожнього бланка запиту.

Простий запит – створює простий запит з певних полів.

Перехресний запит – створює запит, дані в якому мають компактний формат, подібний формату зведених таблиць в Excel.

Повторювані записи – створює запит, що вибирає повторювані записи з таблиці або простого запиту.

Записи без підлеглих – створює запит, що вибирає з таблиці записи, не пов'язані з записами з іншої таблиці.

Створення запиту за допомогою конструктора

Для створення запиту слід перейти на вкладку **Запити** та натиснути кнопку **Створити**. З'явиться вікно **Новий запит** для вибору способу побудови запиту

При виборі Конструктора через діалогове вікно **Додавання таблиці** (рис. 3.1), додаються імена таблиць у вікно конструктора запиту.

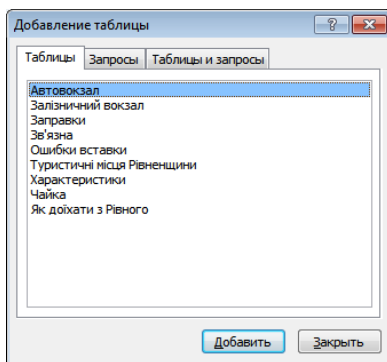


Рис. 3.1. Вікно додавання таблиці

Вікно **Додавання таблиці** складається з трьох вкладок, що містять переліки об'єктів, пропонованих програмою для проектування запиту: *Таблиці*, *Запити* і *Таблиці та запити*. При виборі вкладки **Таблиці** слід виділити потрібну нам таблицю із запропонованого списку і за допомогою кнопки **Додати** можна додати кілька таблиць.

Вікно конструктора розділене на дві частини. У верхній частині знаходяться вікна таблиць зі списками полів. Ім'я кожної таблиці відображається в рядку заголовка такого вікна. Нижня частина є бланком запиту, або, як його називають, QBE-областю (Query by Example – запит за зразком). Тут вказуються параметри запиту та дані, які потрібно відібрати, а також визначається спосіб їх відображення на екрані.

Для переміщення з верхньої панелі вікна в нижню і назад використовується клавіша **F6**.

У запит не слід включати всі поля обраних таблиць. Додати потрібні поля в бланк запиту можна шляхом перетягування їх імен зі списку, що знаходиться у верхній частині вікна конструктора в рядок бланка **Поле**.

Ще один спосіб – подвійне клацання на імені поля.

У рядку **Вивід на екран** прапорцем позначаються ті поля, які повинні виводитися на екран.

У загальному випадку поля, що вводяться в набір записів запиту, успадковують властивості задані для відповідних полів таблиці. Можна визначити інші значення властивостей, виконавши команду **Властивості** з меню **Вид**. Видалити поле з бланка запиту можна клавішею **Delete** або через меню **Редагування** командою **Видалити стовпці**.

При створенні запиту можна задати критерії, внаслідок чого за запитом буде здійснено відбір тільки потрібних записів.

Щоб знайти записи з конкретним значенням, потрібно ввести це значення в відповідне поле в рядку бланка **QBE Умова відбору**.

Критерії, що встановлюються в **QBE-області**, повинні бути взяті в лапки

Обчислення у запитах, побудова виразів

При створенні виразів для обчислюваних полів можна використовувати **Побудовник виразів**. Для цього потрібно клацнути по пустому полю в бланку **QBE**, а потім по кнопці панелі інструментів **Побудувати**, при цьому відкриється вікно **Побудовник виразів** (рис. 3.2). У верхній частині вікна розташована порожня область введення, в якій створюється вираз. Можна самим ввести вираз, але простіше використовувати кнопки, розташовані під областю введення.

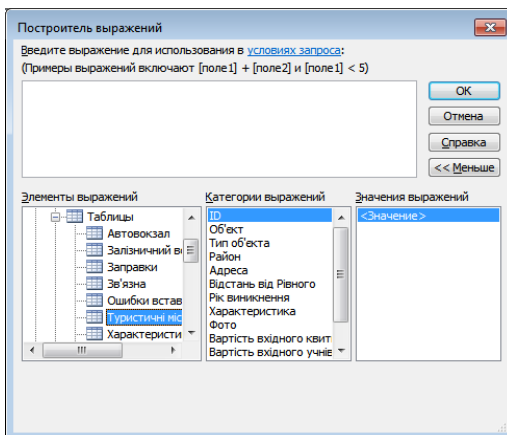


Рис. 3.2. Вікно Побудовника виразів

2) Запит з параметром

Якщо необхідно знайти запис, який відповідає конкретній умові, наприклад, відомості про земельну ділянку із кадастровим номером “1234567890123456789”, то ця умова записується у рядку **Умова відбору** в **Конструкторі запитів**. Такий запит називається *запитом на вибірку*. Для редагування умови запиту на вибірку необхідно відкрити його в режимі конструктора та змінити умову відбору.

Також поширеним є *запит з параметром*, коли при запуску запиту з’являється спеціальне вікно, де вводиться умова відбору. Для його створення в рядку **Умова відбору** в **Конструкторі запитів** у квадратних дужках записується вираз, який буде виводитися у вікні **Введіть значення параметру** запиту з параметром. Наприклад, для пошуку інформації про власника земельної ділянки за прізвищем, введемо у рядок **Умова відбору** вираз **[Введіть прізвище]** (рис. 3.3). Тоді при запуску запиту з’являється вікно **Введіть значення параметру**, де необхідно ввести умову відбору (введений у квадратних дужках текст слугуватиме підказкою, що саме потрібно ввести) (рис. 3.4). Для вибору за частиною запису в комірці слід у рядку **Умова відбору** вписати конструкцію: **Like "*" & [Введіть прізвище] & "*"**

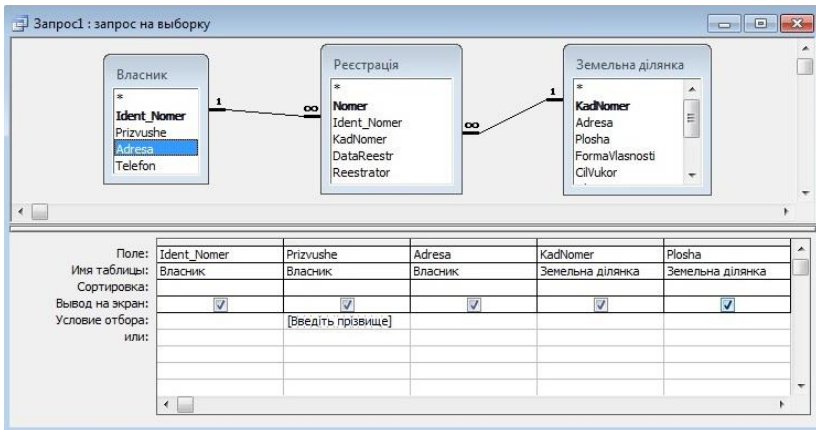


Рис. 3.3. Створення запиту з параметром у Конструкторі запитів

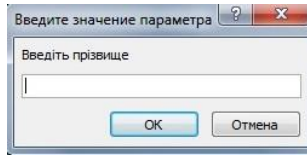


Рис. 3.4. Вікно введення умови відбору для запиту з параметром

3) Вибір даних за допомогою запитів - дій

Для швидкої зміни, вставки, створення або видалення наборів даних з БД використовуються керуючі запити або запити-дії.

Створення запитів-дій аналогічне створенню запитів-вибірок. Відмінність лише в тому, що в режимі конструктора запитів необхідно вказати тип дій над вибраними записами. Існує всього чотири типи запитів-дій: створення таблиці, оновлення, видалення, додавання (рис. 3.5).

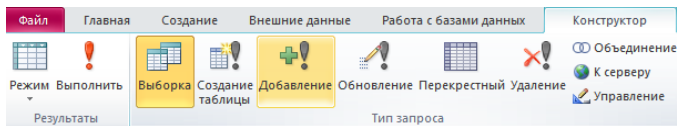


Рис. 3.5. Меню для вибору типу запиту

Створення таблиці. Цей запит дозволяє зберегти набір даних, що створені в результаті запиту, у вигляді таблиці.

Оновлення. Використовується для того, щоб оновити значення деяких полів для всіх записів, що відповідають результатам запиту.

Видалення. Використовується для видалення всіх записів, що відповідають результатам запиту.

Додавання. Використовується для додавання даних в наявну таблицю.

Microsoft Access відзначає запити-дії спеціальними значками.

Особливості роботи із запитом - діями

Щоб убезпечити себе від будь-яких випадковостей, необхідно перед виконанням запиту-дії створити резервну копію

використовуваної таблиці. Для цього необхідно перейти у вікно бази даних, вибрати таблицю, в якій відбудуться зміни, а потім скористатися командами **Правка/Копіювати** та **Правка/Вставити**, і в діалоговому вікні вказати ім'я для копії таблиці.

Створення запитів на оновлення, додавання і видалення неможливо без руйнування цілісності даних, тому тренуватися краще на створеній копії таблиці.

Новий запит бажано створити спочатку як запит на вибірку. Так ви зможете перевірити правильність відбору записів.

Перш ніж клацнути по кнопці **Запуск**, потрібно переглянути результат в режимі таблиці, скориставшись кнопкою **Вид**.

Запит на створення таблиці

Для збереження результатів вибірки в окремо існуючій таблиці використовуються запити на створення таблиці. Це прискорює доступ до даних, отриманих за запитом. Збереження у вигляді окремої таблиці корисно і в тому випадку, якщо які-небудь дані потрібно зберігати протягом тривалого часу.

Наприклад, зведену інформацію по роках необхідно зберігати протягом 5 років. Для цього створюється підсумковий запит і результат його виконання зберігається у вигляді окремої таблиці. Для цього в режимі **Конструктора запитів** потрібно вибрати команду **Запит/Створення таблиці**.

Access виведе на екран діалогове вікно «Створення таблиці», де потрібно вказати ім'я таблиці, в якій будуть збережені результати вибірки (рис. 3.6)

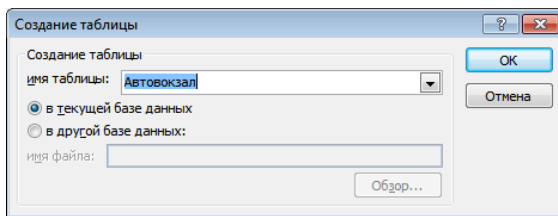


Рис. 3.6. Вигляд вікна запиту на створення таблиці

Запит на оновлення записів

Цей тип запитів-дій призначений для зміни значень деяких полів для всіх обраних записів. Після входу в режим **Конструктора запиту** потрібно вибрати команду **Запит/Оновлення**. Після цього в бланку запиту з'явиться рядок **Оновлення**. Цей рядок використовується для того, щоб задати нові значення для обраних записів.

Оновлювати значення можна не тільки в полях критеріїв, а в будь-якому полі таблиці. Також для обраних записів можна створювати обчислювані поля.

Запит на видалення записів

Цей вид запитів служить для видалення групи записів бази даних, які відповідають певним умовам. Звичайний запит можна перетворити в запит на видалення з допомогою команди **Запит/Видалення** в режимі **Конструктора запитів**. При цьому в бланку запиту з'явиться рядок **Видалення**, де можна ввести умову для видалення записів.

У режимі **Конструктора запитів** необхідно вказати таблицю, в якій проводитимуться видалення. Потім визначити критерії і обов'язково переглянути результати виконання запиту на вибірку. Якщо в результаті запиту присутні тільки записи, які підлягають видаленню, слід перетворити його в запит на видалення.

Запит на додавання записів

За допомогою запиту на додавання записів можна скопіювати вибрані записи і помістити їх в іншу таблицю. Можна також використовувати запит на додавання для перенесення даних у використовувану базу даних з іншого джерела (з якої-небудь іншої таблиці).

Виконання запиту на додавання записів можна почати (як і в інших запитах-діях) з запуску запиту-вибірки – для того, щоб переконатися , що копіюються необхідні записи.

Якщо необхідно додати записи до таблиці в іншій базі даних, спочатку слід приєднати таблицю-джерело до бази, яка містить цільову таблицю, за допомогою команди

Файл/Зовнішні дані/ Зв'язок з таблицями. Для відбору записів потрібно скласти запит на вибірку. Після цього, впевнившись у відборі необхідних даних, активізувати команду **Запит/Додавання** .

У вікні діалогу потрібно в полі **Ім'я таблиці** вказати ім'я таблиці, до якої потрібно приєднати дані з результуючого набору записів запиту.

4) Перехресні запити

За допомогою перехресного запиту можна більш наочно представити дані підсумкових запитів, які передбачають групування за кількома ознаками (за двома, зокрема).

У цьому випадку значення полів за першою ознакою групування можуть стати заголовками рядків, а за другою – заголовками стовпців. Для перетворення підсумкового запиту-вибірки необхідно перейти в режим **Конструктора** і виконати команду **Запит / Перехресний**. Після цього в бланку запиту рядок **Виведення на екран** буде замінений на рядок **Перехресна таблиця** . У цьому рядку потрібно вказати, як буде використано дане поле в перехресній таблиці: у вигляді заголовку рядків, полів або значень.

Існує можливість вивести дані в перехресній таблиці без створення в базі даних окремого запиту. Для цього слід використовувати **Майстер зведених даних**. У зведеній таблиці користувач має можливість змінювати заголовки рядків або стовпців , що дозволяє аналізувати дані різними способами.

У перехресному запиті відображаються результати статистичних розрахунків (такі як суми, кількість записів і середні значення), виконаних за даними з одного поля таблиці. Ці результати групуються за двома розділами даних, один з яких розташований в першому стовпчику таблиці, а другий – у верхньому рядку.

Можна створити перехресну таблицю в інтерактивному режимі за допомогою майстра **«Перехресний запит»**. Звернення до Майстра значно спрощує процес створення перехресної таблиці. Майстру необхідно вказати початкову таблицю, яка може бути таблицею з бази даних або бути результатом вибірки

даних з декількох таблиць. На основі вихідної таблиці або запиту Майстер створює підсумкову вибірку, у якій групує і сортує дані по полях, а також обчислює підсумкове значення по заданому полю даних. У перехресній таблиці, створеній за допомогою Майстра, присутній додатковий стовпець **Разом**, в якому відображаються сумарні підсумки даних.

5) Створення звітів

Звіт – це об'єкт бази даних, який використовується для відображення та підсумування даних. З допомогою звітів дані можна подати фактично в будь-якому форматі, з різними рівнями деталізації. Звіти можуть бути: *звичайні* – створені на основі однієї таблиці БД, та *реляційні* – створені на основі декількох зв'язаних таблиць.

Перш ніж створювати звіт, його необхідно спланувати.

Для створення звіту потрібно в головному вікні БД вибрати розділ **Звіти**, а в ньому командну кнопку **Створити**. При цьому відкривається вікно **Новий звіт**, де потрібно вибрати режим створення звіту. Після цього у випадіючому списку, в нижній частині вікна, потрібно вказати назву таблиці БД, запиту або фільтра, на основі якого буде створений звіт. В списку режимів створення звіту можна вибрати:

- **Конструктор** – створення структури звіту вручну, за допомогою конструктора звітів;
- **Майстер звітів** – створення звітів за допомогою майстра звітів;
- **Автозвіт в стовпець** – автоматичне створення звіту в колонку за зразком останнього створеного звіту такого типу. У цих звітах всі поля відображаються в одну колонку. Такі звіти дуже схожі на форми, але вони використовуються лише для відображення або друку даних, а не для їх введення та редагування
- **Автозвіт стрічковий** – автоматичне створення рядкового звіту за зразком останнього створеного звіту такого типу;
- **Майстер діаграм** – створення звіту у вигляді діаграми;
- **Почтовые наклейки** – створення звітів у вигляді поштових наклеюк.

Найбільш швидкий спосіб створення звітів – це використання функції **автозвіт**. При цьому автоматично створюється звіт за зразком останнього створеного звіту такого типу. Якщо звітів такого типу на ПК ще не було, то створиться звіт з параметрами прийнятими в Access за замовчуванням.

Для створення звіту за допомогою майстра звітів потрібно в діалоговому вікні **Новий звіт** вибрати режим **Майстер звітів** (у випадаючому списку в нижній частині вікна потрібно вибрати таблицю БД, запит або фільтр по формі). На першому кроці роботи майстра звітів відкривається вікно, де за допомогою кнопки **>** потрібно вибрати поля (**>>** – для вибору всіх полів), які повинні ввійти в звіт (рис. 3.7). Далі необхідно виконати всі наступні кроки по створенню звіту, які пропонує майстер, аж до виведення його на екран.

Поглиблене налаштування звітів можна здійснити за допомогою **Конструктора звітів**. Він дозволяє налаштувати шрифти та кольори відображення звіту, порядок полів і їх розміщення, вставити додаткові рисунки тощо. Зразок звіту відредагованого у **Конструкторі звітів** наведено на рисунку 3.8.

Завдання для самостійної роботи. Створіть описані вище запити-дії. Змініть стиль оформлення звіту.

Запитання для контролю:

1. Що таке запит?
2. Як створити запит на вибірку у режимі конструктора?
3. Як створити запит з параметром?
4. Що таке запити-дії?
5. Як створити запит на створення таблиці?
6. Як створити запит на додавання записів?
7. Як створити перехресний запит?
8. Що таке Звіт?
9. Як створити звіт у режимі майстра?
10. Як змінити стиль оформлення звіту?

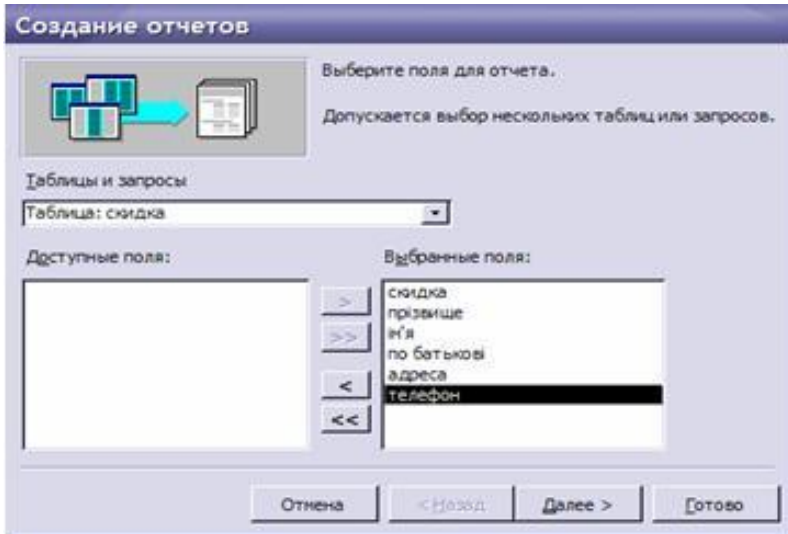


Рис. 3.7. Вікно майстра створення звітів



Замовлення №15

Доставка	Експортер	Замовлена продукція	Тара	Маса, кг	Кількість	Ціна за 1 кг
<input checked="" type="checkbox"/>	Рівненська кондитерська фабрика	Болеро вершково-апельсинові	Пошлучно	0,036	54	26,00 грн
					Вартість	50,54 грн
	Адмірал Бенбоу	Гофроящик		5	78	48,00 грн
					Вартість	18 720,00 грн
	Тандем кавове	Гофроящик		6	460	24,00 грн
					Вартість	66 240,00 грн
	Шоколадочка молочна	Гофроящик		4	450	18,00 грн
					Вартість	32 400,00 грн
	Болеро шоколадно молочні	Гофроящик		4	125	17,00 грн
					Вартість	8 500,00 грн
Загальна вартість						125 910,54 грн

Рис. 3.8. Звіт створений за допомогою конструктора звітів

Лабораторна робота № 4

Розробка схеми бази даних у DbDesigner

Мета: навчитись описувати схему даних у вигляді SQL коду.

Завдання: Розробити схему бази даних у DbDesigner, конвертувати у SQL код та створити БД у PostgreSQL

1) Робота з DbDesigner

Необхідно описати структуру бази даних, реалізованої у MS Access, у вигляді SQL коду. Для перегляду схеми даних у Access переходимо на панель **Робота з базами даних** та обираємо команду **Схема даних**. На екрані відобразиться структура таблиць та зв'язки між ними (рис. 4.1).

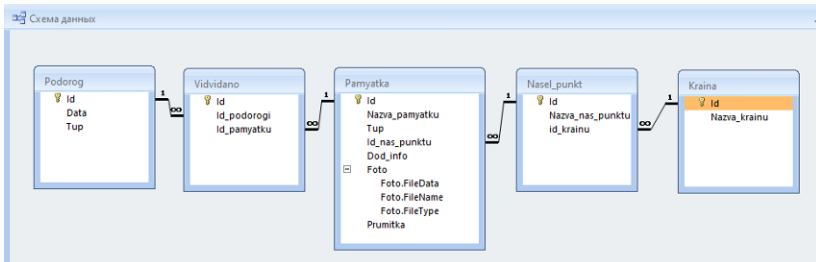



Рис. 4.1. Схема даних у MS Access

Для полегшення опису структури БД на мові SQL використаємо сайт <https://www.dbdesigner.net/>.

1. Заходимо на сайт та проходимо процедуру реєстрації натиснувши на **Login**. Можна спростити процес реєстрації авторизувавшись через акаунт Google.

2. Створюємо нову схему даних. Обираємо **New Project**. У вікні, що з'явилося обираємо **Create from scratch**, вводимо **назву проекту** та тип **postgresql** (рис. 4.2). Натискаємо **Create a new schema**.

3. Створюємо нову таблицю натиснувши на панелі інструментів кнопку **New Table** .

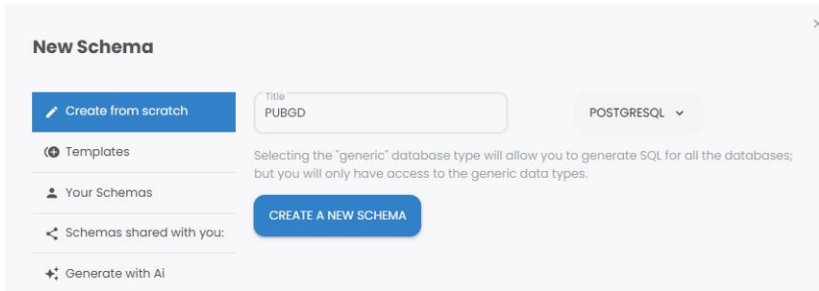


Рис. 4.2. Створення нової схеми у DbDesigner

4. У вікні редагування таблиці (рис. 4.3) задаємо **ім'я таблиці** та описуємо **імена полів**. Додавання нового поля здійснюється натисканням клавіша із + у лівій частині вікна. Для кожного поля задаємо **тип даних** та **розмір**.

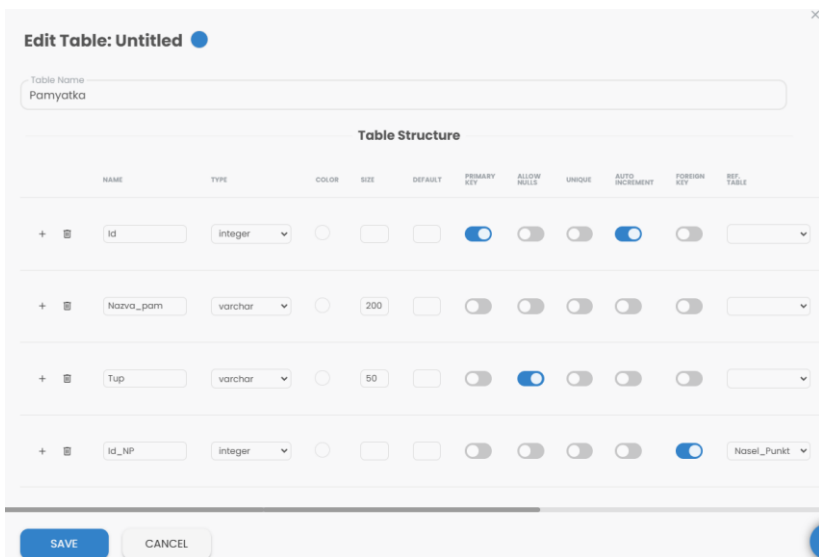


Рис. 4.3. Опис нової таблиці у DbDesigner

5. Для кожного поля доступний ряд налаштувань, які вмикаються/вимикаються відповідним перемикачем (рис. 4.3): Для ключового поля таблиці вмикаємо перемикач **Primary key**. Перемикач **Allow nulls** дозволяє залишати комірки у даному полі

не заповненими. При потребі забезпечити у полі лише унікальні значення вмикаємо перемикач **Unique**. За забезпечення автонумерації значень у полі відповідає перемикач **Auto Increment**.

6. За зв'язок між таблицями відповідає перемикач **Foreign key**. Для поля, що є зовнішнім ключем таблиці вмикаємо відповідний перемикач та з переліку **Ref. table** обираємо таблицю з відповідним йому первинним ключем (рис. 4.3).

Таким чином описуємо всі необхідні таблиці та зв'язки між ними (рис. 4.4).

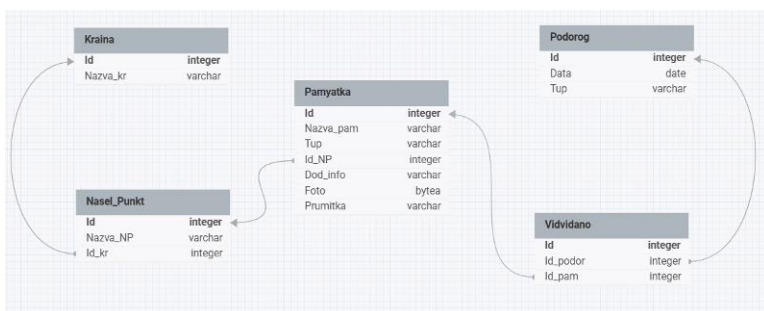


Рис. 4.4. Схема даних у DbDesigner

Для експорту створеної схеми у SQL-код обираємо на панелі інструментів **Export / Export SQL**. У вікні, що з'явилося обираємо тип бази даних **Postgresql** та натискаємо **Export** (рис. 4.5). Буде створено файл із розширенням *.sql, вміст якого можна переглянути за допомогою блокноту.

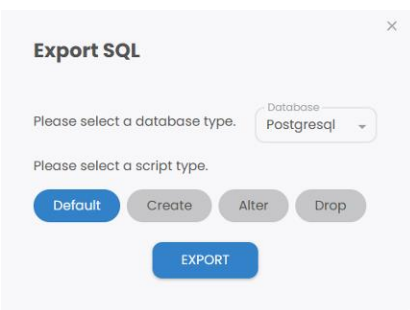


Рис. 4.5. Вікно експорту схеми даних у SQL

2) Робота з PostgreSQL

Необхідно запустити програму PostgreSQL, створити нову базу даних та виконати створений на попередньому етапі SQL-запит.

1. Запускаємо програму PostgreSQL через **pgAdmin**.
2. Створюємо нову базу даних, виконавши з **контекстного меню** розділу баз даних команду **Create / Database** (рис. 4.6).



Рис. 4.6. Створення нової БД у PostgreSQL

3. У вікні створення бази даних на вкладці **General** задаємо **ім'я** бази даних та натискаємо **Save** (рис. 4.7).

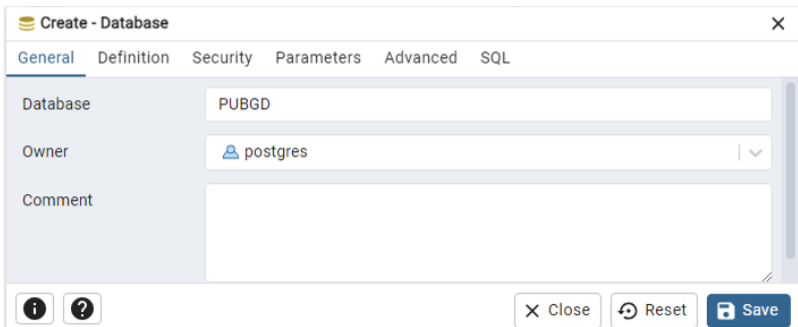


Рис. 4.7. Опис нової БД у PostgreSQL

4. З **контекстного меню** створеної БД обираємо пункт **Query Tool**.

5. У вікні запитів **вставляємо SQL код** згенерований у DbDesigner та запускаємо його на виконання натиснувши на панелі інструментів кнопку **Execute script** або **F5** (рис. 4.8). У разі успішного виконання – в нижній частині вікна має з'явитися повідомлення *Query returned successfully*.

Після цього у створеній базі даних у розділі **Schemas / public / Tables** мають відображатися створені таблиці (рис. 4.9).

Завдання для самостійної роботи. Розгляньте які типи даних підтримуються DbDesigner.

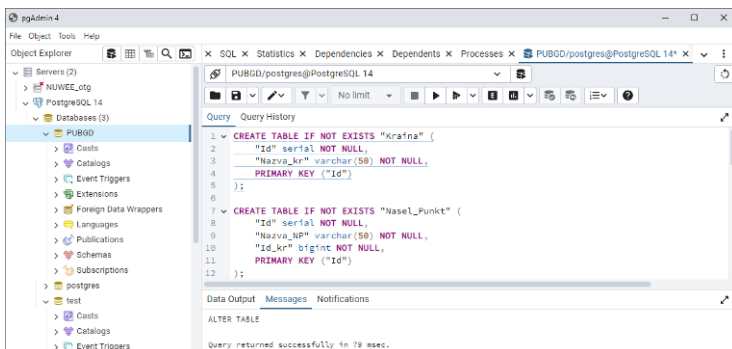


Рис. 4.8. Виконання SQL запиту на створення таблиць

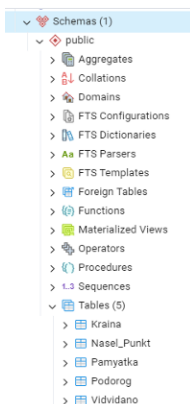


Рис. 4.9. Створені таблиці

Запитання для контролю:

1. Як створити нову схему даних у DbDesigner?
2. Як створити нову таблицю у DbDesigner?
3. Як описати зв'язок між двома таблицями у DbDesigner?
4. Які налаштування полів доступні у DbDesigner?
5. Як створити базу даних PostgreSQL?
6. Як запусити на виконання запит у PostgreSQL?

РЕКОМЕНДОВАНА ЛІТЕРАТУРА ТА РЕСУРСИ

1. Гайна Г. А. Основи проектування баз даних. К. : Кондор, 2008. 200 с.
2. Геоінформаційні системи і бази даних : монографія / В. І. Зацерковний, В. Г. Бурачек, О. О. Железняк, А. О. Терещенко. Ніжин : НДУ ім. М. Гоголя, 2014. 492 с.
3. Геоінформаційні технології та інфраструктура геопросторових даних: у шести томах. Том 2: Системи керування базами геоданих для інфраструктури просторових даних. Навчальний посібник / Кейк Д., Лященко А. А., Путренко В. В., Хмелевський Ю., Дорошенко К. С., Говоров М. К. : Планета-Прінт, 2017. 456 с.
4. Методичні вказівки до виконання лабораторних робіт з дисципліни «Побудова та управління базами геоінформації» студентами напряму підготовки 6.080101 „Геодезія, картографія та землеустрій” професійного спрямування „Геоінформаційні системи і технології” Частина 1 «Робота з базами даних» 05-04-38 / О. Є. Янчук, Н. В. Левчук., А. В. Прокопчук. Рівне : НУВГП, 2014. 52 с. URL: <http://ep3.nuwm.edu.ua/1339/>
5. Організація баз даних та знань / В. В. Пасічник, В. А. Резніченко; за заг. ред. М. З. Згуровського. Київ, 2006. 386 с.
6. Основи створення інтероперабельних геопросторових даних. / Ю. О. Карпінський та ін. Київ : КНУБА, 2023. 302 с.
7. DB Designer. Online Database Schema Design and Modeling Tool. URL: <https://www.dbdesigner.net/>
8. MS Office. Access. URL: <http://office.microsoft.com/uk-ua/access>