

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт та самостійних робіт
з дисципліни «Інформаційні системи і технології»
для студентів усіх спеціальностей.
Розділ «Система управління базами даних *Microsoft Access*»

Затверджено
методичною радою університету
протокол № 3 від 03 листопада 2023 р.

Харків
ХНАДУ
2024

Укладачі: Шевченко В. О.
Карпішен Б. С.
Костікова М. В.

Кафедра інформатики та прикладної математики

ВСТУП

Методичні вказівки призначені, як для самостійного вивчення і придбання навичок роботи студентами в системі управління базами даних (СУБД), так і під керівництвом викладача в аудиторіях, оснащених персональними комп'ютерами (ПК) або дистанційно. В наш час існує багато різних СУБД. Серед них гідне місце займає СУБД *Access*. Вона є складовою частиною найбільш поширеного пакету офісних програм *Microsoft Office*, має зручний інтерфейс і набір методів розробки, які дозволяють створювати професійні бази даних. Нині велика увага приділяється автоматизованим засобам обробки інформації у багатьох сферах діяльності: наукових і учбових організаціях, на підприємствах, довідкових службах, державних установах, банках тощо, тому знання і уміння працювати з програмним забезпеченням, що призначене для обробки та супроводу великих обсягів даних різних типів необхідно усім студентам як майбутнім кваліфікованим фахівцям. Саме таке призначення у СУБД *Access* і на це спрямовані дані методичні вказівки.

Методичні вказівки призначені для отримання знань про поняття і термінологію, що використовуються в базах даних та придбання навичок роботи із створення і експлуатації БД в СУБД *Access*. До цих умінь відносяться: створення структури таблиць БД і їх заповнення, формування запитів, створення форм, звітів тощо. Методичні вказівки складаються з восьми практичних робіт, кожна з яких розрахована на виконання студентами за дві академічні години. Кожна практична робота містить в короткому викладі необхідні теоретичні відомості для виконання роботи, робочі завдання і контрольні запитання, які корисні як для самоперевірки, так і для контролю набутих студентами знань викладачем. Виконання завдань практичних робіт контролює викладач, що проводить заняття. У кінці кожного заняття студент повинен представити викладачеві звіт по відпрацьованій темі, що містить: номер практичної роботи, назву теми, мету роботи, контрольні запитання, відповіді на них і, при необхідності, додаткові записи про особливості виконання завдань.

ПРАКТИЧНА РОБОТА № 1

ПОЧАТКОВІ ВІДОМОСТІ ПРО *MICROSOFT ACCESS*. РОЗРОБКА СТРУКТУРИ ТАБЛИЦЬ

Мета роботи: отримання навичок роботи зі створення структури таблиць.

Теоретичні відомості

Access – це система управління базами даних (СУБД), яка складається з комплексу програм, що дозволяють зберігати великі масиви даних у певному форматі, обробляти їх і представляти в зручному для користувача вигляді. *Access* дає також можливість автоматизувати багато керуючих функцій, які виконуються на підприємстві, такі як: взаємодія з постачальниками і покупцями продукції; робота з кадрами; розрахунок заробітної плати, облік матеріальних цінностей, відстеження маршрутів технологічних процесів тощо. За допомогою *Access* можна розробляти зручні форми введення і перегляду даних, а також складати складні звіти.

Всі об'єкти бази даних зберігаються в єдиному дисковому файлі. Основним структурним компонентом бази даних є таблиця, в якій зберігаються введені дані. Таблиця складається зі стовпців, які називають **полями**, і рядків, названих **записами**. Кожен запис таблиці містить всю необхідну інформацію про окремий елемент бази даних.

Поля визначають структуру бази даних і групові властивості даних.

Основні параметри поля:

- **Ім'я поля** - визначає, як слід звертатися до даних цього поля. Ім'я повинно бути унікальним, коротким і зрозумілим, відображати смислове навантаження змісту поля, містити не більше 64 символів включаючи пробіли. Бажано уникати вживання імен полів, що збігаються з іменами вбудованих функцій або властивостей *Microsoft Access*. Імена полів можуть містити будь-які символи, включаючи літери, цифри, пробіли, спеціальні символи, за винятком крапки (.), Знаку оклику (!), Апострофа (') і квадратних дужок ([,]), проте ім'я поля не повинно починатися з пробілу.

▪ **Типи даних полів**, які використовуються в *Access*, визначаються залежно від типів даних, які зберігаються в цих полях. В *Access* використовується дванадцять типів різних полів і способів їх використання.

Розглянемо ці типи.

1) **Текстові** – в полях цього типу можна зберігати до 255 символів – букви, розділові знаки, цифри та інші символи. Необхідно пам'ятати, що при введенні чисел в текстові поля вони сприймаються як текст, незважаючи на те, що зовні вони виглядають як числа.

2) **Числові** – поля цього типу містять число (тільки одне). З даними полів цього типу можна виконувати арифметичні і логічні операції.

3) **Грошові** – поля цього типу містять зазначені грошові одиниці. В *Access* вбудовані процедури, які відстежують, щоб були дотримані всі правила роботи з грошовими сумами, особливо при округленні рахунків.

4) **Дата/час** – залежно від заданого формату в поле зберігаються дата, час або дата і час спільно. Використовується для зберігання дат відпустки або отримання товарів, днів народжень та інших дат. В *Access* можна використовувати дати від 100 по 9999.

5) **Майстер підстановок** – даний тип поля вибирається в тому випадку, якщо необхідно зміст поля однієї таблиці використовувати для заповнення даними поля іншої таблиці. При виборі Майстра підстановок з'являється список що відкривається і містить набір постійних значень або значень з інших таблиць. Тип поля вибирається із запропонованого списку значень. При заповненні такого типу поля з'являється список, що відкривається, в якому треба вибрати необхідне для заповнення поля значення даних.

6) **Логічний** – цей тип поля зберігає логічне значення нуль (0, хибність) або одиниця (1, істина).

7) **Поле об'єкта OLE** – використовується для прикріплення до записів об'єктів OLE, таких як таблиці *Microsoft Office Excel*, документів *Word* тощо. Всі програми *Microsoft Office* підтримують технологію OLE, що дозволяє спільно працювати з даними за допомогою впровадження і зв'язування об'єктів.

8) **Поле МЕМО** – призначено для зберігання описів, приміток та інших довгих текстових даних, які мають більше ніж 255 симво-

лів. Воно може містити до 65535 символів інформації. Це майже 18 сторінок книжкового тексту.

9) **Гіперпосилання** – ці типи полів «розуміють» мову Гіперпосилань, що використовуються в Інтернеті. Ці типи полів можуть використовуватися для звернення до інших джерел інформації.

10) **Обчислюваний** – дозволяє записувати формули, які використовують одне або кілька полів в тій же таблиці, результат виконання обчислення формули зберігається в цьому ж полі. Наприклад, таблиця містить поле **ціна** і поле **кількість_товару**, якщо потрібно в таблицю включити витрати на придбання товару, то необхідно скористатися обчислюваним полем. При цьому з'являється підменю, в якому можна вибрати необхідні дані, а так само скористатися будівельником виразів для створення потрібної формули.

11) **Вкладення** – поле цього типу дозволяє приєднувати до запису різні файли: документи *Word*, електронні таблиці *Excel*, презентації *PowerPoint* або файли інших типів, включаючи графічні (фотографії різних об'єктів).

12) **Лічильник** – поле лічильника дозволяє створювати унікальні значення, єдина мета яких - забезпечити унікальність кожного запису. Найчастіше поле лічильника використовується в якості первинного ключа, особливо за відсутності відповідного природного ключа (заснованого на полі у якому дані не повторюються).

Кожне поле характеризується своїми властивостями, які використовуються при роботі з таблицями в базі даних.

Основні властивості поля:

- **Розмір поля визначає** граничну довжину даних (у символах) для текстового поля, або підтип даних (формат представлення) для числового поля.

- **Формат поля визначає** спосіб представлення даних.

- **Обов'язкове поле** – при присвоєнні полю цієї властивості необхідний обов'язковий введення даних в це поле при заповненні таблиці.

- **Індексоване поле** – властивість, яка забезпечує пошук і сортування записів за значенням даного поля і перевірку наявності повторів у значеннях полів.

- **Підпис поля** – властивість призначена для визначення змістовної назви поля. При відсутності цієї властивості використову-

ється ім'я поля.

При розробці **структури таблиці** необхідно, насамперед, визначити назви полів, з яких вона складатиметься, типи полів і їх розміри.


Розглянемо структуру таблиці. Більшість баз даних мають табличну структуру. Як ми знаємо, в табличній структурі адрес даних визначається перетином стрічок і стовпців. В базах даних стовпці називаються полями, а стрічки – записами. Поля формують структуру бази даних, а записи зберігають інформацію, яка в них міститься. Обсяг бази даних в *Access* обмежений одним гігабайтом, тобто сумарний обсяг усіх таблиць та інструментальних засобів для роботи з ними не повинен перевищувати цього значення. Запис містить інформацію про один елемент бази даних: одну людину, книжку, продукцію, рейс тощо. Він складається з полів, які формують структуру запису. Структура запису фактично визначає структуру таблиці, а також БД, якщо в ній є лише одна таблиця.

Ключове поле – поле, дані в якому повторюватися не можуть. Значення цього поля є унікальним для кожного запису. Значення ключових полів, що збігаються в двох таблицях, визначають зв'язок між даними цих таблиць.

Робочі завдання

Завдання 1. Створити навчальну базу даних **ВІДДІЛ КАДРІВ**, яка повинна допомогти виконувати деякі роботи співробітникам відділу кадрів помістивши в неї три таблиці: **СПІВРОБІТНИК**, **СКЛАД СІМ'Ї** і **ШТАТНИЙ РОЗКЛАД**, що містять інформацію про співробітників підприємства.

Порядок виконання.

Завантажити *Microsoft Access*, виконавши дії: **Пуск** → **Всі програми** → **Microsoft Office** → **Microsoft Access** (або клацнути на відповідній піктограмі  на **Робочому столі** або на **Панелі задач**).

На Робочому столі з'являється вікно додатка *Microsoft Access* (рис. 1.1). Вікно має типову структуру вікон додатків *Microsoft Office*.

Для створення нової бази даних виконати такі дії: у вікні *Microsoft Access* виконати команду **Файл** → **Створити** або клацну-

ти відповідну піктограму.

У полі **Доступні шаблони**, в лівій частині вікна, клацнути на піктограмі **Пуста база даних** (рис. 1.1).

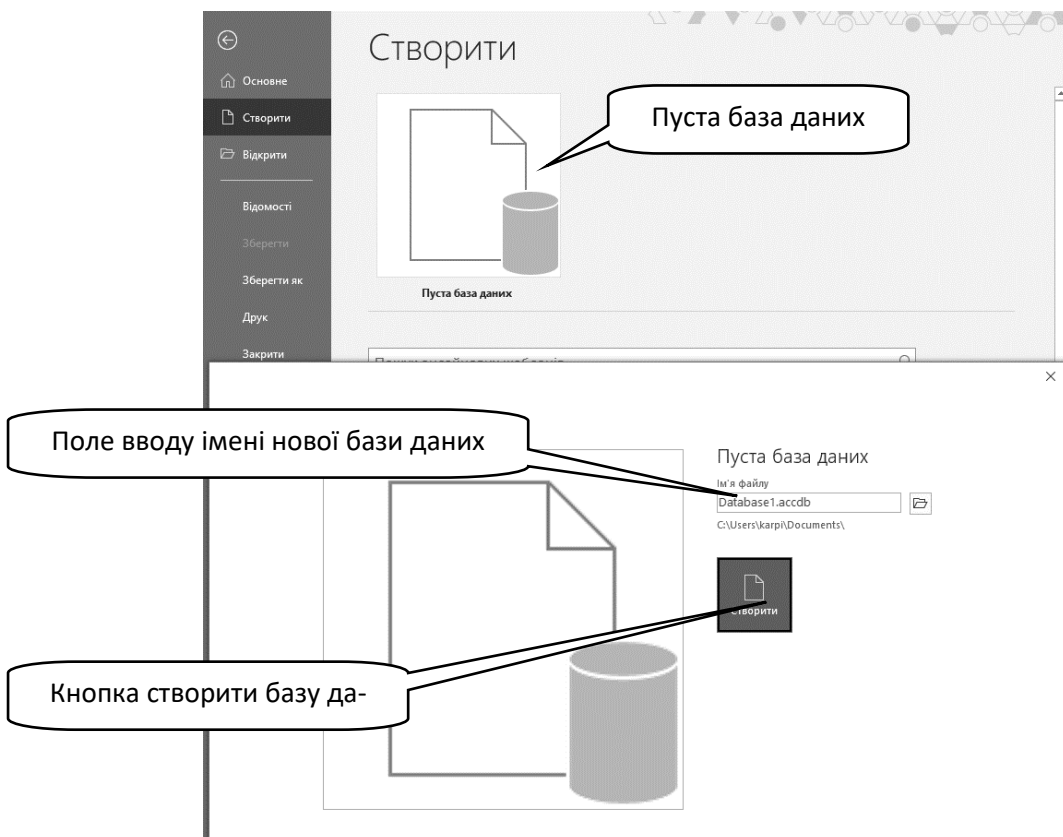


Рисунок 1.1 – Вид вікна додатку *Microsoft Access 2021*

У правій області вікна в полі **Пуста база даних** знаходимо поле **Ім'я файлу**, набираємо ім'я **БД_ВК_Прізвище**, де **Прізвище** – прізвище студента, що виконує дану практичну роботу.

Праворуч від цієї області є значок папки, клацаємо по ньому, відкривається вікно **Файл нової бази даних**. На диску **D:** створюємо або відкриваємо, якщо вона існує, папку своєї групи і клацаємо кнопку **ОК**. Нижче поля **Ім'я файлу** з'являється шлях до файлу **БД_ВК_Прізвище**, клацаємо по кнопці **Створити**, яка знаходиться ще нижче.

На екрані в рядку заголовка з'явиться назва **БД_ОК_Прізвище**, у вікні бази даних буде обраний режим роботи **Робота з таблицями**. З лівого боку стрічки вибираємо групу **Подання**, а в ній команду **Конструктор**. З'являється діалогове вікно **Зберегти як** (рис. 1.2), за допомогою якого присвоюється ім'я пер-

шій таблиці бази даних. У нашому випадку вводимо ім'я **ШТАТНИЙ РОЗКЛАД**.

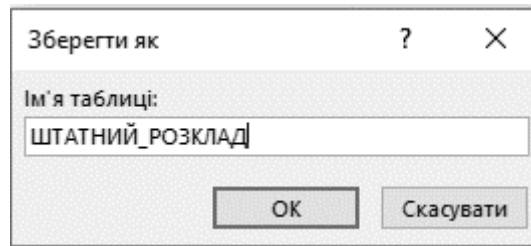


Рисунок 1.2 – Діалогове віно присвоєння імені таблиці

Додаток переходить в режим **Конструктор**, який призначений для вводу імен полів та їх властивостей (рис. 1.3).

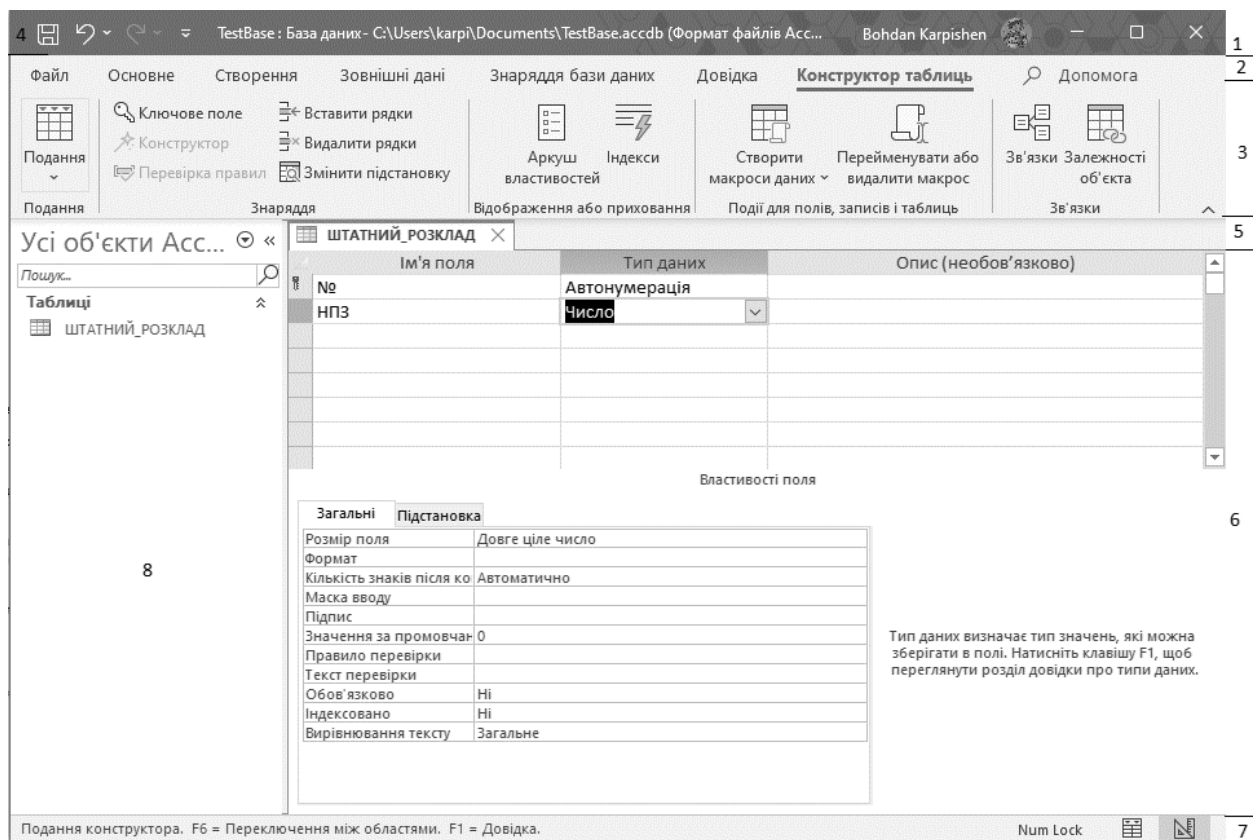


Рисунок 1.3. – Вигляд вікна для заповнення імен полів та їх властивостей

Перш ніж приступати до роботи з базою даних Ви повинні вивчити структуру вікна СУБД Access, що представлена на рис. 1.3.

1. **Рядок заголовка** містить назву бази даних, назва програми *Access 2021* і з правого боку кнопки управління: **Згорнути, Розгорнути/Згорнути в окно, Закрити**.

2. **Рядок вкладок** містить імена вкладок, назва яких визначає напрям дій, які можуть бути виконані за допомогою цих вкладок. Вкладки відкривають **Стрічки**.

3. **Стрічки** складаються з окремих груп, групи призначені для виконання окремих закінчених операцій за допомогою команд, які входять в цю групу. Наприклад, вкладка **Основне** містить групу **Подання**, яка має команди: **Подання таблиці, Конструктор**. За допомогою цих команд можна створювати структуру таблиць із зазначенням їх типів.

4. **Панель інструментів швидкого доступу** призначена для швидкого виконання команд, які в ній знаходяться. По замовчанню панель швидкого доступу має команди: **Зберегти, Скасувати, Повернути**. Як правило, розташована над вкладкою **Файл**, в нашому випадку під **Стрічкою**. З правого боку має кнопку налаштування за допомогою, якої можна внести, додатковими команди швидкого доступу, і вибрати місце розміщення під стрічкою або над стрічкою.

5. **Рядок заголовка об'єкта БД**, з яким працює користувач. Складається з імені об'єкта і кнопки закрити. При її використанні з'являється діалогове вікно, в якому запитується необхідність збереження змін. Об'єкти зберігаються у вигляді окремих файлів в базі даних.

6. **Робоче поле** містить об'єкти, з якими працює користувач.

7. **Рядок стану** відображає стан об'єкта, підказки (рекомендації) користувачеві, і з правого боку команди у вигляді піктограм, які можна виконати.

8. **Область навігації** – поле для вибору об'єктів БД.

Інші елементи ми будемо розглядати в процесі вивчення можливості *Access* при створенні баз даних.

Завдання 2. Створити структуру таблиці **ШТАТНИЙ РОЗКЛАД**, включивши в неї зазначені поля, вибравши для них відповідні типи. Відомості про характеристики полів таблиці **ШТАТНИЙ РОЗКЛАД** наведено в табл. 1.1

Таблиця 1.1 – Склад і характеристика полів таблиці ШТАТНИЙ РОЗКЛАД

Ім'я поля	Характеристика поля		Опис поля
	Тип даних	Властивості	
НПП	числовий	довге ціле, обов'язкове	№ по порядку
НП	текстовий	30 символів, обов'язкове	назва підрозділу
Посада	текстовий	15 символів, обов'язкове	посада
Кількість	числовий	довге ціле, обов'язкове	кількість штатних одиниць
Оклад	числовий	довге ціле, обов'язкове	посадовий оклад
ФЗПМ	числовий	довге ціле, обов'язкове	фонд заробітної плати на місяць
ФЗПР	числовий	довге ціле, обов'язкове	фонд заробітної плати на рік

Порядок виконання.

Для переходу в режим **Конструктор** на вкладці **Створити** в групі **Таблиці** виконайте команду **Конструктор таблиць**. При виконанні цієї команди з'являється вкладка контекстні інструменти **Робота з таблицями** (рис. 1.4), яка має контекстну вкладку **Конструктор**.

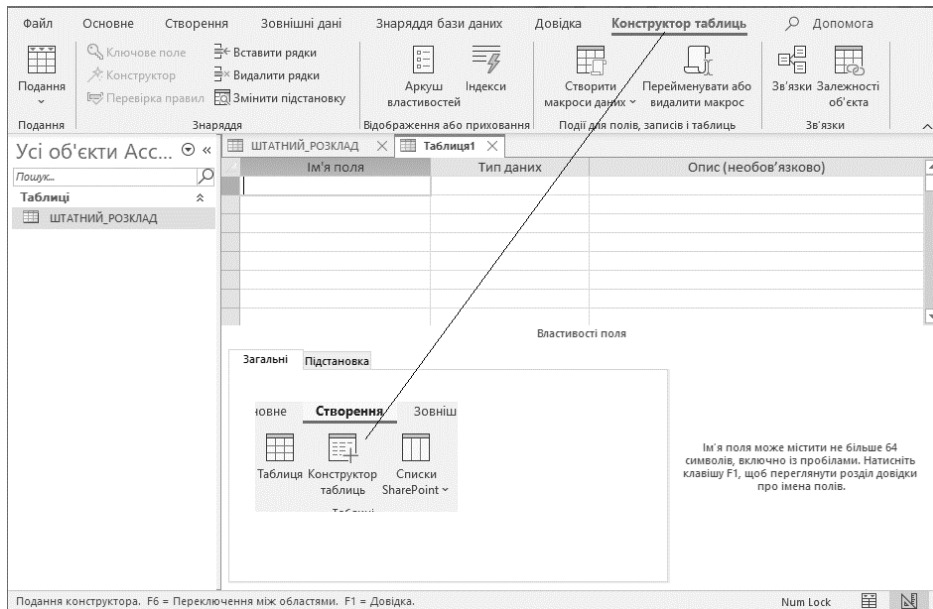



Рисунок 1.4 –Спливаюча вкладка **Робота з таблицями** з вкладкою **Конструктор**

У вікні конструктора таблиць для кожного поля необхідно заповнити відповідні розділи: **Ім'я поля**, **Тип даних**, **Опис**.

В розділ **Ім'я поля** необхідно занести дані стовпця Ім'я поля з табл. 1.1.

У розділі **Тип даних** треба встановити необхідний тип згідно даних стовпця **Тип даних** з табл. 1.1. Для зміни типу треба розкрити вікно вибір типу, клацнувши по кнопці розкриття списку , та вибрати зі списку необхідний тип. При цьому в нижній частині екрана в розділі **Розмір поля** поля з'явиться інформація про властивості даного типу поля. Ці дані треба скоригувати відповідно до даних стовпця **Властивості** з табл. 1.1.

Наприклад: для створення поля № за порядком в стовпець **Ім'я поля** вводимо НЗП. Потім переходимо в стовпець **Тип даних**, розкриваємо список типів клацанням по стрілці і вибираємо тип **Числовий**. Після чого в розділі **Розмір поля** поля на вкладці **Загальні** встановлюємо в рядку **Розмір поля** значення **Довге ціле число**, а в рядку **Обов'язково** поле значення **Так**. Всі інші властивості, установка значень яких не передбачені, залишаються такими, які вони мають за замовчуванням. Тепер можна перейти до створення наступного поля.

Після створення структури таблиці необхідно задати ключове поле. В якості ключового поля вибрати поле **Посада**. Для створення ключового поля виділити поле, для цього клацнути зліва від імені поля на смузі виділення. Потім вибрати вкладку **Робота з таблицями** → **Конструктор** і клацнути по піктограмі **Ключове поле**. Зліва від імені поля з'явиться зображення ключа.

Після створення структури таблиці необхідно зберегти її, виконавши команди **Файл** → **Зберегти як**. У вікні **Зберегти як** ввести ім'я створеної таблиці: **ШТАТНИЙ РОЗКЛАД**, потім – клацнути **ОК**. Таблиця буде збережена в **БД_ВК_Прізвище**, де **Прізвище** – Ваше прізвище.

Завдання 3. Аналогічним чином створити структуру таблиці **СПІВРОБІТНИК**. Відомості про поля таблиці наведено в табл. 1.2.

Таблиця 1.2 – Склад та характеристика полів таблиці СПВРОБІТНИК

Ім'я поля	Характеристика поля		Опис поля
	Тип даних	Властивості	
Ідент код	текстовий	10 символів, індексоване, ключове	Ідентифікаційний код
Прізвище	текстовий	20 символів, обов'язкове	Прізвище
Ім'я	текстовий	15 символів, обов'язкове	Ім'я
По-батькові	текстовий	15 символів, обов'язкове	По-батькові
Стать	текстовий	1 символ, необов'язкове	Стать
Дата народж	дата/вчас	маска вводу 00.00.0000, необов'язкове	Дата народження
Місце народж	текстовий	15 символів, обов'язкове	Місце народження
Освіта	текстовий	15 символів, обов'язкове	Освіта
Посада	майстер підстановок	15 символів, індексоване, допускаються збіг, обов'язкове	Посада
Стаж	числове	довге ціле, обов'язкове	Стаж роботи
Сім полож	текстовий	11 символів, , необов'яз.	Сімейне полож.
Дата зарахування	дата/час	маска вводу 00.00.0000, необов'язкове	Дата зарахування на роботу
Тел	текстовий	8 символів, , необов'язкове	Телефон
Адреса	поле МЕМО	необов'язкове	Домашня адреса

Для створення поля **Посада** в полі **Тип даних** необхідно вибрати **Майстер підстановок**. Відкриється перше діалогове вікно **Підстановки**. У цьому вікні вибирається спосіб, яким стовпець підстановки отримає свої значення: з таблиці або запиту, або буде введений фіксований набір значень. Вибираємо **Поле після підстановки** отримає значення з **іншої таблиці або запиту** і клацаємо кнопку **Далі**.

В наступному діалоговому вікні вибираємо таблицю **ШТАТНИЙ РОЗКЛАД** і знов клацаємо по кнопці **Далі**.

В наступному діалоговому вікні у списку **Доступні поля** вибираємо поле **Посада**, яке використовується як стовпець підстановки, і натискаємо кнопку для перенесення поля **Посада** до списку

Вибрані поля, після чого знову клацаємо по кнопці **Далі**.

У наступному діалоговому вікні в разі необхідності задається порядок сортування списку записів стовпця підстановки, тут ми нічого не встановлюємо і просто клацаємо по кнопці **Далі**.

Наступне діалогове вікно дозволяє змінити ширину стовпця підстановки. Потім запитується назва стовпчика - залишаємо **Посада**. Клацаємо по кнопці **Далі**. З'явиться повідомлення про те, що перед створенням зв'язку необхідно зберегти таблицю. Клацнути кнопку **Так** і зберегти таблицю під ім'ям **СПВРОБІТНИК**. При збереженні буде запропоновано створити ключові поля. Слід відмовитися від створення ключових полів, для чого клацнути на кнопці **Ні**. У розділі **Тип даних** буде вказаний тип **Текстовий**, тобто тип, відповідний типу поля підстановки з таблиці **ШТАТНИЙ РОЗКЛАД**.

Закінчити створення структури таблиці **СПВРОБІТНИК**. У якості ключового, вибрати поле **Ідент код**.

Поле **Посада** вибрати як індексоване. Для цього в розділі **Властивості поля** вибрати рядок **Автонумерація**, клацнути по кнопці розкриття списку і вибрати рядок **Так (Допускаються співпадіння)**.

Закрити вікно структури таблиці **СПВРОБІТНИК** і зберегти при цьому таблицю із змінами клацанням на кнопці **Так**.

Завдання 4. Створити структуру таблиці **СКЛАД СІМ'Ї** (табл. 1.3).

Таблиця 1.3 – Структура таблиці **СКЛАД СІМ'Ї**

Ім'я поля	Характеристика поля		Опис поля
	Тип даних	Властивість	
Ідент код	майстер підстановок	10 символів, обов'язкове, індексоване, допускаються збіг (совпадения)	Ідентифікаційний код
Відношення	текстовий	10 символів, необов'язко.	Відношення
Прізвище	текстовий	20 символів. обов'язкове	Прізвище
Ім'я	текстовий	15 символів. обов'язкове	Ім'я
По-батькові	текстовий	15 символів. обов'язкове	По-батькові
Дата народж	дата/час	маска вводу 00.00.0000 обов'язкове	Дата народження

При створенні поля **Ідент код** в якості типу даних вибрати **Майстер підстановок** як стовпця підстановки використати поле **Ідент код** з таблиці **СПВРОБІТНИК**.

Поле **Ідент код** вибрати як індексоване. Для цього в розділі **Властивості поля** вибрати рядок **Автономерація**, клацнути по кнопці розкриття списку і вибрати рядок **Так (Допускаються співпадіння)**.

Ключове поле не задавати. Зберегти створену структуру таблиці **СКЛАД СІМ'Ї**.

Контрольні запитання

1. Призначення програми *Microsoft Access*.
2. Що називається полем і записом у БД?
3. Назвіть основні параметри поля.
4. Які типи даних полів Вам відомі?
5. Назвіть основні властивості полів.
6. Яке поле можна вибрати в якості ключового?
7. Як зробити поле індексованим?
8. Що таке тип поля?
9. Що таке формат поля?
10. У яких випадках застосовують Майстер подстановок?
11. Як змінити розмір поля?
12. Опишіть основні елементи вікна *MS Access*.

ПРАКТИЧНА РОБОТА № 2 ЗАПОВНЕННЯ ТА ЗВ'ЯЗУВАННЯ ТАБЛИЦЬ

Мета роботи: отримання навичок по заповненню таблиць, створенню і модифікації зв'язків між таблицями.

Теоретичні відомості

Access – реляційна база даних. Одне з найважливіших переваг реляційних баз даних полягає в можливості зберігання логічно згрупованих даних у різних таблицях і в наданні зв'язків між ними шляхом об'єднання їх в єдину базу. Для здійснення зв'язку таблиці

повинні мати поля з однаковими іменами або хоча б з однаковими форматами даних. Зв'язок між таблицями встановлює відношення в цих полях між значеннями, що збігаються. Завдяки зв'язкам інформація з однієї таблиці стає доступною для іншої, що дозволяє забезпечити цілісність даних. Така організація даних дозволяє зменшити надмірність збережених даних, спрощує їх введення, організацію запитів і звітів.

В *Access* можна задати три види зв'язків між таблицями: один-до-багатьох, багато-до-багатьох і один-до-одного.

Зв'язок один-до-багатьох – найбільш частий тип зв'язку, який використовується між таблицями. У такому зв'язку кожному запису в таблиці А може відповідати кілька записів у таблиці В, а запис у таблиці В може мати тільки один відповідний йому запис в таблиці А.

Зв'язок один-до-багатьох створюється в тому випадку, коли тільки одне з полів є ключовим або має унікальний індекс, тобто значення в ньому не повторюються.

Зв'язок багато-до-багатьох – одному запису в таблиці А може відповідати кілька записів у таблиці В, а одному запису в таблиці В може відповідати кілька записів у таблиці А. Така схема реалізується тільки за допомогою третьої (*зв'язуючої*) таблиці, ключ якої складається, принаймні, з двох полів, одне з яких є спільним з таблицею А, інше – спільним з таблицею В.

Зв'язок один-до-одного – запис у таблиці А може мати не більше одного пов'язаного запису в таблиці В і навпаки.

Зв'язок між таблицями відображається у вікні **Схема даних** у вигляді лінії, яка з'єднує два поля різних таблиць. При цьому одна з таблиць вважається головною, а інша – зв'язаною. Головна таблиця бере участь у зв'язку своїм ключовим полем (назва цього поля на схемі даних відображається напівжирним шрифтом).

Зв'язок між таблицями дозволяє виключити можливість видалення або зміни даних у ключових полях головної таблиці, якщо з ним пов'язані які-небудь поля інших таблиць, або зробити так, що при видаленні або зміні даних в ключових полях головної таблиці відбувається автоматичне відстеження (*видалення або зміна*) відповідних даних в полях пов'язаної таблиці.

Робочі завдання

Завдання 1. Заповнити даними структури таблиць, створені на першому занятті.

Заповнення таблиць доцільно починати з таблиці **ШТАТНИЙ РОЗКЛАД** (табл. 2.1), оскільки поле **Посада** цієї таблиці використовується як стовпець підстановки для заповнення відповідного поля таблиці **СПІВРОБІТНИК**.

Для заповнення у вікні бази даних двічі клацнути на значку потрібної таблиці. На екрані з'явиться структура БД в табличному вигляді.

Заповнення проводиться по записами, повністю для всього рядка. Після заповнення першого рядка слідом за ним з'явиться новий порожній рядок.

Таблиця 2.1 – Таблиця ШТАТНИЙ РОЗКЛАД

НЗП	НП	Посада	Кількість	Оклад	ФЗПМ	ФЗПР
1	дирекція	бухгалтер	2	2300	4600	55200
2	дирекція	гол. бухгалтер	1	4300	4300	51600
3	дирекція	директор	1	5300	5300	63600
4	сл. управління	диспетчер	1	1000	1000	12000
5	сл. управління	інженер по ТБ	1	1500	1500	18000
6	дирекція	гол. інженер	1	4300	4300	51600
7	дирекція	зам. директора	1	5000	5000	60000
8	сл. перевозок	методист	2	2000	4000	48000
9	дирекція	начальник ВК	1	1500	1500	18000
10	сл. перевозок	водій	4	3500	14000	168000
11	сл. управління	статистик	1	1000	1000	12000
12	сл. управління	лікар	1	1500	1500	18000

Примітка: ВК – відділ кадрів

Таблиця 2.2 – Таблиця СПІВРОБІТНИК

№ п/п	Ідент. код	Прізвище	Ім'я	По батькові	Стать	Дата народження
1	1314152347	Старченко	Світлана	Борисівна	ж	22.04.1963
2	1545678990	Архипов	Сергій	Іванович	ч	23.03.1969
3	1624790203	Круговий	Геннадій	Іванович	ч	22.04.1965

№ п/п	Ідент. код	Прізвище	Ім'я	По батькові	Стать	Дата народження
4	1748576413	Царьова	Анна	Миколаївна	ж	30.07.1970
5	1934789231	Каменєва	Тетяна	Дмитрієвна	ж	24.06.1979
6	1955443781	Безродний	Володимир	Михайлович	ч	05.09.1973
7	2014654788	Садчиков	Аркадій	Вікторович	ч	10.01.1977
8	2055894321	Бронзов	Станіслав	Іванович	ч	12.11.1980
9	2178943214	Мапошенко	Юрій	Миколайович	ч	21.11.1984
10	2200987654	Коваль	Олександра	Миколаївна	ж	31.03.1985
11	2233668943	Строков	Олег	Вікторович	ч	05.08.1975
12	2314743296	Бородулін	Андрій	Васильович	ч	31.12.1989

№ п/п	Місце народж	Освіта	Посада	Стаж роботи	Сім пол
1	м. Кривий рік	середня	статистик	27	одружена
2	м. Харків	вища	директор	21	одружений
3	м. Одеса	вища	гол. інженер	25	вдівець
4	м. Харків	вища	гол. бухгалтер	20	одружена
5	м. Львів	вища	начальник ВК	11	одружена
6	м. Харків	вища	зам. директора	17	одружений
7	м. Вінниця	середня	диспетчер	13	неодружений
8	м. Житомир	вища	водій	10	одружений
9	м. Одеса	вища	інженер по ТБ	6	одружений
10	м. Київ	вища	методист	5	одружена
11	м. Кривий ріг	вища	водій	14	одружений
12	м. Київ	вища	лікар	1	неодружений

№ п/п	Дата зачісл	Тел.	Адреса
1	24.09.2013	65-12-13	прос. Тракторобудівників, 21, кв.30
2	10.12.2001	23-10-70	пров. Хрустальний, 8
3	01.09.2005	68-14-13	вул. Світла, 14, кв.55
4	01.01.2003	47-23-15	вул. Алчевських, 24, кв.1
5	30.12.2010	65-67-72	вул. Героїв праці, 28-Б, кв.76
6	01.09.2008	32-32-14	прос. Незалежності, 44, кв.55
7	15.10.1999	10-12-10	пров. Короленка, 2, кв.11
8	31.08.2004	47-15-20	вул. Кулиничівська, 6, кв.2
9	31.08.2007	43-35-13	вул. Сумська, 54, кв. 2
10	01.10.2008	47-67-33	вул. Свободи 5, кв. 2

№ п/п	Дата зачисл	Тел.	Адреса
11	10.09.2010	69-05-03	прос.Олександровський,162,кв.161
12	31.08.2012	27-14-12	вул. Каштанова, 117, кв.20

Таблиця 2.3 – Таблиця СКЛАД СІМ'Ї

№ п/п	Ідент. код	Відношення	Прізвище	Ім'я	По батькові	Дата народження
1	1314152347	батько	Старченко	Микола	Іванович	12.01.1940
2	1314152347	мати	Старченко	Людмила	Яковлівна	25.12.1942
3	1545678990	син	Архипов	Дмитро	Сергійович	01.09.1993
4	1748576413	чоловік	Царьов	Петро	Олексійович	14.11.1968
5	1934789231	чоловік	Каменєв	Олександр	Іванович	29.04.1970
6	1955443781	дочка	Безродна	Алла	Володимирівна	24.06.1998
7	2014654788	мате	Садчикова	Марія	Іванівна	29.04.1957
8	2055894321	дочка	Бронзова	Інна	Станіславівна	15.12.2001
9	2178943214	син	Мапошенко	Ігор	Юрійович	22.06.2007
10	2178943214	син	Мапошенко	Володимир	Юрійович	23.08.2010
11	2233668943	дочка	Строкова	Юлія	Олегівна	28.07.2000
12	2233668943	дочка	Строкова	Наталія	Олегівна	14.03.2010

Завдання 2. Змінити зв'язок між таблицями для автоматизації поновлення зв'язаних полів.

В Access система самостійно встановлює зв'язки між таблицями, якщо в таблицях вже встановлені ключові поля. Але у випадку, якщо зв'язки не створені, або їх необхідно змінити, це можна зробити вручну.

Для виконання цього завдання необхідно вибрати вкладку **Знаряддя бази даних** → **Зв'язки** → **Зв'язки**. З'явиться вкладка **Конструктор зв'язків** та відкриється поле **Зв'язки** із зображенням таблиць і їх зв'язків. Якщо в цьому полі вказані не всі таблиці, їх можна додати. Для цього на порожньому місці поля **Зв'язки** натисніть правою кнопкою миші і виберіть команду **Відобразити таблицю**. З'явиться діалогове вікно **Додавання таблиць**. В ньому вибрати необхідну таблицю і натиснути кнопку **Додати**. Таблиці розташувати за порядком згідно рис. 2.1.

Далі необхідно замінити існуючі зв'язки між таблицями на зв'язки Один-до-багатьох. Для цього видалити існуючі зв'язки,

клацнувши на кожній лінії правою кнопкою миші та створити нові. Для створення зв'язків, виконати перетягування лівою кнопкою миші назву полів однієї таблиці на відповідне поле іншої. При цьому з'явиться прямокутник зі знаком +, при відпуску ЛКМ з'явиться діалогове вікно **Редагування зв'язків**. У цьому вікні вказано, звідки і куди прокладено зв'язок, якщо типи даних з'єднаних полів співпадають (виконуються правила створення зв'язків), то в полі **Тип відношення** з'явиться повідомлення, наприклад, **Один-до-багатьох**. Для автоматичної підтримки цілісності БД встановити прапорці на властивостях зв'язків: **Забезпечення цілісності даних**; **каскадне оновлення пов'язаних полів**; **каскадне видалення пов'язаних полів**. Далі клацнути кнопку **Створити**. Якщо таблиці були створені та заповнені коректно, з'явиться зображення зв'язку у вигляді лінії як на рис 2.1. На одному кінці буде стояти одиниця, ці дані є унікальними, а на іншому кінці зв'язку стоятиме знак **∞**, тобто багато.

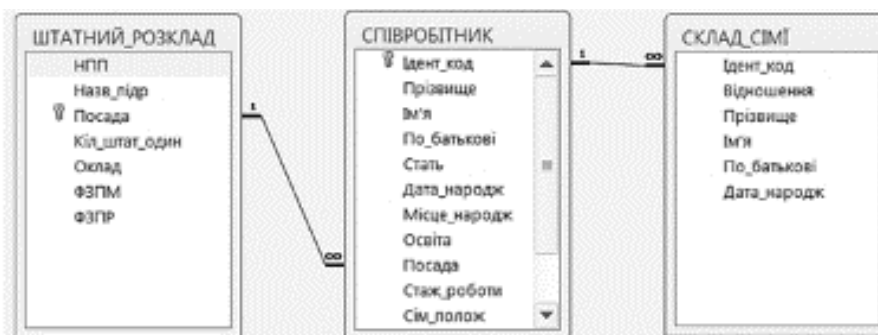


Рисунок 2.1 – Схема зв'язків даних

Якщо зв'язок Один-до-багатьох не створився, треба перевірити вірність створення структур та заповнення відповідних таблиць.

Коли всі зв'язки створені та відповідають зображенню на рис. 2.1, поле **Схема даних** можна закрити.

Контрольні запитання

1. Що таке реляційні бази даних?
2. Навіщо зв'язують таблиці?
3. Назвіть види зв'язків між таблицями та надайте характеристику кожному з них.

4. Як відкрити Схему даних?
5. Як можна змінити зв'язки між таблицями?
6. Як відкрити таблицю для заповнення даними?

ПРАКТИЧНА РОБОТА № 3 РОБОТА ІЗ QBE-ЗАПИТАМИ

Мета роботи: отримання навичок роботи по створенню запитів та дослідження можливостей по їх використанню.

Теоретичні відомості

Запит (query) – це засіб вибору необхідної інформації з бази даних. Питання, сформовані по відношенню до бази даних, і є запит.

Запит є широко використовуваним засобом обробки даних, які зберігаються в таблицях Access. За допомогою запитів можна переглядати, аналізувати і змінювати дані з декількох таблиць. Вони також використовуються як джерело даних для форм і звітів. Запити дозволяють обчислювати підсумкові значення і виводити їх в компактному форматі, подібному формату електронної таблиці, а також виконувати обчислення над групами записів.

Створити запити можна двома способами: **за зразком** (*QBE – Query by example*) та **структурованою мовою запитів** (*SQL – Structured Query Language*).

QBE – це запити для відшукування необхідної інформації в базі даних. Вони формуються не на спеціальній мові, а шляхом заповнення бланка запиту.

SQL – це запити, які складаються програмістами з послідовності *SQL* - інструкцій. Ці інструкції задають, що треба зробити з вхідним набором даних для генерації вихідного набору.

Всі запити Access будує на основі *SQL*-запитів. Щоб подивитися їх, необхідно в активному вікні проектування запиту виконати команду **Вид** → **SQL**.

QBE-запит можна створювати за допомогою Майстра або в режимі Конструктора. Майстер запитів автоматично виконує осно-

вні дії в залежності від відповідей користувача на поставлені питання.

В *Access* можна створювати такі типи запитів:

- запит на вибірку;
- запит з параметрами (критерій відбору задає користувач, ввівши потрібний параметр при виклику запиту);
- перехресний запит (дозволяє створювати результуючі таблиці на основі результатів розрахунків, отриманих при аналізі групи таблиць);
- запит на зміну (виключення, оновлення і додавання) записів (дозволяє автоматизувати заповнення полів таблиць).

Запит на вибірку використовується найбільш часто. При його виконанні дані, що задовольняють умовам відбору, вибираються з однієї або декількох таблиць і виводяться за певною чергою. Можна також використовувати запит на вибірку, щоб згрупувати записи для обчислення сум, середніх значень, перерахунку та для інших дій.

При створенні запиту необхідно визначити:

- поля в базі даних, за якими буде йти пошук інформації.
- предмет пошуку в базі даних.
- перелік полів в результаті виконання запиту.

Для створення нового запиту потрібно у вікні бази даних вибрати на вкладці **Створення групи Запити**. В ній розташовані команди **Майстер запитів** і **Макет запиту**.

Майстер запитів може створити: **Простий запит**, **Перехресний запит**, **Повторювані записи**, **Незв'язані записи**.

Простий запит – це запит на вибірку з певних полів таблиць або інших запитів (найкращий спосіб створення запиту для початківців користувачів).

Перехресний запит – це різновидність запиту на вибірку, виводить дані в компактному форматі, подібному формату електронної таблиці.

Повторювані записи – це запит на пошук записів, які повторюються, в простій таблиці або в запиті.

Незв'язані записи – запит виявляє записи, яким не відповідає ні один запис у підпорядкованій таблиці.

Конструктор запитів дозволяє самостійно створити будь-який

тип запиту, але цей режим рекомендується користувачам, які вже мають певний досвід створення запитів.

Робочі завдання

Завдання 1. Створити простий запит, в якому будуть відображені наступні дані про співробітника: **Прізвище, Ім'я, По-батькові, Дата народження, Ідентифікаційний код, Назва підрозділу, Посада.** Запиту присвоїти ім'я **СПІВРОБІТНИК Посада.** Запит виконати, використовуючи Майстер запитів.

Порядок виконання.

Вибрати вкладку **Створення**, в групі **Запити** виконати команду **Майстер запитів**. Відкриється вікно **Новий запит**. В цьому вікні в полі вибору типів запитів, вибрати **Майстер простих запитів** і клацнути **ОК**.

В наступному діалоговому вікні **Майстер простих запитів** (рис. 3.1) в списку **Таблиці та запити** вибрати таблицю **Співробітник** та перенести необхідні поля зі списку **Доступні поля** до списку **Вибрані поля**. Для цього виділити по черзі необхідні поля у списку **Доступні поля** і перемістити їх до списку **Доступні поля** клавішею

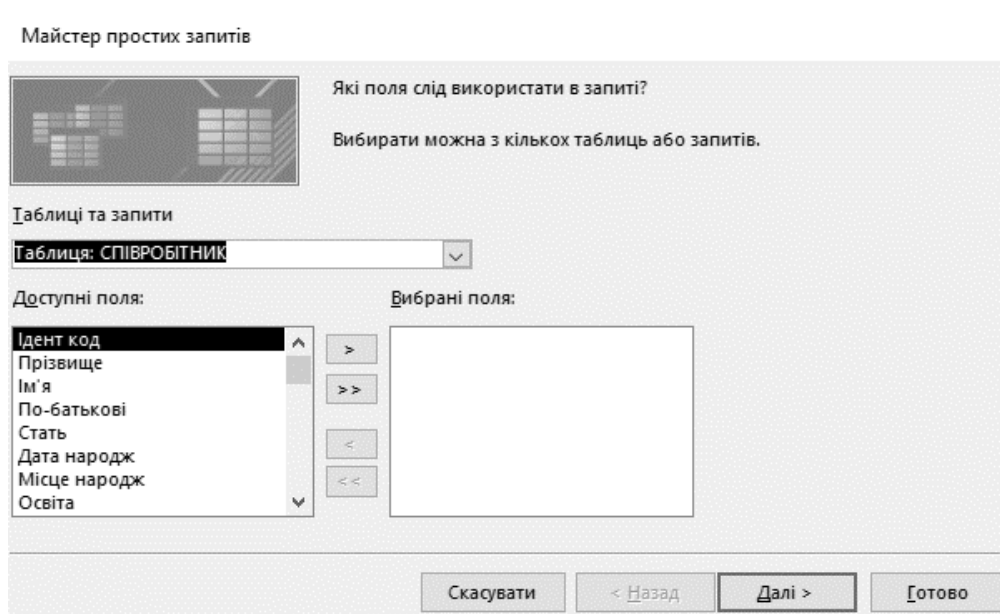



Рисунок 3.1 – Вигляд діалогового вікна **Майстер простих запитів**

Потім в списку **Таблиці і запити** вибрати таблицю **ШТАТ-**

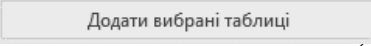

НИЙ РОЗКЛАД та перенести з неї поля, які були відсутні в таблиці **СПВРОБІТНИК**.

По закінченню переміщення, клацаємо кнопку **Далі**. В наступному діалоговому вікні Майстра запитів задаємо ім'я **СПВРОБІТНИК Посада** та клацаємо кнопку **Готово**. У робочому полі з'являється результат запити. Перевіряємо правильність, виконаного запити, закриваємо його кнопкою закрити  в заголовку вікна запити.

Завдання 2. Використовуючи Майстер запитів, самостійно створити простий запит, в якому будуть відображені такі дані про співробітника з таблиці **СПВРОБІТНИК: Прізвище, Посада, Оклад**, з таблиці **СКЛАД СІМ'Ї: Відношення**. Запиту присвоїти ім'я **СПВРОБІТНИК Сім'я**.

Завдання 3. Створити запит на вибірку, що містить поля: **Ідент код, Прізвище, Ім'я, По-батькові, Дата народж**, що відображає список тільки тих співробітників, прізвища яких починаються з букви **К**. Список повинен бути відсортований по даті народження, за зростанням. Запит створити, використовуючи команду **Макет запити**.

Порядок виконання.

На вкладці **Створення** в групі **Запити** вибрати команду **Макет запити**. В діалоговому вікні **Додавання таблиць** вибрати таблицю **СПВРОБІТНИК** та клацнути по кнопці  потім . В робочому полі залишиться вкладка з ім'ям **Запит1**, яка складається з 2-х частин: у верхній частині розміщуються вибрані таблиці або запити, на базі яких будується запит; в нижній частині розташований бланк побудови запити **QBE** (*Query example* – запит за зразком).

Виділити поля **Ідент код, Прізвище, Ім'я, По-батькові, Дата народж** та відбуксувати їх на бланк побудови запити **QBE** в рядок **Поле**.

У рядку **Поле** назви полів розміщуються по стовпцям зліва направо.

У рядку **Ім'я таблиці** відображається ім'я таблиці, з якої вибрано поле.

Рядок **Відображення** дозволяє включити / відключити ви-вод

поля на екран за допомогою прапорця.

Для поля **Дата народж** встановити сортування записів за зростанням. Для цього в стовпці поля **Дата народж** клацнути в рядку **Сортування** та вибрати тип сортування за зростанням зі списку.

Для реалізації в запиті умови вибору співробітників, прізвище яких починаються з букви **К**, в стовпці **Прізвище** в рядку **Критерії** ввести умову **Like "К*"**. Перед введенням літери **К** перейти на український шрифт. Зірочка після букви **К** означає, що після може бути будь-яка послідовність символів. Вікно конструктора запитів буде мати вигляд рис. 3.2.

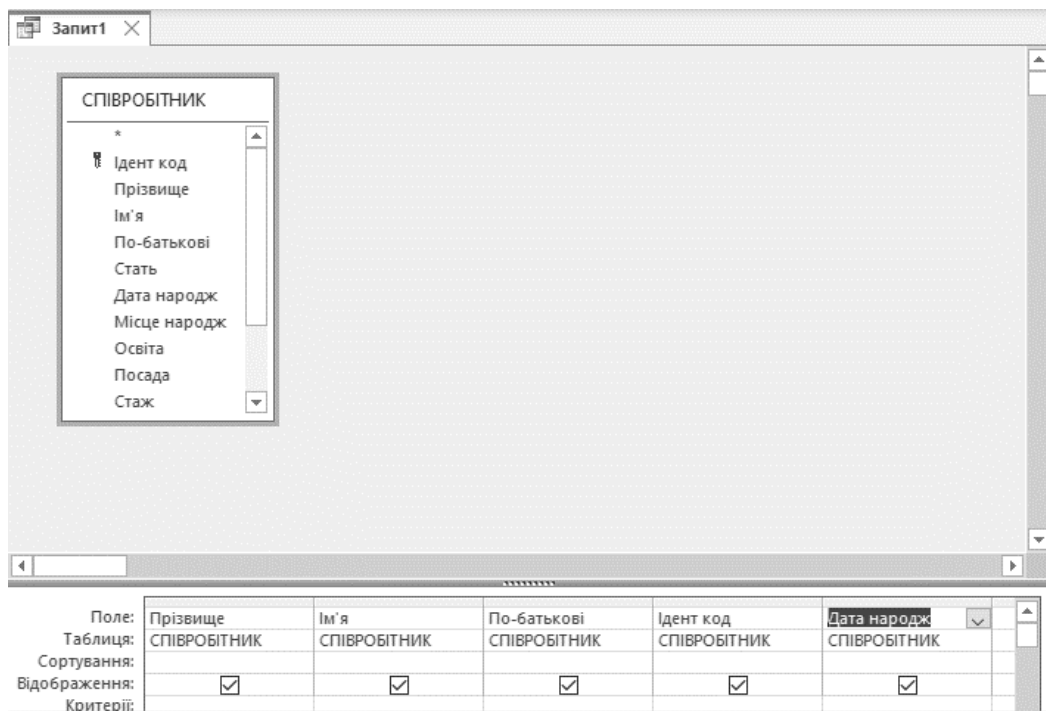



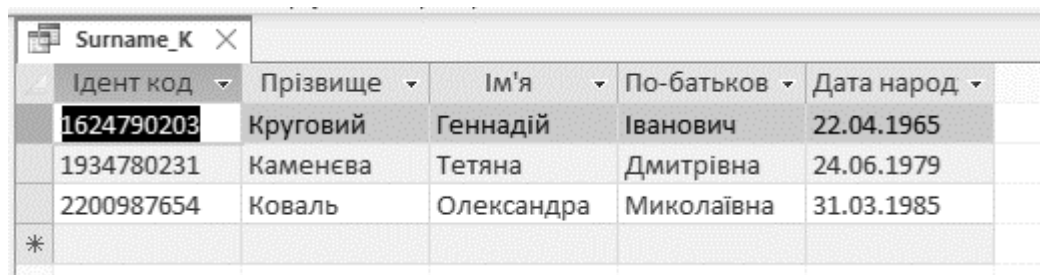
Рисунок 3.2 – Вікно конструктора запиту на вибірку

Закрити вікно конструктора запиту, клацнувши по кнопці **×**, у відповідь на питання зберегти зміни чи ні, натиснути кнопку **Так**, а у вікні **Зберегти як** ввести ім'я запиту **Surname_K** і клацнути **ОК**.

Відкрити запит на вибірку для перегляду. Для цього виділити запит **Surname_K** і клацнути по імені запиту два рази лівою кнопкою миші. На екран виводиться таблиця рис. 3.3, в якій відображаються всі записи з прізвищами, що починаються на букву **К**, записи відсортовані за датою народження по віку.

У випадку, якщо очікуваний результат не отримано, то внести зміни до запит **Surname_K**. Для цього на вкладці **Основне** в групі

Подання клацнути по кнопці **Конструктор** , внести необхідні зміни, зберегти запит, повторити його виконання. Результат виконання запиту **Surname_K** наданий на рис. 3.3.



Ідент код	Прізвище	Ім'я	По-батьков	Дата народ
1624790203	Круговий	Геннадій	Іванович	22.04.1965
1934780231	Каменєва	Тетяна	Дмитрівна	24.06.1979
2200987654	Коваль	Олександра	Миколаївна	31.03.1985

Рисунок 3.3 – Результат виконання запиту **Surname_K**

Завдання 4. Самостійно створити запит за допомогою Конструктора, який містить такі дані: **Ідент код**, **Прізвище**, **Ім'я**, **Посада**, **Освіта**. Запит повинен містити дані про працівників, імена яких починаються на букву **С**. Запиту присвоїти ім'я **Name_C**.

Завдання 5. Самостійно створити запит з відбору працівників які мають вищу освіту. Прізвища працівників розташувати в алфавітному порядку. Створеному запиту присвоїти ім'я **Освіта**.

Завдання 6. Самостійно створити запит на вибірку співробітників, які мають стаж роботи менше зазначеного, наприклад, менше 10, 15, або 20 років. Для створення умови відбору використовувати знак **<**. Дані в запиті розташувати по зменшенню стажу роботи. Запиту присвоїти ім'я **Стаж**.

Завдання 7. Створити параметричний запит, в результаті виконання якого будуть виводитись поля **Прізвище**, **Ім'я**, **По батькові**, **Ідент код** та **Стаж** співробітника, прізвище якого буде вказане у запиті як параметр відбору. Запиту присвоїти ім'я **Ідент код**.

При виконанні параметричного запиту виводиться діалогове вікно із запрошенням ввести параметр для умови відбору записів. Параметрів може бути кілька.

Порядок виконання.

На вкладці **Створення** в групі **Запити** клацнути по кнопці **Макет запиту**.

В діалоговому вікні **Додавання таблиць** вибрати таблицю **СПІВРОБІТНИК**, з якої перемістити поля **Прізвище**, **Ім'я**, **По-батькові**, **Ідент код** та **Стаж** в бланк побудови запиту.

В стовпці **Прізвище** в рядку **Критерії** ввести в квадратних дужках повідомлення, яке буде виводитися на екран при виконанні запиту, а саме: **[Введіть прізвище]**.

Створений запит матиме вигляд, представлений на рис. 3.4.

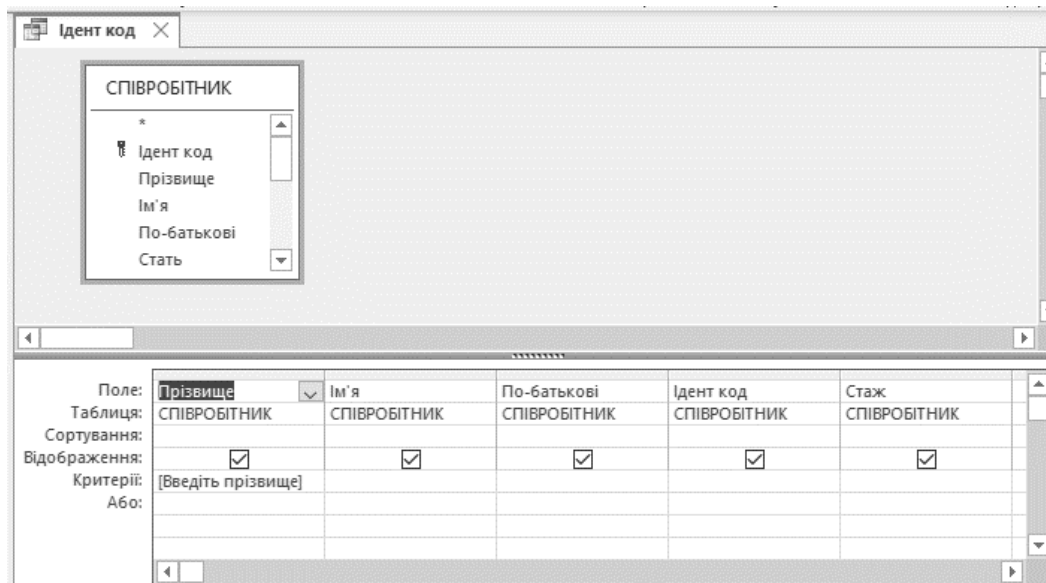


Рисунок 3.4 – Вікно конструктора параметричного запиту

Закрити вікно **Тип запиту:Вибір**, на питання про збереження зміни відповісти – **Так**. У діалоговому вікні **Збереження** ввести ім'я запиту **Ідент код**, клацнути **ОК**.

Виконати запит, двічі клацнувши по запиту **Ідент код**. В діалоговому вікні **Введіть значення параметра** введіть прізвище співробітника, інформацію про якого необхідно отримати, наприклад, *Старченко* і клацнути по кнопці **ОК**. На екрані з'явиться таблиця з даними про обраного співробітника. Завершивши перегляд, закрити вікно.

Завдання 8. Створити запит по вибору співробітників, які мають стаж роботи, наприклад, від 10 до 20 років. У запиті повинні бути відображені наступні параметри: **Ідент код**, **Прізвище**, **Ім'я**, **По-батькові** та **Стаж**. Стаж роботи буде використовуватися в запиті як параметр відбору записів. Прізвища в запиті розташувати за

алфавітом. Запиту присвоїти ім'я **Стаж за параметрами**.

Порядок виконання.

Виконати команду **Створення** → **Запити** → **Макет запиту**. У вікно **Запит1** додати таблицю **СПІВРОБІТНИК**. При закритті вікна **Додавання таблиць** в рядку вкладок з'являється контекстна вкладка **Макет запиту** і активної командою **Вибір** з групи **Тип запиту**.

Перемістити поля **Прізвище**, **Ім'я**, **По-батькові**, **Ідент код** та **Стаж** в бланк побудови запиту. У стовпці **Стаж** в рядку **Критерії** ввести в квадратних дужках повідомлення, які будуть виводитися на екран, і параметри відбору: **>= [Мінімальний стаж] And <= [Максимальний стаж]**. Створений запит матиме вигляд, представлений на рис. 3.5.

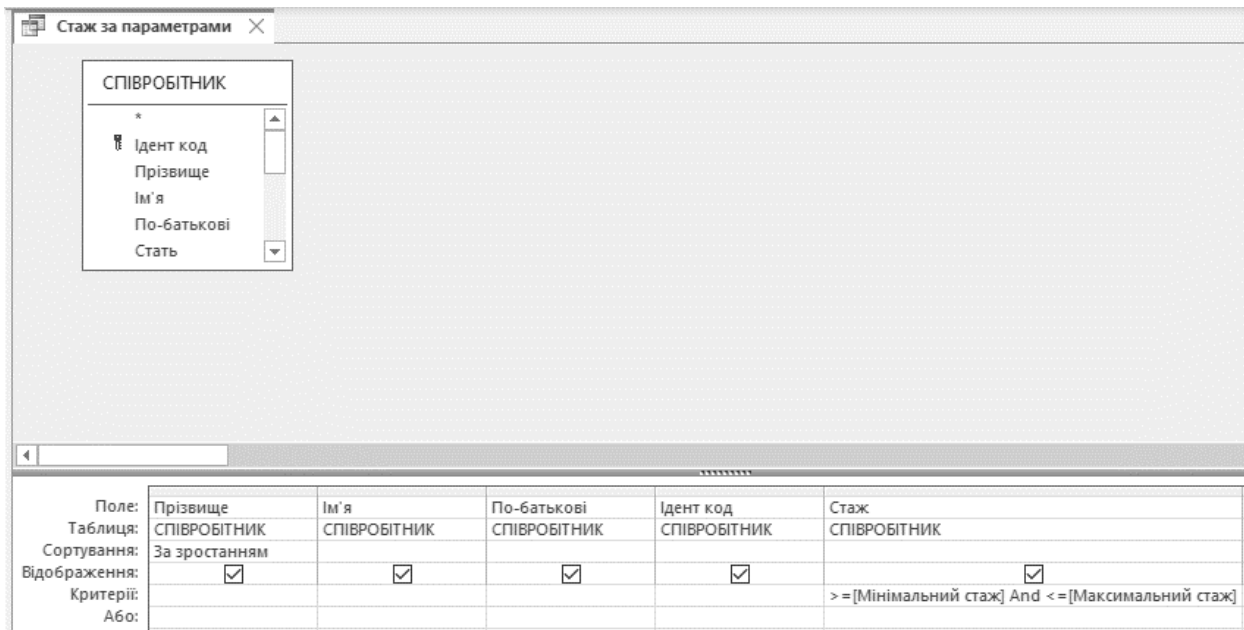


Рисунок 3.5 – Вікно конструктора параметричного запиту на вибір інтервалу стажу роботи співробітників

Зберегти запит з ім'ям **Стаж за параметрами**. Виконати запит.

Завдання 9. Самостійно створити запит щодо вибору співробітників залежно від статі. Наприклад, вивести дані тільки про жінок або тільки про чоловіків. Запиту присвоїти ім'я **Відбір по статі**.

Завдання 10. Скласти запит для створення складного звіту, в якому повинна відображатися інформація про співробітника (прізвище, ім'я, по батькові, посада та назва підрозділу, де він працює), а також визначається кількість членів його сім'ї.

Порядок виконання.

Виконати команду **Створення** → **Запити** → **Конструктор запитів**. У робочому полі відкривається вікно **Запит1** і активізується вікно **Додавання таблиці**, в якому вибрати таблиці **СПІВРОБІТНИК**, **СКЛАД СІМ'Ї**, **ШТАТНИЙ РОЗКЛАД**, після чого закрити вікно **Додавання таблиці**.

З таблиці **СПІВРОБІТНИК** помістити на бланк запиту поля **Прізвище**, **Ім'я**, **По-батькові**, **Посада**.

З таблиці **СКЛАД СІМ'Ї** вибрати поле **Ідент код**, помістити його на бланк запиту.

З таблиці **ШТАТНИЙ РОЗКЛАД** вибрати поля **НП** і **Оклад** і також помістити їх на бланк.

Додати до бланку розділ **Групова операція (Робота із записами)** → **Конструктор** → **Показати або сховати** → **Підсумки**).

У стовпці **Ідент код** клацнути мишею в рядку **Групова операція**, з'явиться кнопка розкриття списку. Клацнути по ній мишею і вибрати зі списку функцію **Count** (кількість записів з однаковими значеннями для даного поля). Вікно запиту матиме вигляд, представлений на рис. 3.6.

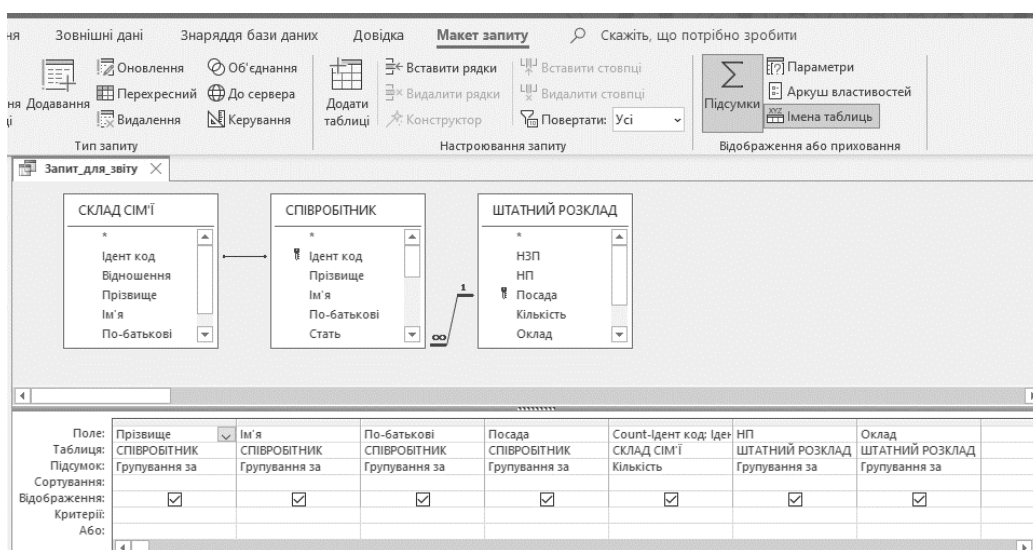


Рисунок 3.6 – Вікно запиту для звіту в режимі **Конструктор**

Змінити назву поля **Ідент код** на **Кількість членів сім'ї**, для цього виділити поле, виконати команду група **Показати або сховати** → **Аркуш властивостей**, або клацнути ПКМ по полю **Ідент код**, в контекстному меню вибрати **Властивості** і у вікні **Вікно властивостей** на вкладці **Загальні** в полі параметра **Підпис** ввести текст **Кількість членів сім'ї**. Закрити **Вікно властивостей**. Закрити вікно створення запиту, виконавши його збереження під ім'ям **Запит для звіту**.

Завдання 11. Створити складний запит за допомогою майстра, в результаті виконання якого будуть отримані відомості про співробітників, які не мають родичів. У звіт включити поля, які містять **ідентифікаційний код, прізвище, ім'я, по-батькові** співробітника, а також його **дату народження**.

Порядок виконання.

Виконати команду **Створення** → **Запити** → **Майстер Запитів**. У діалоговому вікні **Новий Запит** вибирати **Майстер пошуку незв'язаних записів** → **ОК** (рис. 3.7).

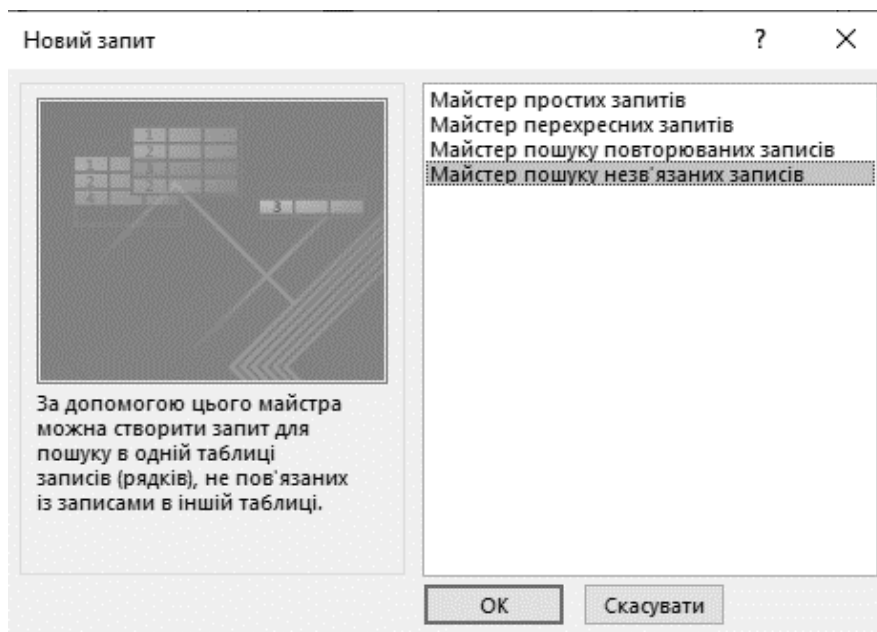
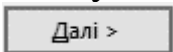


Рисунок 3.7 – Діалогове вікно **Новий Запит**

Відкриється вікно **Майстер пошуку незв'язаних записів**, в якому вибрати таблицю **СПІВРОБІТНИК** і натиснути кнопку



У наступному вікні вибрати таблицю, яка містить підпорядковані записи. Це таблиця **СКЛАД СІМ'Ї**. Натиснути кнопку

У наступному вікні майстра перевірити, що таблиці **СПІВРОБІТНИК** і **СКЛАД СІМ'Ї** пов'язані по полю **Ідент код** як на рис. 3.8.

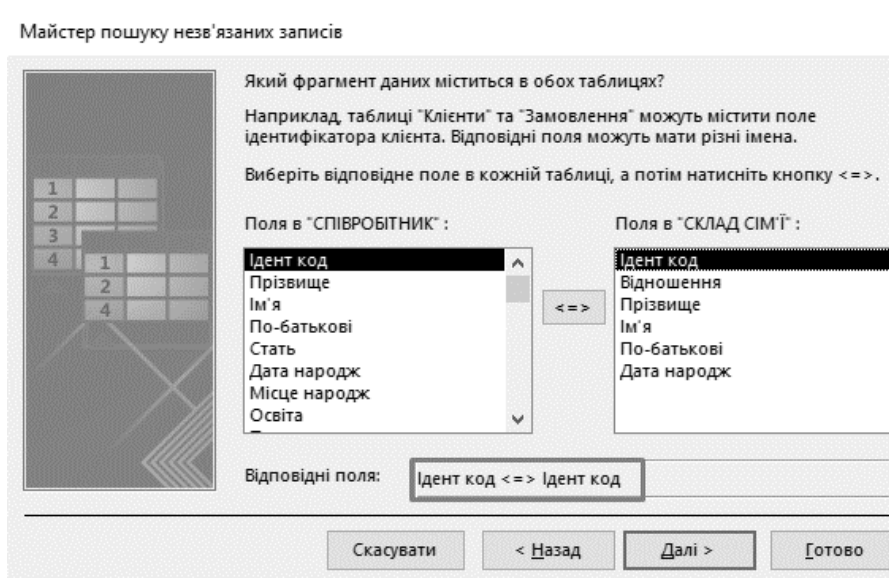


Рисунок 3.8 – Визначення зв'язків між таблицями

Якщо це не так, в кожному списку полів обох таблиць виділити поле **Ідент код** і клацнути на кнопку , що розташована між списками. Натиснути кнопку

В наступному вікні поля **Ідент код**, **Прізвище**, **Ім'я**, **По-батькові**, **Дата народж**, які повинні відображатися в звіті, перенести з лівого списку до правого кнопкою . Натиснути кнопку

На екрані з'явиться запит з необхідними полями з даними про співробітників без родичів. Створенному запиту надати ім'я **Без родичів**.

Контрольні запитання

1. Яке призначення запитів в *Access*?
2. Які типи запитів в *Access* вам відомі?
3. Викладіть послідовність створення нового запиту в режимі

Конструктор.

4. Навіщо створюються запити на вибірку?
5. Як виконати створений запит?
6. Як і навіщо задаються умови вибору при створенні запитів?
7. Яке призначення параметричного запиту?
8. Як набрати необхідні поля в бланк побудови запиту?
9. Як створити запит за допомогою Майстра?

ПРАКТИЧНА РОБОТА № 4 СТВОРЕННЯ SQL-ЗАПИТІВ

Мета роботи: вивчення основних понять *SQL*, набуття навичок по створенню *SQL*-запитів та дослідження можливостей використання *SQL* в *Microsoft Access*.

Теоретичні відомості

SQL (*Structured Query Language* - мова структурованих запитів) – стандартна мова, створена для роботи з реляційними базами даних і призначена для створення баз даних, додаванням нових і підтримки наявних даних, а також отримання потрібної інформації за допомогою спеціальних операторів .

Мова *SQL* складається з трьох основних компонентів: мови визначення даних (*DLL*), мови маніпулювання даними (*DML*) і мови управління даними (*DCL*). У *Microsoft Access* використовується тільки компонент маніпулювання даними (*DML*).

Мова маніпулювання даними (*DML*) призначена для підтримки бази даних. За допомогою цієї мови можна створювати запити на додавання, зміну, видалення або вилучення даних, що знаходяться в базі. У запитах на мові *DML* можна використовувати оператори *INSERT*, *UPDATE*, *DELETE* і *SELECT*.

Оператор *INSERT* дозволяє створити запит на додавання записів (даних) в таблицю. За допомогою оператора *INSERT* можна додати в таблицю новий запис:

INSERT INTO ім'я_таблиці (список_полів) VALUES ('список_значень');

або скопіювати записи з іншої таблиці:

```
INSERT INTO ім'я_таблиці1 (список_полів1) SELECT список_полів2 FROM ім'я_таблиці2;
```

де **ім'я_таблиці1** – назва таблиці, в яку додаються дані; **список_полів1** – назви полів, в які додаються дані; **ім'я_таблиці2** назва таблиці, з якої копіюються дані; **список_полів2** назви полів, з яких копіюються дані.

За допомогою оператора **UPDATE** створюється запит на зміну даних у таблиці. Формат оператора:

```
UPDATE ім'я_таблиці SET ім'я_поля = 'значення';
```

Оператор **DELETE** призначений для створення запиту на видалення записів з таблиці. Формат оператора для видалення усіх записів:

```
DELETE * FROM ім'я_таблиці;
```

де ***** – це символ-маска, що має значення «усі».

Оператор **SELECT** використовується для створення запиту на вибірку даних з бази. Щоб вибрати всі дані, що зберігаються у всіх рядках певної таблиці, використовується оператор **SELECT** наступного формату:

```
SELECT * FROM ім'я_таблиці;
```

Якщо необхідно вибрати рядки таблиці, що задовольняють деякій умові, формат оператора вилучення даних буде мати наступний вигляд:

```
SELECT * FROM ім'я_таблиці WHERE умова;
```

Умови, що визначаються в реченні **WHERE**, називаються предикатами. Якщо твердження, задане предикатом, істинно, умова вважається виконаною. У SQL використовуються наступні предика-

ти порівняння (табл. 4.1).

Таблиця 4.1 – Основні предикати порівняння мови *SQL*

Позначення предикатів	Значення предикатів
=	Дорівнює
<>	Не дорівнює
<	Менше
<=	Менше або дорівнює
>	Більше
>=	Більше або дорівнює
BETWEEN	Задає межі діапазону зміни параметра: WHERE параметр BETWEEN нижня_границя AND верхня_границя
IN [NOT IN]	Перевіряє значення параметра на збіг (не збіг обставин) зі значеннями зі списку: WHERE параметр IN [NOT IN] (список значень)
LIKE [NOT LIKE]	Визначає частковий збіг (не збіг обставин) параметра із заданим значенням: WHERE параметр LIKE [NOT LIKE] значення
NULL	Виконує пошук рядків, в яких параметр має пусте значення: WHERE параметр IS NULL
ALL	Все
SOME	деякі
ANY	Будь-які

Умова відбору може бути складеною, тобто включає в себе комбінацію двох і більше умов за допомогою ключових слів **AND** або **OR**. Ключове слово **AND** використовується у випадку, коли необхідно виконання всіх умов відбору одночасно. Ключове слово **OR** використовується у випадку, коли досить виконання хоча б однієї умови відбору.

Щоб скласти запит на мові *SQL* необхідно на вкладці **Створення** клацнути **Макет запиту**, закрити вікно **Додавання таблиць** і на впливаючій вкладці **Макет запиту** → **Конструктор** вибрати **SQL** → **Режим SQL** (рис. 4.1).

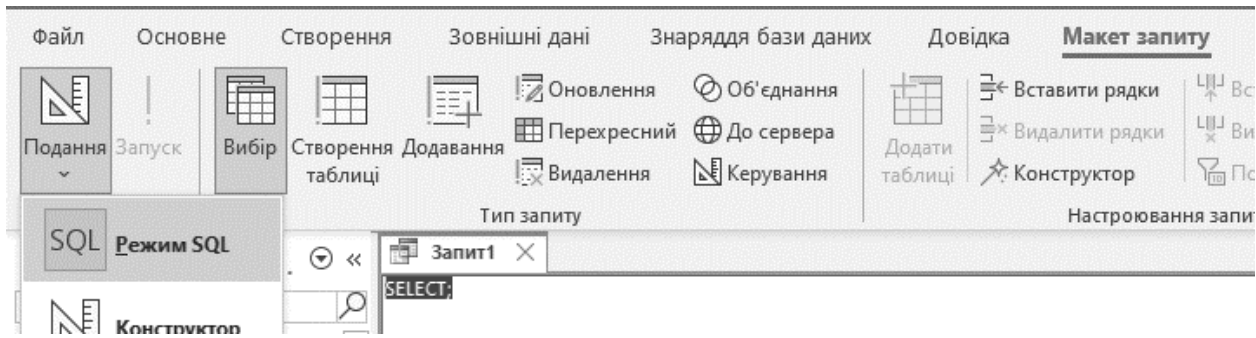


Рисунок 4.1 – Створення запиту в режимі *SQL*

В результаті відкриється вікно **Запит1**, в якому буде прописано службове слово **SELECT**;

Наприклад, необхідно вибрати всі дані таблиці **СПВРОБІТНИК**. У цьому випадку оператор виділення даних буде мати вигляд:

SELECT*FROM СПВРОБІТНИК;

Зірочка в даному прикладі замінює список імен всіх стовпців таблиці **СПВРОБІТНИК**.

У випадку, коли необхідно вибрати всі дані співробітників, прізвища яких починаються з літери **К**, оператор буде мати наступний запис:

SELECT*FROM СПВРОБІТНИК WHERE Прізвище Like 'К*';

Якщо необхідно вибрати всі дані співробітників, прізвища яких починаються з літери **К** зі стажем роботи не менше 10 років, оператор вилучення даних буде мати вигляд:

SELECT*FROM СПВРОБІТНИК WHERE Прізвище Like 'К*' AND Стаж>10;

Якщо необхідно вибрати всі дані співробітників, прізвища яких починаються або з букви **К**, або з букви **С**, оператор вилучення даних буде наступний:

SELECT*FROM СПВРОБІТНИК WHERE Прізвище Like 'К*' OR Прізвище Like 'С*';

Якщо необхідно вибрати дані окремих стовпців таблиці, слід символ * у форматі оператора замінити списком імен стовпців, наприклад:

```
SELECT [Дент код], Прізвище, [Ім'я], [По-батькові], [Дата народж] FROM СПІВРОБІТНИК;
```

У даному прикладі є імена полів, що виділені квадратними дужками. Це необхідний формат тому, що дані поля мають в іменах символи або їх імена складаються з декількох слів.

Для розташування даних в алфавітному порядку (у порядку зростання) використовується оператор **ORDER BY**. Формат оператора:

```
ORDER BY ім'я_поля;
```

Наприклад, необхідно відсортувати дані за датою народження за зростанням:

```
SELECT [Дент код], Прізвище, [Ім'я], [По-батькові], [Дата народж] FROM СПІВРОБІТНИК ORDER BY Прізвище;
```

Для сортування даних в зворотному порядку (в порядку зменшення) в операторі **ORDER BY** використовується ключове слово **DESC**, наприклад:

```
SELECT [Дент код], Прізвище, [Ім'я], [По-батькові], [Дата народж] FROM СПІВРОБІТНИК ORDER BY Прізвище DESC;
```

При складанні запитів можна звертатися до полів з різних таблиць. У такому випадку ім'я поля вказується разом з ім'ям відповідної таблиці. Наприклад, необхідно отримати дані про сім'ї працівників:

```
SELECT СПІВРОБІТНИК.[Дент код], СПІВРОБІТНИК.Прізвище, [СКЛАД СІМ'Ї].Прізвище, [СКЛАД
```

СІМ'Ї].[Ім'я], [СКЛАД СІМ'Ї].[По-батькові], [СКЛАД СІМ'Ї].Відношення FROM СПІВРОБІТНИК, [СКЛАД СІМ'Ї] WHERE СПІВРОБІТНИК.[Ідент код] = [СКЛАД СІМ'Ї].[Ідент код];

У SQL передбачені можливості узагальнення числових даних і отримання за ними підсумків за допомогою спеціальних підсумкових функцій. До підсумкових функцій відносяться:

Функція **COUNT** повідомляє, скільки рядків знаходиться в таблиці або скільки рядків таблиці задовольняють заданим умовам. Формат функції:

SELECT COUNT (ім'я_поля) FROM ім'я_таблиці;

Функція **AVG** підраховує середнє арифметичне всіх значень, що зберігаються в заданому стовпці. Формат функції:

SELECT AVG (ім'я_поля) FROM ім'я_таблиці;

Функція **MAX** визначає максимальне значення в заданому стовпці. Формат функції:

SELECT MAX (ім'я_поля) FROM ім'я_таблиці;

Функція **MIN** визначає мінімальне значення в заданому стовпці. Формат функції:

SELECT MIN (ім'я_поля) FROM ім'я_таблиці;

Функція **SUM** підраховує суму всіх значень заданого стовпчика. Формат функції:

SELECT SUM (ім'я_поля) FROM ім'я_таблиці;

Наприклад, необхідно отримати дані по співробітнику з найбільшим стажем роботи:

SELECT * FROM СПІВРОБІТНИК WHERE Стаж = (SELECT MAX (Стаж) FROM СПІВРОБІТНИК);

Робочі завдання

Завдання 1. Використовуючи мову SQL створити запит, що містить поля: **Ідент код, Прізвище, Ім'я, По-батькові, Дата народж**, що відображає список тільки тих співробітників, прізвища яких починаються з букви **К**. Список повинен бути відсортований

по даті народження по зростанню. Запит зберегти під ім'ям **Завдання1**.

Завдання 2. Використовуючи мову SQL створити запит з відбору працівників, які мають вищу освіту. Прізвища працівників розташувати в алфавітному порядку. Запит зберегти під ім'ям **Завдання2**.

Завдання 3. Використовуючи мову SQL створити запит щодо вибору співробітників, які мають стаж роботи менше 20 років. Дані в запиті розташувати по зменшенню стажу роботи. Запит зберегти під ім'ям **Завдання3**.

Завдання 4. Використовуючи мову SQL створити запит щодо вибору співробітників, які мають стаж роботи від 10 до 20 років. Прізвища співробітників в запиті розташувати за алфавітом. Запит зберегти під ім'ям **Завдання4**.

Завдання 5. Використовуючи мову SQL створити запит, який містить відомості про співробітника з мінімальним стажем роботи. Запит зберегти під ім'ям **Завдання5**.

Завдання 6. Використовуючи мову SQL створити запит, який містить відомості про оклад кожного співробітника. Дані для запиту взяти з таблиць **СПІВРОБІТНИК** і **ШТАТНИЙ РОЗКЛАД**. Дані в запиті розташувати в порядку зменшення значень поля **Оклад**. Запит зберегти під ім'ям **Завдання6**.

Завдання 7. Використовуючи мову SQL самостійно додати в таблицю **СКЛАД СІМ'Ї** дані про нову сім'ю **Коваль Олександрі Миколаївни**. Запит зберегти під ім'ям **Завдання7**.

Завдання 8. Створити порожню таблицю **Співробітник1**, структура якої складається з полів **Ідент код**, **Прізвище**, **Ім'я**, **Побатькові**, **Дата народж**, **Освіта**. Використовуючи мову *SQL* переписати в таблицю **Співробітник1** дані з відповідних полів таблиці **СПІВРОБІТНИК**. Запит зберегти під ім'ям **Завдання8**.

Завдання 9. Використовуючи мову *SQL* видалити з таблиці **Співробітник1** данні про **Кругового Геннадія Івановича**. Запит зберегти під ім'ям **Завдання9**.

Контрольні запитання

1. Що таке мова *SQL*?
2. Які компоненти входять до *SQL*?
3. За допомогою якого оператора проводиться видалення записів з таблиці? Вкажіть його формат запису.
4. Які оператори можна використовувати при складанні запитів на мові *DML*?
5. Вкажіть формат оператора, який дозволяє копіювати записи з іншої таблиці.
6. Який оператор використовується для створінь запитів на вибірку?
7. Що таке предикати?
8. Вкажіть підсумкові функції мови *SQL*. Опишіть їх призначення.
9. За допомогою якого оператора проводиться упорядкування даних в запиті?
10. Опишіть предикати порівняння мови *SQL*.

ПРАКТИЧНА РОБОТА № 5 СТВОРЕННЯ ТА ВИКОРИСТАННЯ ЕКРАННИХ ФОРМ

Мета роботи: дослідити методи та отримати навички роботи по створенню та використанню екранних форм в *MS Access*.

Теоретичні відомості

Зазвичай розробник бази даних створює структуру таблиць і запитів, але заповненням таблиць інформацією він не займається. Для цього є спеціальні кадри, які виконують функції операторів. Для спрощення їхньої праці розроблювач бази готує спеціальні об'єкти – форми.

Форма – це об'єкт бази даних, що являє собою електронний бланк, в якому є поля для введення даних. Користувач вводить дані в ці поля, і дані автоматично заносяться в таблиці бази.

Форми – це варіант подання на екрані комп'ютера одного запису, що дає можливість користувачеві переглядати послідовно відомості з таблиці, здійснювати пошук і бистрий доступ до будь-якого запису, а також здійснювати операції корекції даних в записах і створювати нові записи (поповнювати таблицю). Фактично, форму можна розглядати, як найбільш зручний інтерфейс користувача при роботі з таблицями.

Форми забезпечують самий гнучкий спосіб введення, перегляду і видалення даних і фактично є шаблонами, які управляють відображенням інформації. Більшість форм зазвичай приєднано до однієї або декількох таблиць або запитів бази даних. Джерелом даних, які відображаються в них, є поля в базових таблицях і запитах.

В Access форма є самостійним об'єктом, який зберігається всередині бази. При розробці бази даних можна скористатися різними способами створення форми, які пропонуються на вкладці **Створення** в групі **Форми** панелі *Access* (рис. 5.1).

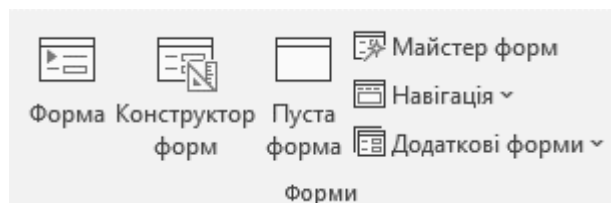


Рисунок 5.1 – Засоби для створення та редагування форм

Макет форми складається з розділів. Будь-яка форма може включати наступні розділи:

- розділ **Заголовок форми** визначає верхню частину форми. В область заголовка форми можна помістити текст, графіку і інші елементи управління. Тут можуть міститися назва форми, написи до полів даних, дата, час тощо. При друку форми, яка складається з багатьох сторінок, розділ заголовка відображається тільки на першій сторінці;
- розділ **Верхній колонтитул** визначає верхній колонтитул сторінки при друці форми. У ньому відображаються назва форми і заголовки стовпців, які виводяться у верхній частині кожної друко-

ваної сторінки. Верхній колонтитул відображається тільки тоді, коли форма відкрита в режимі попереднього перегляду. При друку форми з багатьох сторінок верхній колонтитул відображається вгорі кожної сторінки;

- розділ **Область даних** визначає основну частину форми, що містить дані (записи). Може бути заданий вивід тільки одного запису у вікні форми, або декілька записів. Даний розділ може містити елементи управління, що відображають дані з таблиць і запитів, а також незмінювані дані, наприклад надписи. При друку форми з багатьох сторінок цей розділ відображається на кожній сторінці;

- розділ **Нижній колонтитул** визначає нижній колонтитул сторінки при друці форми. Цей розділ додається у форму разом з розділом, що визначає верхній колонтитул сторінки, і може містити дату, номер сторінки та інші відомості. Він відображається тільки тоді, коли форма відкрита в режимі попереднього перегляду. При друці форми, яка складається з багатьох сторінок, нижній колонтитул відображається внизу кожної сторінки;

- розділ **Примітка форми** визначає нижню частину форми. Цей розділ додається у форму разом з розділом заголовка форми. У ньому відображаються елементи, які залишаються незмінними для кожного запису (кнопки, інструкції по роботі з формою). При друку форми, яка складається з багатьох сторінок, примітка форми буде відображена тільки внизу останньої сторінки.

Призначення і створення елементів керування

На відміну від таблиць і запитів форми складаються з багатьох елементів керування. За допомогою елементів керування можна переглядати дані і працювати з ними в додатку. Найбільш поширеним елементом є текстове поле, крім нього використовуються кнопки, написи, прапорці та елементи керування підлеглої форми або звіту.

Елементи керування можуть бути **пов'язаними, вільними та обчислюваними**.

Пов'язані елементи керування – це елементи керування, джерелом даних яких служить поле таблиці або запиту. Пов'язані елементи керування служать для відображення значень полів бази даних.

Вільні елементи керування – це елементи керування, які не

мають джерела даних. Вільні елементи керування використовуються для виведення на екран написів, малюнків, ліній або прямокутників. Прикладом вільного елемента є напис, який відображає заголовок форми.

Обчислювані елементи керування – це елементи керування, джерелом даних яких є вираз, а не поле.

Вираз – це поєднання арифметичних операторів, імен інших елементів керування, імен полів, функцій і констант, які повертають єдине значення. Наприклад, в наступному виразі заробітна плата з 5% надбавкою співробітнику за бездоганне дотримання техніки безпеки розраховується множенням значення поля **Оклад** на константу (1,05).

= [Оклад] * 1,05;

Елемент керування – об'єкт графічного інтерфейсу користувача, що дозволяє управляти програмами. Елементи керування використовуються для спрощення роботи користувача при відображенні даних, введенні параметрів, виконанні дій.

Панель елементів, що містить кнопки для створення елементів керування на формі (рис. 5.2) можна відкрити таким чином: вкладка **Створення** → **Форми** → **Конструктор форм** → **Конструктор форм** → **Елементи керування** → **Елементи керування**.




Рисунок 5.2 – Панель для створення елементів керування на формі

Розглянемо елементи керування, що використовуються найчастіше.

Щоб зробити форму більш зрозумілою, до неї додають заголовки, підписи чи пояснення. Цей текст є незмінним і створюється за допомогою елемента керування **Підпис**. Щоб додати на форму елемент **Підпис**, треба клацнути на панелі на елемент *Aa*, а потім на формі з використанням лівої кнопки миші намалювати прямокутник. В результаті ми отримуємо текстову рамку, в яку можна вводити довільний текст. При введенні тексту не потрібно піклуватися про його форматування. Неважливо, як він виглядає і де розташований. Закінчивши введення, потрібно натискати клавішу **Enter**, після чого можна приступати до оформлення тексту.

Для форматування елемента керування його потрібно спочатку виділити, клацнувши на ньому мишкою. При виділенні елемента керування навколо нього утворюється рамка з вісьмома маркерами (по кутах і по центрах сторін рамки). Рамку можна розтягувати або стискати методом перетягування кордонів за маркери. При наведенні на маркер покажчика миші він змінює форму, приймаючи зображення двобічної стрілки. У цей момент, натиснувши ліву кнопку миші можна змінювати розмір рамки збільшуючи або зменшуючи в напрямку, вказаному стрілками. Особливу роль відіграє лівий верхній маркер рамки. При наведенні на нього або на кордон рамки покажчик миші приймає форму чотирьох спрямованої стрілки. В цей момент елемент керування можна переміщувати.

Для елементів керування можна також змінювати параметри шрифту, метод вирівнювання тексту та інші елементи форматування. Це виконують звичними засобами форматування, доступними через контекстну вкладку **Формат** → **Шрифт** Якщо клацнути на елементі правою кнопкою миші, відкриється його контекстне меню, в якому є додаткові можливості зміни оформлення.

Для створення елементів керування, які призначені для відображення змісту полів таблиці (пов'язаних елементів керування), служить елемент  (**поле**) на панелі елементів.

При створенні пов'язаного поля разом з ним одночасно утворюється ще один елемент управління – **Приєднаний напис**. Він переміщується разом з полем і утворює з ним єдине ціле. Від'єднати поле від приєданого напису дозволяє маркер, розташований у лівому верхньому куті. При наведенні на нього покажчик миші приймає форму чотирьохспрямованої стрілки. У цей момент пов'язане

поле можна відірвати від приєднаного напису і переміщати окремо. Текстові поля дозволяють вводити довільні значення. Щоб надати користувачеві вибір з певного набору значень, замість текстових полів застосовують такі елементи управління, як прапорці, Кнопка-перемикачі, вимикачі, з'єднані в групи однотипних елементів, а також списки.

Прапорець встановлює або скасовує параметр, пи цьому може бути встановлено/скасовано декілька параметрів або жоден.

Кнопка-перемикач встановлює або скасовує параметр, але, на відміну від прапорця, обов'язково повинен бути встановлений тільки один параметр.

Список може містити фіксований набір значень або певне значення із заданого поля однієї з таблиць. Дозволяє не вводити дані, а вибирати їх зі списку.

Поле зі списком застосовується так само, як і список, але займає менше місця у формі, оскільки список відкривається тільки після клацання на спеціальній кнопці .

Кнопки . З кожною з них можна пов'язати яку-небудь ко-рисну команду, наприклад команду пошуку запису, переходу між записами та інші.

Вкладки дозволяють розмістити багато інформації на обмеже-ній площі. На вкладках розміщують інші елементи управління.

Поле об'єкта OLE. Служить для розміщення зовнішнього об'єкта, відповідного прийнятої в Windows концепції зв'язування та впровадження об'єктів. Об'ктом, як правило, є ілюстрація, напри-клад фотографія, але це може бути і відеозапис, і музикальний фра-гмент, і голосове повідомлення. Існують два типи полів для розмі-щення об'єктів OLE: **Вільна рамка об'єкта** та **Приєднана рам-ка об'єкта** . Вільна рамка не пов'язана ні з яким полем таблиць бази даних. Об'єкт, що знаходиться в ній, виконує, роль ілюстрації і служить для оформлення форми. З приєднаною рамкою пов'язане одне з полів таблиці. В ній відображається вміст цього поля. Цей вміст може змінюватись при переході від одного запису до іншого.

Способи створення форм

Залежно від постановки завдання розробник вибирає варіант створення інтерфейсу користувача для роботи з даними бази. В *Access* форми можна створювати одним з трьох способів (рис. 5.1):

- за допомогою інструментів **автоформи**;
- за допомогою **Майстра форм** (*FormWizard*), який супроводжує розробника в процесі створення форми, ставлячи питання і конструюючи форму відповідно до відповідей;
- за допомогою **Конструктора форм**, де створення форми починається з «чистого аркуша». Крім того в цьому режимі можна редагувати форми, створені в режимах автоформи і майстра форм.

Автоформи застосовуються, якщо в формі мають бути всі поля таблиці, запиту або звіту та якщо не потрібно управляти стилем виведення інформації у формі. В *Access* представлені такі типи автоформ:

- **Форма** (*SimpleForm*) виводить одночасно тільки один запис.
- **Розділена форма** (*SplitForm*) виводить всі записи як таблицю у верхній частині форми і поточний запис в нижній частині.
- Автоформа **Кілька елементів** (*MultipleItems*) відображає всі записи у вигляді «удосконаленої» таблиці. Форму типу Кілька елементів створюють, коли потрібно одночасно бачити всі записи. Зазвичай вона непогано працює з таблицями, що містять невелику кількість стовпців, оскільки кожен стовпець таблиці відображається як стовпець форми.

Іншим простим і швидким способом створення форми є використання **Майстра форм**. За допомогою майстра можна створювати форми на основі однієї таблиці і більш складні форми на основі кількох таблиць і запитів, які мають підлеглі форми. Для створення форми за допомогою майстра форм необхідно вибрати вкладку стрічки **Створення** та у групі **Форми** клацнути кнопку **Майстер форм**.

Майстер форм розбиває процес створення форми на кілька етапів. На кожному етапі потрібно встановити певні параметри в одному з діалогових вікон майстра, кожне з яких визначає один крок створення форми. Якщо на якому-небудь кроці була допущена помилка або виникла необхідність зміни яких-небудь встановлених параметрів, для повернення до попереднього кроку натискається

кнопка **Назад**. Крім того, в будь-який момент можна натискати кнопку **Скасувати** для відмови від створення форми і поверненню до вікна бази даних.

Конструктор форм використовується створення форм вручну. Робота з формами *Access* може відбуватися в чотирьох режимах: у режимі **конструктора**, в режимі **розмітки**, в режимі **форми**, в режимі **таблиці**. Перемикати режими можна за допомогою кнопки **Режими**, розташованої на вкладці **Основне**, або використовуючи контекстне меню форми, клацнувши правою кнопкою миші по ярличку форми і вибравши відповідний режим (рис. 5.3).

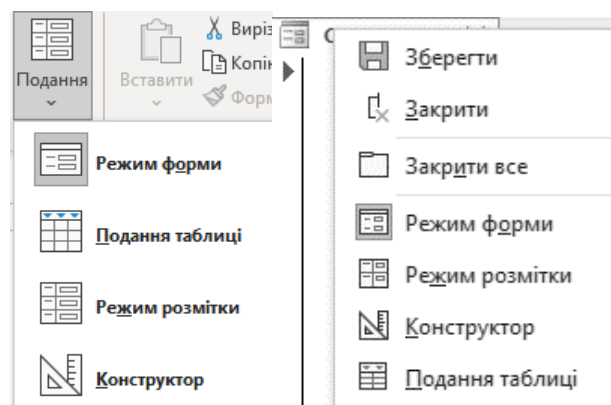


Рисунок 5.3 – Вибір режимів для роботи з формами

Режим конструктора є найпотужнішим і гнучким режимом за можливостями виконання дій. У ньому можна робити все що загодно: змінювати макет, додавати і редагувати елементи керування, змінювати властивості форми.

Режим розмітки є проміжним між режимом конструктора і режимом форми. У ньому форма виводиться в тому вигляді, в якому вона постає перед користувачем, і в той же час в ньому можна налаштувати параметри форми.

Режим форми досить часто використовується в роботі з формами. У цьому режимі форма виводиться в тому вигляді, як її бачить користувач, елементи керування виводять дані, отримані з таблиць, але на відміну від попередніх режимів змінити форму в режимі форми неможливо.

Режим таблиці представляє форму як сітку даних в стилі Excel. Він зазвичай є вбудованим в форму. Використовується в разі необхідності готувати презентацію в стилі *Excel*.

Робочі завдання

Завдання 1. Створити форми для заповнення таблиць: **СПІВРОБІТНИК**, **СКЛАД СІМ'Ї**, **ШТАТНИЙ РОЗКЛАД** за допомогою конструктора форм.

Порядок виконання.

Відкрити базу даних. Для створення форми на вкладці **Створення** в групі **Форми** натиснути кнопку **Конструктор форм**.

У робочому полі з'являється макет вікна форми, з ім'ям **Form1**, а в рядку вкладок додається контекстна вкладка **Інструменти конструктора форм**, яка має три вкладки: **Конструктор форм**, **Упорядкування**, **Формат**. Відкрити вкладку **Конструктор форм**, в ній знайти групу **Знаряддя** і клацнути кнопку **Додавання наявних полів**. З'являється панель **Список полів**, де клацнути посилання **Відображати всі таблиці**. Для створення форми **Співробітник** клацнути по кнопці «+» перед ім'ям таблиці **СПІВРОБІТНИК**. Відкривається перелік полів таблиці **СПІВРОБІТНИК**.

Перенести усі поля таблиці на макет форми в поле **Подробиці**. Перенесення можна виконати двома способами: перетягнути поля таблиці мишею на макет або клацнути два рази по імені поля і **Access** буде переносити поля автоматично, послідовно розташовуючи їх на макеті форми.

Відформатувати поля і розмістити їх на макеті форми в потрібних місцях, за зразком, наведеним на рис. 5.4.

Верхній колонтитул форми			
СПІВРОБІТНИК			
Подробиці			
Ідент код	Ідент код	Стаж	Стаж
Прізвище	Прізвище	Сім полож	Сім полож
Ім'я	Ім'я	Дата зарахуванн	Дата зарахуванн
По-батькові	По-батькові	Тел	Тел
Стать	Стать	Адреса	Адреса
Дата народж	Дата народж		
Місце народж	Місце народж		
Освіта	Освіта		
Посада	Посада		
Нижній колонтитул форми			

Рисунок 5.4 – Макет форми **Співробітник** в режимі **Конструктора**

Створити заголовок форми. Для створення заголовка необхідно відкрити розділ **Заголовок форми**, для чого на вкладці **Конструктор** в групі **Колонтитули** клацнути кнопку **Назва**. На макеті форми з'являться розділи **Верхній колонтитул форми** (вище області даних) та **Нижній колонтитул форми** (нижче області даних). Розділ **Верхній колонтитул форми** має стандартне текстове поле **Form1**. В це поле треба ввести ім'я форми **Співробітник** та розташувати напис по середині поля за допомогою вкладки **Формат**.

Відформатувати форму з наступними параметрами: розміри **Область даних** – 16 см ширина, 9 см висота. Розміри встановлюється шляхом переміщення крайньої правої та нижньої меж відносно горизонтальної і вертикальної лінійок. Якщо лінійка відсутня, клацнути правою кнопкою миші по вільному місці в області даних і виконати команду **Лінійка**. Установіть висоту розділу **Заголовок Форми** 1 см, висоту розділу **Примітка форми** 0,5 см.

Усі текстові написи на формі зробити шрифтом Times New Roman. В заголовку форми розмір шрифту встановити 16 пт, колір тексту ліловий, заливку поля виконати блакитним кольором 2. У назві полів і тексту розмір шрифту 12 пт, колір чорний, заливка назви полів світло-коричнева, заливка полів даних зробити коричневим кольором 2. Усі налаштування можна зробити на вкладці → **Формат**.

Зберегти створену форму під ім'ям **Співробітник**.

В області переходів в розділі форми відкрити форму **Співробітник** та перевірити її структуру. В результаті ми отримаємо форму, як на рис. 5.5.

The screenshot shows a web form titled "СПІВРОБІТНИК" (Employee) with the following fields and values:

Ідент код	1314152347	Стаж	27
Прізвище	Старченко	Сім полож	одружена
Ім'я	Світлана	Дата зарахуванн	24.09.2013
По-батькові	Борисівна	Тел	65-12-13
Стать	ж	Адреса	вул. Гв. Широнінцев, 21, кв.30
Дата народж	22.04.1963		
Місце народж	м.Казань		
Освіта	середня		
Посада	статистик		

At the bottom of the form, there is a status bar with the text: "Запис: 1 з 12", "Без фільтра", and "Пошук".

Рисунок 5.5 – Форма **Співробітник**

Якщо необхідно виправити форму, перейти в режим **Конструктор**, зробити потрібні зміни і знову зберегти форму під тим же ім'ям.

Аналогічним чином самостійно створити форми **Склад сім'ї** та **Штатний розклад**.

Завдання 2. Створити складну форму, що включає дві раніше створені: **Співробітник** і **Склад сім'ї**.

Порядок виконання.

Відкрити форму **Співробітник** в режимі **Конструктор**, клацнувши у правій частині рядка стану кнопку **Конструктор**.

Змінити розташування елементів керування на формі **Співробітник** таким чином, щоб можна було розташувати на цій формі форму **Склад сім'ї** (рис. 5.6).

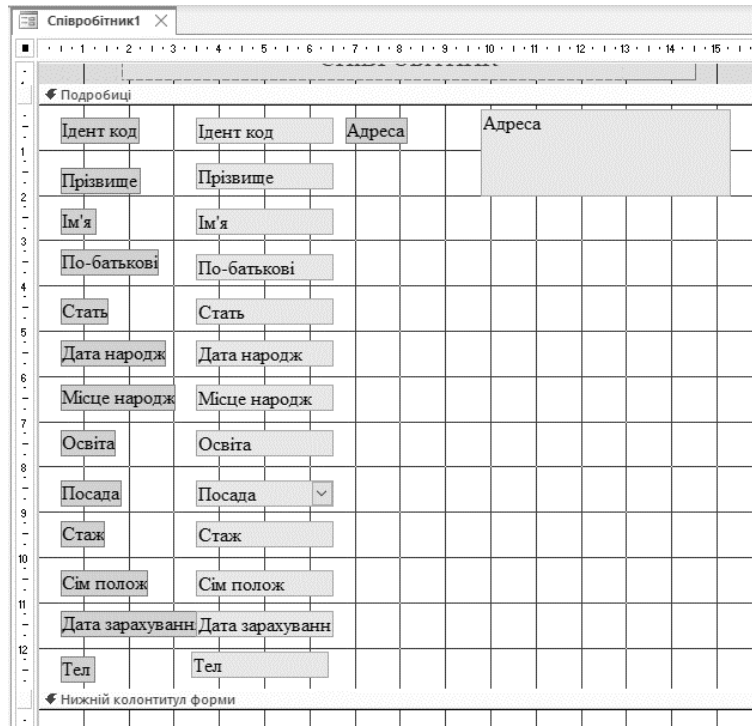


Рисунок 5.6 - Розташування елементів керування для створення складної форми

В **Області переходів** вибрати форму **Склад сім'ї** і перетягнути її у вікно форми **Співробітник** на вільне місце в правій частині форми, як показано на рис. 5.7.

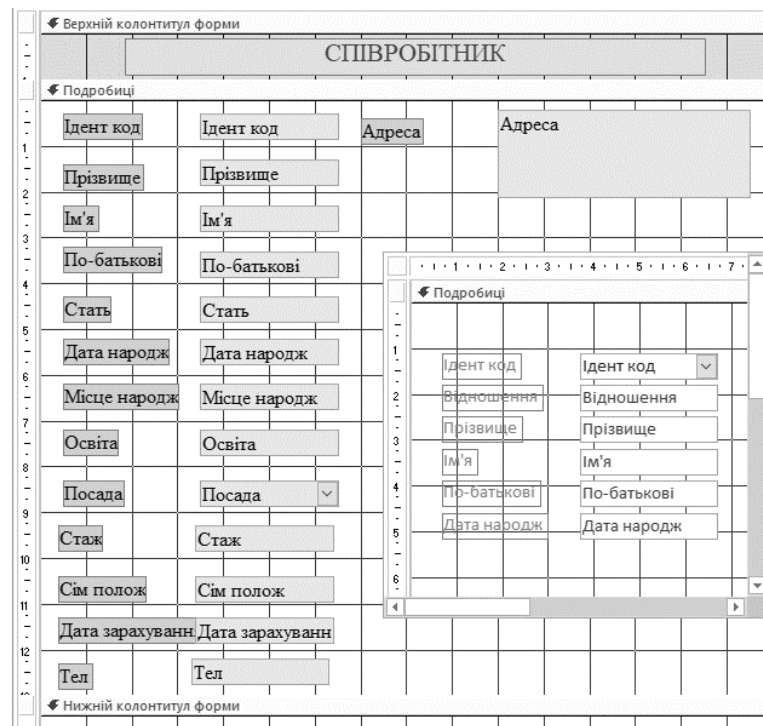


Рисунок 5.7 – Розташування елементів керування складної форми

Зберегти форму **Співробітник** під ім'ям **Співробітник_складна**, використовуючи команду **Файл** → **Зберегти як...** Перевірити роботу форми **Співробітник_складна** в режимі форми.

Завдання 3. Використовуючи форму **Співробітник_складна** доповнити таблиці **СПІВРОБІТНИК** і **СКЛАД СІМ'Ї** даними на нових співробітників:

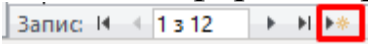
– двох бухгалтерів, жінки, які мають сім'ї, вік від 30 до 40 років, одна має одну дитину, друга двох, мають чоловіків, одна має матір друга батька і матір;

– двох водіїв, чоловіки віком від 25 до 40 років, один одружений має двох дітей і дружину, другий ні (живе з батьком і матір'ю).

– одного методиста, жінка віком до 28 років, заміжня, має сина і чоловіка.

Решту даних придумати самостійно.

Порядок виконання.

Спочатку необхідно створити нові записи в частині основної форми **Співробітник_складна**. Для цього в **Рядку стану** основної частини форми треба клацнути по кнопці **Створити запис**  та ввести дані про нових співробітників. Після закінчення введення всіх нових записів до основної частини форми, натиснути кнопку **Створити запис** в підпорядкованій частині форми для введення даних про членів сімей нових співробітників. Під час заповнення поля **Ідент код** обов'язково користуйтеся списком, що розкривається. Після закінчення зберігти зміни, для чого натиснути кнопку **Зберегти** на рядку швидкого запуску.


Перевірити зміст таблиць **СПІВРОБІТНИК** та **СКЛАД СІМ'Ї**. Таблиця **СПІВРОБІТНИК** повинна збільшитись на **5** записів, а таблиця **СКЛАД СІМ'Ї** – на **15** записів.


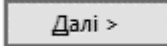
Завдання 4. Самостійно створити складну форму, що містить форми **Співробітник** і **Штатний розклад**. Форму зберегти під ім'ям **Штатний розклад_складна**.

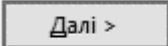
Завдання 5. За допомогою **Майстра форм** створити форму, в якій будуть відображатися дані про співробітників з певним стажем

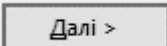
роботи. Форму створити за даними запиту **Стаж за параметрами** і таблиці **СПВРОБІТНИК**. З таблиці **СПВРОБІТНИК** використати дані полів: **Посада, Тел, Адреса**; з запиту **Стаж за параметрами** взяти усі поля.

Порядок виконання.

На вкладці **Створення** в групі **Форми** клацнути кнопку **Майстер форм**. В діалоговому вікні **Створення форм** в полі **Таблиці і запити** вибрати запит **Стаж за параметрами** та перенести усі поля цього запиту зі списку **Доступні поля** до списку **Вибрані поля** за допомогою кнопки . Всі поля будуть поміщені до списку **Вибрані поля**.

Далі в полі **Таблиці і запити** вибрати таблицю **СПВРОБІТНИК** та зі списку **Доступні поля** послідовно перенести поля: **Посада, Тел, Адреса** до списку **Вибрані поля**, використовуючи кнопку , потім натиснути кнопку .

На наступному кроці виберіть зовнішній вигляд форми на ваш розсуд і клацніть .

На наступному кроці в полі **Задайте ім'я Форми** залиште ім'я **Стаж за параметрами** і натисніть кнопку . Перевірити створену форму та зберегти її, виконавши команду **Файл → Зберегти як**.

Контрольні запитання

1. Поясніть призначення форм в *Access*.
2. Що називається автоформою?
3. Які види автоформ вам відомі?
4. Що називається елементом керування?
5. Які елементи керування Вам відомі?
6. З яких розділів складається макет форми?
7. Які існують способи створення форм?
8. Як проводиться форматування елементів керування?
9. У яких режимах користувач може працювати з формою?
10. Що таке складні форми?

ПРАКТИЧНА РОБОТА № 6

СТВОРЕННЯ ЗВІТІВ

Мета роботи: огляд призначення, структурних елементів та дослідження методів створення звітів в *Microsoft Access*.

Теоретичні відомості

Звіт – це гнучкий і ефективний засіб для організації перегляду і отримання підсумкової друкованої інформації. У звіті можна отримати результати складних розрахунків, статистичних порівнянь, а також помістити в нього малюнки та діаграми.

Структура звіту складається з п'яти розділів: заголовка звіту, верхнього колонтитула, області даних, нижнього колонтитула і примітки звіту (рис. 6.1).

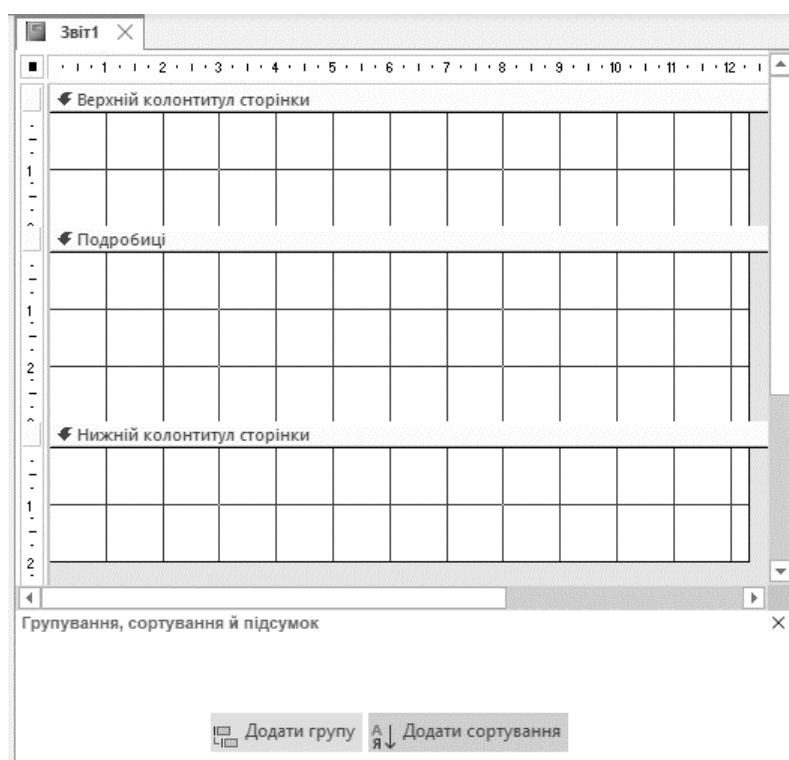


Рисунок 6.1 – Загальна структура вікна звіту

Розділ заголовка служить для друку загального заголовка звіту.

Розділ верхнього колонтитула можна використовувати для друку підзаголовків, якщо звіт має складну структуру і займає багато сторінок. Тут можна також поміщати і колонцифри (номера сто-

рінок), якщо це не зроблено в нижньому колонтитулі.

Область даних. В області даних розміщують елементи керування, пов'язані з вмістом полів таблиць бази. У ці елементи керування видаються дані з таблиць для друку на принтері. Порядок розміщення і вирівнювання елементів керування той же, що і при створенні структури форм.

Розділ нижнього колонтитула використовують для тих же цілей, що і розділ верхнього колонтитула

Розділ примітки використовують для розміщення додаткової інформації.

Не всі звіти містять зазначені розділи. Спочатку необхідно визначитись, які дані необхідно включити в звіт, а потім – які розділи. Наприклад, всі звіти повинні містити розділ даних, але необхідність додавання заголовка або групування залежить від кожного конкретного випадку. Виходячи з цього, вибирають макет звіту.

Макет – це спосіб організації представлення даних у звіті. В *Access* є два базових макета – **табличний** та «**в стовпчик**», - але дані можна впорядкувати практично будь-яким зручним для Вас способом.

1. **Табличний** макет нагадує електронну таблицю. Використовуйте його, коли необхідно представити дані в форматі простого списку.

2. Макет «**в стовпчик**» схожий на форму, яку заповнюють в банках або при здійсненні покупок через Інтернет. Використовується такий тип макетів, якщо звіт містить занадто багато полів для відображення в табличному вигляді.

3. У **змішаних** макетах використовуються елементи і **табличного** макета, і макета «**в стовпчик**». Наприклад, у межах звіту частина полів може розміщуватися вздовж рядка, а інша частина – в стовпчик. У додатку *Access* змішані макети створюються вручну в режимі розмітки або в режимі конструктора.

Визначившись з дизайном можна приступати до створення звіту. Інструменти створення звіту знаходяться на вкладці **Створення** в групі **Звіти** (рис. 6.2).

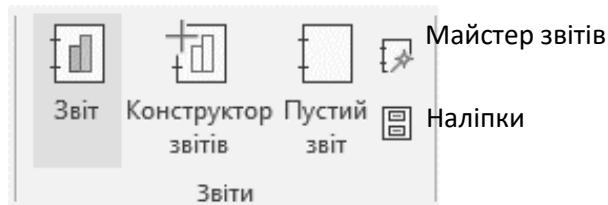


Рисунок 6.2 – Інструменти для створення звітів в *Access*


Інструмент **Звіт** – це найпростіший спосіб створення звіту. Як і інструмент **Форма** (який розглядався у попередній роботі), інструмент **Звіт** автоматично створює звіт, пов'язаний з одним джерелом даних (таблицею або запитом). Для його використання необхідно в області навігації вибрати таблицю або запит, який потрібно використовувати як джерело даних для звіту. На вкладці **Створення** у групі **Звіти** натиснути кнопку **Звіт**. *Access* створить звіт, що включає в себе всі поля джерела даних. У звіті автоматично буде використовуватися табличний макет, і якщо джерело даних містить досить багато полів, *Access* створить звіт з альбомної орієнтацією.

Звіт відкривається в режимі розмітки, в якому можна вносити зміни. Наприклад, можна змінити розмір поля методом перетаскування маркерів елемента керування або переключитися на макет «**в стовпчик**». При цьому з'являється спливаюча контекстна вкладка **Робота з макетами форм**, яка включає вкладки **Конструктор**, **Упорядкування**, **Формат**.


Інструмент **Пустий звіт** дозволяє виконати створення звіту з нуля. Створюючи звіт вручну, з'являється більше можливостей управління: можна перетягувати поля зі списку, розташовувати їх по своєму смаку, а потім групувати. Для запуску режиму **Пустий звіт** необхідно на вкладці **Створення** в групі **Звіти** виконати команду **Пустий звіт**. З'являється спливаюча контекстна вкладка **Робота з макетами звітів**, яка включає вкладки: **Конструктор**, **Упорядкування**, **Формат**, **Параметри сторінки**. При цьому *Access* створить в робочому полі порожній звіт та відобразить область задач **Список полів**. В області задач **Список полів** треба клацнути посилання **Відобразити всі таблиці**. Відкривається список таблиць, які будуть служити джерелом даних, з яких можна перетягнути поля в звіт, або двічі клацнути по іменах полів, які ми хочемо помістити в звіт.

Для порожнього звіту *Access* автоматично застосовує таблич-

ний макет. Якщо необхідно використовувати інший макет, виділіть необхідні поля та на вкладці **Упорядкування** в групі **Таблиця** натисніть кнопку **В стовпчик**. *Access* переупорядкує виділені поля.

Інструмент **Майстер звітів** дозволяє швидко створювати звіти з великою кількістю полів і складною структурою. Для запуску режиму майстра звітів необхідно на вкладці **Створення** в групі **Звіти** натиснути кнопку **Майстер звітів** . Для побудови звіту слід виконати запропоновані майстром дії. Зокрема, вибрати джерело даних, потрібні поля, параметри угруповання, стиль та інші налаштування.

Конструктор дозволяє створити звіт аналогічно створенню екранної форми в режимі конструктора.

Наліпки  створять звіт, відформатований для друку поштових наклейок, наприклад, для автоматичного заповнення поштових конвертів адресами.

Угруповання і сортування даних. Можливості угруповання і сортування даних - одні з найбільш корисних у звіті. Наприклад, якщо потрібно дізнатися, хто з працюючих має стаж роботи більше 25 років, це можна з'ясувати, згрупувавши працюючих за стажем роботи.

Можна групувати як табличні макети, так і макети «в стовпчик». Для цього необхідно відкрити звіт в режимі розмітки, а потім на вкладці **Конструктор** у групі **Груповання та підсумки** натиснути кнопку **Груповання**. Під звітом з'явиться область **Груповання, сортування і підсумок**.

У цій області треба натиснути кнопку **Додати групу**, а потім вибрати поле, по якому потрібно згрупувати дані. *Access* згрупує дані відповідно з вашим вибором. Якщо необхідно виконати сортування даних, натисніть кнопку **Додати сортування** і вибрати необхідне поле. Внесені зміни також відобразяться в режимі розмітки.

У звіт можна додати 10 рівнів угруповань і, при необхідності, можна виконати сортування для кожного з них.

Крім угруповання і сортування даних у звіті можна додавати загальний підсумок і проміжні підсумки, а також інші обчислення. Наприклад, можна вирахувати суму заробітної плати, отриману водіями за місяць або за рік. Загалом, для додавання підсумків необхідно:

1. Для звіту, відкритого в режимі розмітки, відкрити область **Групування, Сортування і підсумок**, виберіть рівень Угрупування, а потім натиснути кнопку **Розгорнути**.

2. Знайти поле «з підсумками» і клацнути стрілку поруч із ним.

3. У діалоговому вікні **Ітоги** вибрати поле, яке потрібно обчислити, і який тип обчислення застосувати (наприклад, сума або середнє значення), а також налаштувати параметри, такі як загальний підсумок і проміжні підсумки.

4. Коли налаштування будуть закінчені, обчислення відобразяться у звіті.

Вносити зміни в звіт можна як в режимі розмітки, так і в режимі конструктора. Режим розмітки найкраще підходить для зміни зовнішнього вигляду звіту. Наприклад, в ньому можна змінити порядок слідування полів (переупорядкувати їх), змінити їх розміри або застосувати готовий стиль.

На відміну від режиму макета, в режимі конструктора можна керувати вмістом звіту. Наприклад, можна додати текстові поля з датою і часом створення звіту.

Робочі завдання

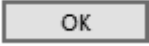
Завдання 1. Створити простий звіт, що містить список співробітників підприємства з полями: **Ідент код, Прізвище, Ім'я, Побатькові, Тел.** Згрупувати дані за першою літерою прізвища. Обчислити кількість прізвищ у кожній групі.

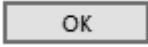
Порядок виконання.

Перш за все, необхідно вибрати тип макета, на основі якого ми будемо будувати звіт. Звіт будемо будувати, використовуючи табличний макет. Як інструмент побудови виберемо інструмент **Пустий Звіт**.

Для створення звіту на вкладці **Створення** в групі **Звіти** клацнути на кнопці **Пустий Звіт**. В робочому полі з'являється вкладка з ім'ям **Звіт1**. Праворуч у діалоговому вікні **Список полів** клацнути посилання **Відобразити всі таблиці**. Відкривається перелік, наявних таблиць. Клацаємо по значку плюс + перед ім'ям таблиці **СПВРОБІТНИК**, відкривається перелік полів цієї таблиці. Клаца-

ємо двічі по іменах тих полів, які необхідно перемістити в звіт: **Идент код, Прізвище, Ім'я, По-батькові, Тел.** *Access* автоматично переносить і розставляє ці поля у вигляді таблиці. В результаті ми отримали заготовку звіту у вигляді таблиці в режимі **Режим розмітки**. Тепер необхідно, додати заголовок і примітку, для цього на контекстній вкладці **Конструктор** в групі **Колонтитули** клацнути кнопку **Назва**.

У робочому полі штриховими лініями позначені розміри сторінки. Для приведення цих параметрів до стандартних розмірів листів, встановити розміри з наступними параметрами: верхнє і нижнє поля по 30 мм, лівє і правє по 21 мм. Для установки цих параметрів на вкладці **Параметри сторінки** в групі **Макет сторінки** клацнути кнопку **Параметри сторінки**. В діалоговому вікні **Параметри сторінки** встановити вище вказані параметри і натиснути кнопку . Перейти в режим **Конструктор** і встановити ширину листа 165 мм шляхом переміщення кордону межі звіту. Перейти в режим **Попередній перегляд** і подивитися загальний вигляд листа. У подальшому можна використовувати цю команду для того, щоб переглянути зміст звіту після внесення будь-яких змін. Знову повернутися в **Режим розмітки**.

Приступити до створення структури заголовка звіту. Треба в заголовку звіту розмістити назву звіту, дату створення і передбачити місце для малюнка. Заголовок звіту представляє окрему таблицю, що складається із комірок. Необхідно об'єднати дві комірки крайнього правого стовпця, для чого на вкладці **Упорядкування** в групі **Об'єднати або розділити** клацнути кнопку **Об'єднати**, а потім на вкладці **Конструктор** в групі **Колонтитули** клацнути **Дата і час**. У діалоговому вікні **Дата і час** вибрати дату з форматом **Дата Місяць Рік** клацнути . Обрана дата буде введена в останню праву комірку. Перемістити цю комірку під рядок **Звіт1**, для чого скористатися командами **Вирізати** і **Вставити** контекстного меню. Ввести назву звіту **Список співробітників підприємства**. Поле назви відформатуватитаким чином: шрифт Times New Roman, розмір 18 пт, колір чорний. Для поля дати встановити шрифт Times New Roman, розмір 14 пт, колір чорний. Для форматування полів скористатися вкладкою **Формат**. В режимі попереднього перегляду звіт повинен виглядатитак, як на рис. 6.3.

Звіт

Список співробітників підприємства

16.01.2024

Ідент код	Прізвище	Ім'я	По-батькові	Тел
1314152347	Старченко	Світлана	Борисівна	65-12-13
1545678990	Архипов	Сергій	Іванович	23-10-70
1624790203	Круговий	Геннадій	Іванович	68-14-13
1748576413	Царьова	Анна	Миколаївна	47-23-15
1934780231	Каменєва	Тетяна	Дмитрівна	65-67-72
1955443781	Безродний	Володимир	Михайлович	32-32-14
2014654788	Садчиков	Аркадій	Вікторович	10-12-10
2055894321	Бронзов	Станіслав	Іванович	47-15-20
2178943214	Мапошенко	Юрій	Миколайович	43-35-13
2200987654	Коваль	Олександра	Миколаївна	47-67-33
2233668943	Строков	Олег	Вікторович	96-05-03
2314743296	Бородулін	Андрій	Васильович	27-14-12

Рисунок 6.3 – Вид макета звіту

Відповідно до завдання сортування і угруповання полів необхідно провести по полю **Прізвище**. Для цього на вкладці **Конструктор** в групі **Групкування та підсумки** клацнути кнопку **Групкування**. В діалоговому вікні **Групкування, сортування і підсумок** клацнути **Додати групу**. Відкривається перелік команд з уточнюючими діями. При необхідності стрічку з уточнюючими командами можна відкрити клацанням по команді **Розгорнути**. Для параметра **Групувати за** вибираємо **Прізвище**. Далі встановлюємо **спочатку А** та **за першим символом**. Для параметра з підсумками вибираємо **З підсумками для** – **Прізвище**. Для параметра з назвою в область вводу вводимо текст **По-батькові** та встановлюємо параметри тексту: *Times New Roman*, 14 пт, чорний за допомогою кнопки **Шрифт...**. Далі вибираємо: **з розділом верхнього колонтитула, з розділом нижнього колонтитула, відобразити верхній колонтитул і перший запис на одній сторінці** (рис. 6.4). Після встановлення параметрів закриваємо діалогове вікно **Групкування, сортування і підсумок**.

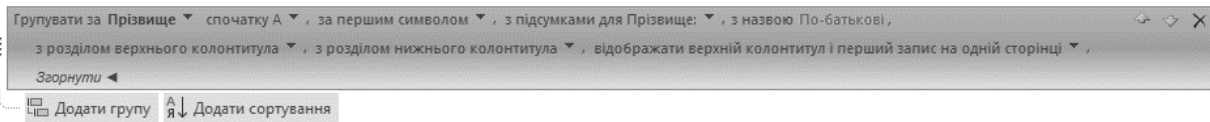


Рисунок 6.4 – Встановлення параметрів для виконання угруповання, сортування та підсумків

В розділі примітки звіту обчислювальне поле перенести в праву частину розділу, а по центру розділу додати напис *Кількість в групі*. Напис виконати шрифтом Times New Roman, розмір 14 пт, колір чорний.

До розділу заголовка додати логотип підприємства. Для цього знайти в інтернеті відповідний малюнок та зберегти його на Вашому комп'ютері. Потім на вкладці **Конструктор** відкрити панель **Елементи керування**, де клацнути елемент **Рисунок** . Намалювати рамку для логотипу ліворуч від назви звіту та за допомогою діалогового вікна **Выбор рисунка** додати збережений логотип до заголовку звіту.

Переглянути звіт в режимі попереднього перегляду. Шляхом переміщення та об'єднання полів, а також за допомогою зміни їх розмірів отримати звіт як на рис. 6.5.

Список співробітників підприємства				16.01.2024
Ідент код	Ім'я	По-батькові	Тел	
Група прізвищ, які починаються на букву <input type="text" value="А"/>				
<input type="text" value="Архипов"/>	<input type="text" value="1545678990"/>	<input type="text" value="Сергій"/>	<input type="text" value="Іванович"/>	<input type="text" value="23-10-70"/>
Кількість в групі				<input type="text" value="1"/>
Група прізвищ, які починаються на букву <input type="text" value="Б"/>				
<input type="text" value="Бородулін"/>	<input type="text" value="2314743296"/>	<input type="text" value="Андрій"/>	<input type="text" value="Васильович"/>	<input type="text" value="27-14-12"/>
<input type="text" value="Бронзов"/>	<input type="text" value="2055894321"/>	<input type="text" value="Станіслав"/>	<input type="text" value="Іванович"/>	<input type="text" value="47-15-20"/>
<input type="text" value="Безродний"/>	<input type="text" value="1955443781"/>	<input type="text" value="Володимир"/>	<input type="text" value="Михайлович"/>	<input type="text" value="32-32-14"/>
Кількість в групі				<input type="text" value="3"/>

Рисунок 6.5 – Фрагмент оформлення першої сторінки звіту з використанням інструменту **Пустий звіт**

Звіт зберегти під ім'ям **Простий звіт**.

Завдання 2. Створити складний звіт, використовуючи Інструмент **Звіт**, який повинен містити:

- Список співробітників по підрозділах, які мають в складі сім'ї більше однієї людини;
- Сумарний оклад і середній по підрозділах;
- Сумарний оклад по всьому підприємству.

Звіт побудувати на основі запиту **Запит для звіту**.

Порядок виконання.

Звіт будемо створювати, використовуючи інструмент **Звіт**. Перш за все знайти серед запитів **Запит для звіту** і виділити його. Потім на вкладці **Створення** в групі **Звіти** виконати команду **Звіт**. В результаті в робочому полі з'являється вікно звіту **Запит для звіту**.

Змінити розміри стовпців, для чого виділити всю таблицю та виконати команду **Конструктор** → **Знаряддя** → **Аркуш властивостей**. У вікні властивостей встановити параметри: Ширина - 2,7 см; Висота - 0,6 см; Від верхнього краю - 0,1 см; Від лівого краю - 0,1 см. Закрити вікно властивостей.

Розташувати стовпці в наступному порядку: **Прізвище, Ім'я, По батькові Посада, Оклад, Кількість членів сім'ї, НП**. Перейменувати поле **Кількість чл. сім'ї** в **Склад сім'ї**.

Перейти в режим режим **Конструктор** та зменшити розміри полів **Оклад** та **Склад сім'ї** таким чином, що б праве останнє поле, не виходило за межі 18 см. Відформатувати заголовок звіту як показано на рис. 6.6.

Верхній колонтитул звіту						
Запит для звіту "Кількість членів сім'ї співробітників"						
Верхній колонтитул сторінки						
Прізвище	Ім'я	По-батькові	Посада	Оклад	Кількість членів сім'ї	НП
Подробиці						
Прізвище	Ім'я	По-батькові	Посада	Оклад	Count-Ідент код	НП
Нижній колонтитул сторінки						
="Сторінка " & [Page] & " з" & [Pages]						
Нижній колонтитул звіту						
=Кількість(*)						

Рисунок 6.6 – Вид заготовки звіту після встановлення параметрів елементів

Для групування та сортування даних по підрозділам на вкладці **Конструктор** в групі **Групування та підсумки** виконати команду **Групування**. У вікні **Групування, сортування і підсумок** клацнути **Додати групу** і встановити наступні параметри: *з назвою НП; спочатку з А; за повним значенням; з підсумками: Прізвище; Тип Кількість записів, Відобразити загальний підсумок, Відображати підсумки в нижньому колонтитулi; з заголовком Назва підрозділу, з розділом верхнього колонтитула, з розділом нижнього колонтитула; відобразити верхній колонтитул і перший запис на одній сторінці*. На рис. 6.7 та 6.8 показані параметри вікна **Групування, сортування і підсумок**.

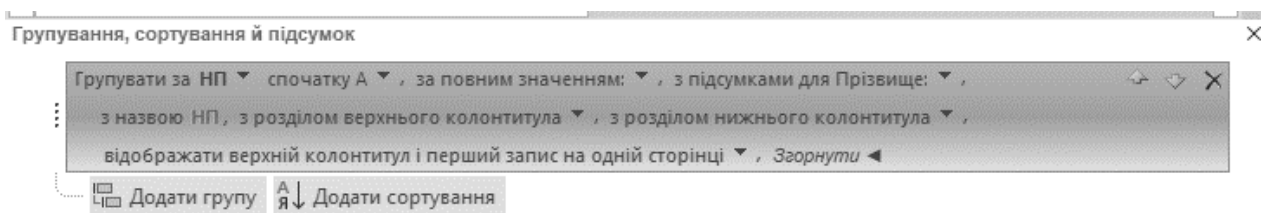


Рисунок 6.7 – Діалогове вікно **Групування, Сортування і підсумок** з встановленими параметрами

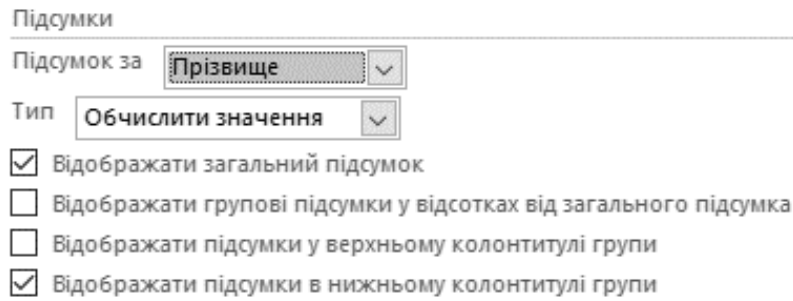


Рисунок 6.8 – Діалогове вікно **Підсумки** з встановленими параметрами

Закрити вікно **Групування, Сортування і підсумок** після встановлення зазначених вище параметрів.

Перейти в Режим розмітки та зробити наступні налаштування:

- в першому стовпці клітинку **НП** розмістити під заголовком стовпця **Прізвище** та змінити в ній напис на **Назва підрозділу**;
- клітинку **дирекція** перемістити під заголовок стовпця **Ім'я**;
- видалити останній стовець;

- клітинку 4 (розрахунок кількості працівників у підрозділі) та клітинку 9 (розрахунок загальної кількості працівників) перенести до стовпця **Склад сім'ї**;

- в рядку з розрахунком загальної кількості працівників в перший стовпець ввести текст: **Загальна кількість працівників, що мають членів сім'ї**.

Перейти в режим попереднього перегляду та порівняти отриманий результат з рис. 6.9.

НП	Прізвище	Ім'я	По-батькові	Посада	Оклад	Кількість членів сім'ї
дирекція						4
	Царьова	Анна	Миколаївна	гол.бухгалтер	4300	1
	Каменєва	Тетяна	Дмитрівна	начальник ВК	1500	1
	Безродний	Володимир	Михайлович	зам.директора	5000	1
	Архипов	Сергій	Іванович	директор	5300	1
сл.перевезень						2
	Строков	Олег	Вікторович	водій	3500	2
	Бронзов	Станіслав	Іванович	водій	3500	1
сл.управління						3
	Старченко	Світлана	Борисівна	статистик	1000	2
	Садчиков	Аркадій	Вікторович	диспетчер	1000	1
	Мапошенко	Юрій	Миколайович	інженер по ТБ	1500	2
Загальна кількість працівників, які мають членів сім'ї						9

Рисунок 6.9 – Звіт в режимі попереднього перегляду після виконання угруповання і сортування

Перейти в режим **Конструктор** та заповнити області приміток групи таким чином:

- розширити область приміток групи для розміщення двох рядків;

- в кожен рядок додати елемент управління **Поле** ;

- у верхньому рядку в ліву частину поля ввести текст **Сумарний оклад по підрозділу** і розмістити її під стовпцем **Прізвище**;

- в праву частину поля верхнього рядка ввести формулу **=Sum([Оклад])** для обчислення сумарного окладу по підрозділу і розмістити її під стовпцем **Оклад**;

- у нижньому рядку в ліву частину поля ввести текст **Середній оклад по підрозділу** і розмістити її під стовпцем **Прізвище**;
- в праву частину поля нижнього рядка ввести формулу **=Avg([Оклад])** розмістити її під стовпцем **Оклад**.

Аналогічно створити рядки для виведення сумарного окладу по підприємству та середнього окладу по підприємству в розділі **Примітка звіту**.

Як заголовок звіту ввести текст **Список співробітників підрозділу із сім'ями**.

Додати номери сторінок командою **Конструктор** → **Колонтитули** → **Номери сторінок**. У вікні **Номери сторінок** вибрати формат номера, встановити Кнопка-перемикач **Нижній колонтитул** і вибрати вирівнювання по центру.

Перейти в режим попереднього перегляду та перевірити звіт згідно рис. 6.10.

Прізвище	Ім'я	По-батькові	Посада	Оклад	Склад сім'ї
Назва підрозділу					4
Царьова	Анна	Миколаївна	гол.бухгалтер	4300	1
Каменева	Тетяна	Дмитрівна	начальник ВК	1500	1
Безродний	Володимир	Михайлович	зам.директора	5000	1
Аркилов	Сергій	Іванович	директор	5300	1
Сумарний оклад по підрозділу				16100	
Середній оклад підрозділу				4025	
Назва підрозділу					2
Стронов	Олег	Вікторович	водій	3500	2
Бронзов	Станіслав	Іванович	водій	3500	1
Сумарний оклад по підрозділу				7000	
Середній оклад підрозділу				3500	
Назва підрозділу					3
Старченко	Світлана	Борисівна	статистик	1000	2
Садчиков	Аркадій	Вікторович	диспетчер	1000	1
Малощенко	Юрій	Миколайович	інженер по ТБ	1500	2
Сумарний оклад по підрозділу				3500	
Середній оклад підрозділу				1166,666666667	
Загальна кількість працівників, які мають членів сім'ї					9
Сумарний оклад по підприємству				26600	
Середній оклад по підприємству				2955,555555556	

Рисунок 6.10 - Звіт в режимі попереднього перегляду в остаточному вигляді

Звіт зберегти під ім'ям **Складний звіт**.

Контрольні запитання

1. Що називається звітом в *Microsoft Access*?
2. Що таке структура звіту?
3. Що таке макет звіту?
4. Які типи макетів використовуються при створенні звітів в *Access*?
5. Назвіть способи створення звітів.
6. Опишіть процес створення звіту за допомогою **Порожній звіт**.
7. Як створити звіт за допомогою інструменту **Звіт**?
8. Назвіть, які режими можна використовувати при створенні звіту?
9. Як виконується сортування даних у звіті?
10. Опишіть призначення основних розділів звіту.
11. Що таке форматування звіту?

ПРАКТИЧНА РОБОТА № 7 СТВОРЕННЯ ГОЛОВНОЇ КНОПКОВОЇ ФОРМИ

Мета роботи: дослідження та придбання навичок по створенню та налаштуванню компонентів кнопкової форми інтерфейсу користувача БД в *MS Access*.

Теоретичні відомості

Кнопкові форми призначені для швидкого доступу до форми або звіту. Вони представляють з себе своєрідні провідники по формам і звітам БД. Якщо форм і звітів стає дуже багато, щоб орієнтуватися в них створюють кнопкові форми, які містять тільки кнопки і призначаються для виконання основних дій по обробці бази даних.

Кнопкова форма є не що інше, як меню для роботи в базі даних. У меню може бути декілька вкладених підменю.

На кнопку форму поміщаються кнопки, при натисканні яких відкриваються форми або звіти (або відкриваються інші кнопкові

форми, за допомогою яких відкриваються інші форми або звіти), здійснюється вихід з *Microsoft Access* або змінюється сама кнопкова форма.

Кнопкову форму можна створити за допомогою **Конструктора форм**, а можна використовувати спеціальний майстер, який називається **Диспетчер кнопок форм**. Починаючи з Access 2010, диспетчер кнопок форм недоступний на стрічці, тому спочатку потрібно додати команду на панель швидкого доступу (рис. 7.1).

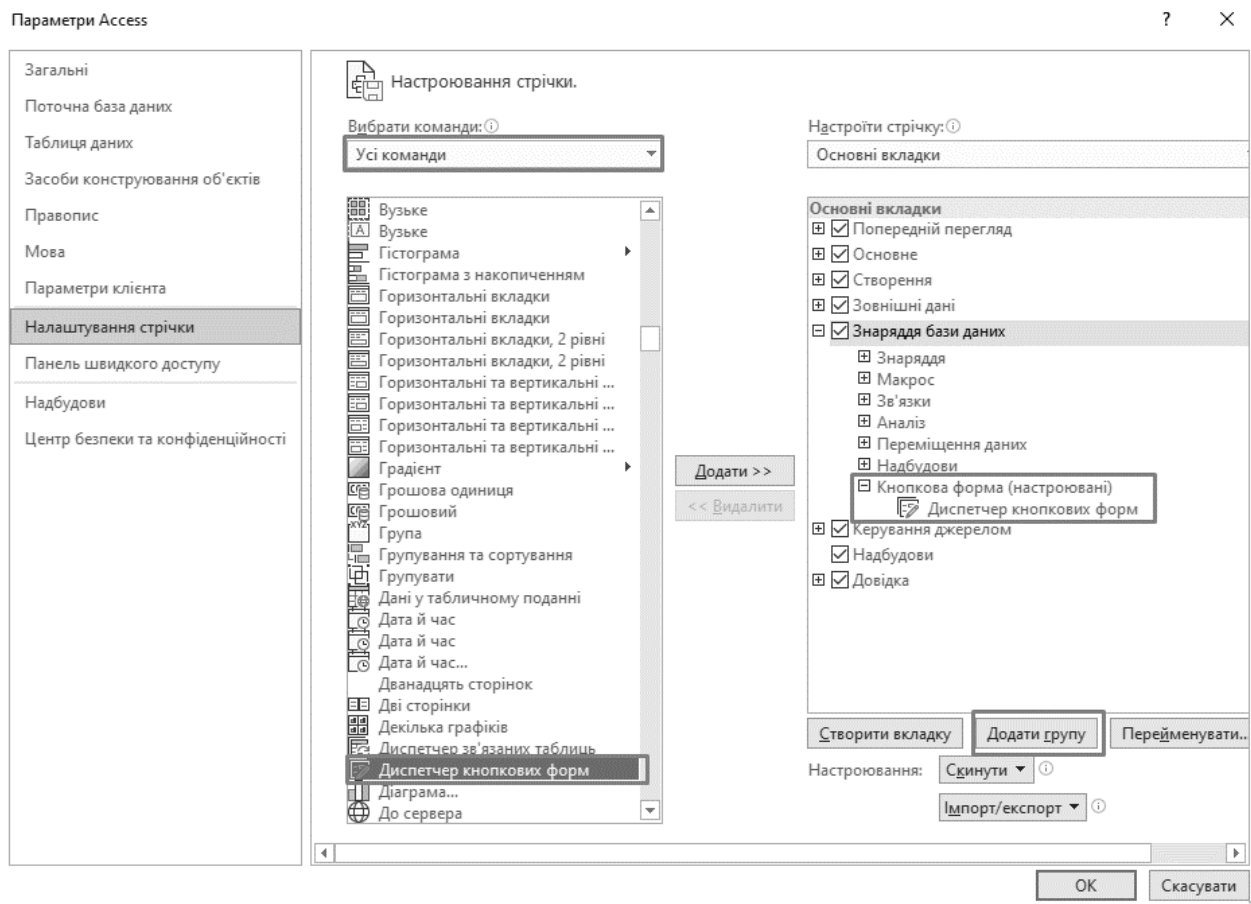


Рисунок 7.1 – Вікно **Параметри Access** для додавання **Диспетчера кнопок форм**

Щоб додати **Диспетчер кнопок форм** слід:

- клацнути по стрічці правою кнопкою миші та вибрати команду **Налаштування стрічки**;
- у вікні **Параметри Access** зліва вибрати пункт **Налаштування стрічки**, а у правій частині вікна в списку **Вибрати команди** вибрати пункт **Усі команди**;
- в списку основних вкладок до вкладки **Створити** додати

нову групу кнопкою **Створити групу**;

- в списку усіх команд знайти **Диспетчер кнопоквих форм** та додати його на вкладку **Створити** до нової групи;

- натиснути **ОК**.

За допомогою диспетчера кнопоквих форм можна створити ієрархічну систему сторінок кнопкової форми, яка може бути альтернативою звичайному ієрархічному меню.

До кнопкової форми додаються різні елементи або команди меню, які виконують різні завдання, наприклад, відкриття форм і звітів у базі даних. Елементи, які додаються до кнопкової форми, відображаються як кнопки.

Основні команди, які здатні виконувати кнопкові форми наведені у табл. 7.1.

Таблиця 7.1 – **Основні команди кнопкової форми**

Команда	Дія
Перехід до кнопкової форми	Відкриває допоміжну кнопкову форму.
Відкрити форму в режимі додавання	Відкриває форму в режимі, який дає змогу додавати лише нові записи.
Відкрити форму в режимі редагування	Відкриває форму в режимі, який дає змогу додавати або редагувати будь-які записи.
Відкрити звіт	Відкриває звіт у режимі попереднього друку.
Програма для розробки	Відкриває Диспетчер кнопоквих форм.
Вихід із програми	Закриває поточну базу даних.

Редагування елементів кнопкової форми, наприклад, зміна підпису або способу виконання команди, відбувається за допомогою **Диспетчера кнопоквих форм**.

Робочі завдання

Завдання 1. Створити кнопкову форму для роботи з екранними формами, створеними раніше. В формі передбачити можливість повернення на головну сторінку та виходу з бази даних.

Порядок виконання. Відкрити вікно Диспетчера кнопкових форм, скориставшись відповідною командою на вкладці Створити. При необхідності додати кнопку Диспетчер кнопкових форм на стрічку, виконавши вказівки з теоретичних відомостей.

Оскільки база даних не містить кнопкових форм (принаймні, відомих майстру), видається діалогове вікно, чи потрібно створювати кнопкову форму. Натисніть кнопку Так. Відкриється вікно диспетчера кнопкових форм. У цьому вікні перераховуються всі сторінки кнопкової форми (їх може бути декілька), але поки створена тільки одна сторінка як на рис. 7.2.

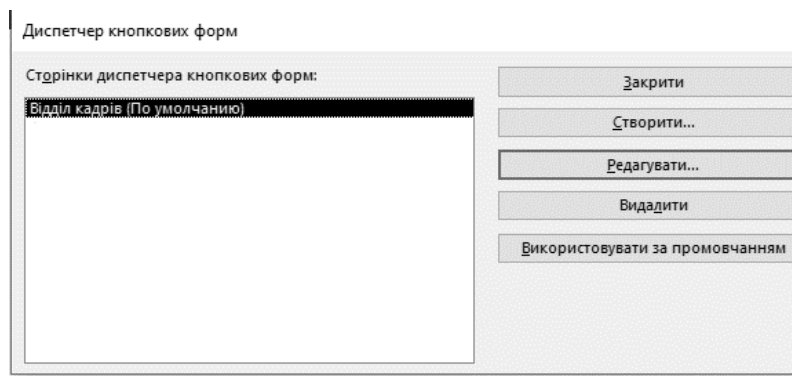


Рисунок 7.2 – Вікно диспетчера кнопкових форм при створенні першої форми

Кнопкою **Створити** додати нову сторінку кнопкової форми з назвою **Форми** для роботи з екранними формами (рис. 7.3). Тепер кнопкова форма має дві сторінки.

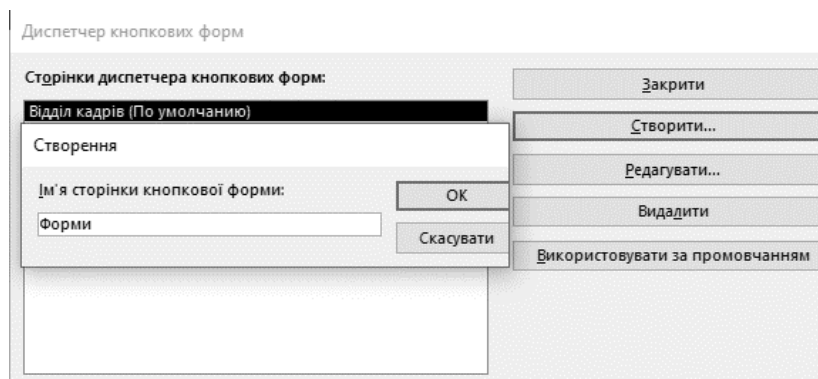
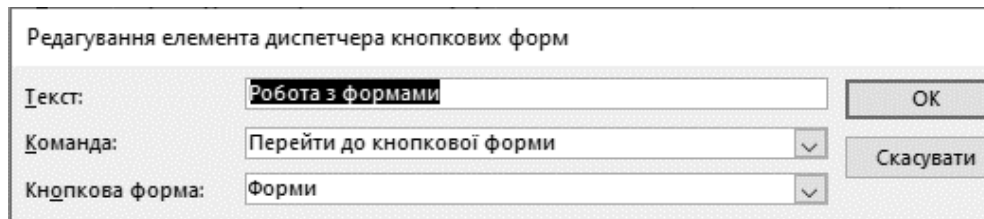


Рисунок 7.3 – Створення наступної сторінки кнопкової форми

Змінити назву головної сторінки кнопкової форми. Для цього двічі клацнути на сторінку **Головна кнопкова форма** і в поле **Назва кнопкової форми** ввести **Відділ кадрів**.

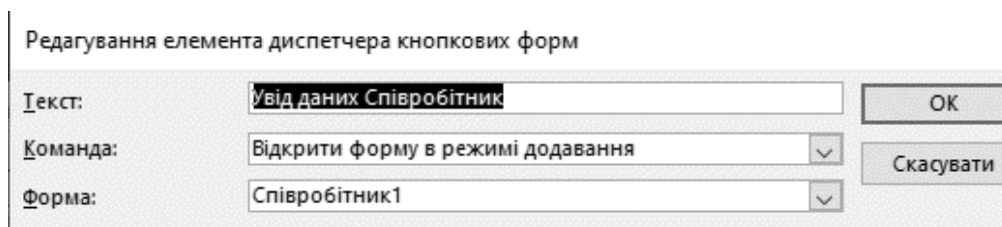
На сторінці **Відділ кадрів** кнопкою **Створити** додати новий елемент **Робота з формами** та задати для нього команду переходу на сторінку **Форми** (рис. 7.4). Закрити вікно редагування сторінки.



Редагування елемента диспетчера кнопкових форм		
Текст:	<input type="text" value="Робота з формами"/>	<input type="button" value="OK"/>
Команда:	<input type="text" value="Перейти до кнопкової форми"/>	<input type="button" value="Скасувати"/>
Кнопкова форма:	<input type="text" value="Форми"/>	

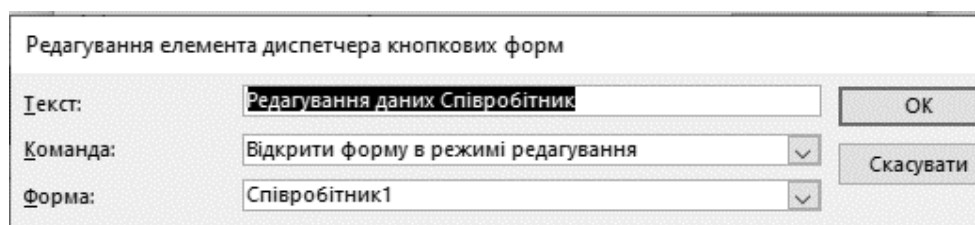
Рисунок 7.4 – Створення нового елемента кнопкової форми

Двічі клацнути на елемент **Форми** та у вікні зміни сторінки додати нові елементи для введення нових даних (рис. 7.5) та редагування існуючих (рис. 7.6) для форми **Співробітник**.



Редагування елемента диспетчера кнопкових форм		
Текст:	<input type="text" value="Увід даних Співробітник"/>	<input type="button" value="OK"/>
Команда:	<input type="text" value="Відкрити форму в режимі додавання"/>	<input type="button" value="Скасувати"/>
Форма:	<input type="text" value="Співробітник1"/>	

Рисунок 7.5 – Створення команди на увід даних до форми **Співробітник**



Редагування елемента диспетчера кнопкових форм		
Текст:	<input type="text" value="Редагування даних Співробітник"/>	<input type="button" value="OK"/>
Команда:	<input type="text" value="Відкрити форму в режимі редагування"/>	<input type="button" value="Скасувати"/>
Форма:	<input type="text" value="Співробітник1"/>	

Рисунок 7.6 – Створення команди на редагування даних до форми **Співробітник**

Створити елемент для повернення на сторінку **Відділ кадрів** як показано на рис. 7.7.

Редагування елемента диспетчера кнопок форм

Текст: Повернутися назад

Команда: Перейти до кнопкової форми

Кнопкова форма: Відділ кадрів

OK

Скасувати

Рисунок 7.7 – Створення команди для повернення на головну сторінку форми **Відділ кадрів**

Аналогічно створити елементи для введення та редагування даних до форм **Штатний розклад** та **Склад сім'ї**. Змінити порядок розташування елементів таким чином, щоб елемент повернення на сторінку **Відділ кадрів** був останнім. Для зміни порядку користуватися кнопками **Вверх** та/або **Вниз**. Закрити вікно редагування сторінки кнопкової форми.

Двічі клацнути на сторінку **Відділ кадрів** та додати новий елемент для завершення роботи з програмою як показано на рис. 7.8.

Редагування елемента диспетчера кнопок форм

Текст: Вихід

Команда: Завершити роботу застосунку

OK

Скасувати

Рисунок 7.8 – Створення елемента для завершення роботи з базою даних

Закрити вікно диспетчера кнопок форм і перевірити роботу створеної форми.

Завдання 2. Самостійно додати до кнопкової форми елементи для роботи зі створеними звітами.

Контрольні запитання

1. Що таке кнопкові форми?
2. У яких випадках створюють кнопкові форми?
3. Які інструменти використовуються для створення кнопкових форм?

4. Що треба зробити, якщо диспетчер кнопоквих форм відсутній на стрічці?
5. Що мають з себе сторінки кнопоквих форм?
6. Що називається елементом кнопкової форми?
7. Як створити елемент, що здійснює вихід з програми?
8. Як змінити порядок розміщення елементів на сторінці кнопкової форми?
9. Як створити нову сторінку кнопкової форми?
10. Яка сторінка кнопкової форми створюється за замовчанням?
11. Як додати нові кнопки на головну кнопкову форму?

ПРАКТИЧНА РОБОТА № 8

САМОСТІЙНА РОБОТА В MS ACCESS. ЗАЛІКОВЕ ЗАВДАННЯ

Мета роботи: узагальнення теоретичних знань та закріплення практичних навичок роботи з системою управління базами даних MS Access.

Теоретичні відомості

База даних (*database*) – сукупність даних, організованих відповідно до концепції, яка описує характеристику цих даних і взаємозв'язки між їх елементами. В загальному випадку база даних містить схеми, таблиці, подання, збережені процедури та інші об'єкти. Дані у базі організовують відповідно до моделі організації даних. Таким чином, сучасна база даних, крім саме даних, містить їх опис та може містити засоби для їх обробки.

Реляційна база даних – це база даних, яка сприймається користувачем як набір нормалізованих відношень різного ступеня. Реляційна база даних є сукупністю елементів даних, організованих у вигляді набору формально описаних таблиць, з яких дані можуть бути доступними або повторно зібрані багатьма різними способами без необхідності реорганізації таблиць бази даних.

У сучасних інформаційних системах для забезпечення роботи з базами даних використовують системи управління (керування) ба-

зами даних СУБД. Система управління або керування базами даних — це система, заснована на програмних та технічних засобах, яка забезпечує визначення, створення, маніпулювання, контроль, керування та використання баз даних. Застосунки для роботи з базою даних можуть бути частиною СУБД або автономними. Найпопулярнішими СУБД є *MySQL*, *PostgreSQL*, *Microsoft SQL Server*, *Oracle*, *Sybase*, *Interbase*, *Firebird* та *IBM DB2*. СУБД дозволяють ефективно працювати з базами даних, обсяг яких робить неможливим їх ручне опрацювання. Найуживаніші СУБД використовують реляційну модель, у якій дані подають у виді таблиць. Для кінцевого користувача (та прикладних програм) робота з базою даних на пряму неможлива. Всі маніпуляції над даними здійснюють через спеціальні запити, які надсилають до СУБД. СУБД опрацьовує їх і повертає результат. Безпосередньо з базою даних працює виключно СУБД.

Сучасні СУБД забезпечують **функції** щодо керування даними, які можна поділити на такі групи:

- **Оголошення даних** – створення, зміна та видалення визначень, які описують організацію даних.

- **Модифікація даних** – додавання даних, їх редагування та видалення.

- **Отримання даних** – надання даних за запитом застосунку у формі, яка дозволяє їх безпосереднє використання. Дані можуть надаватись або у формі, в якій вони зберігаються у базі даних, або в іншій формі (наприклад, через поєднання різних даних).

- **Адміністрування даних** – реєстрування та відслідковування дій користувачів, дотримання безпеки роботи з даними, забезпечення надійності та цілісності даних, моніторинг продуктивності, резервне копіювання та відновлення даних тощо.

Microsoft Office Access – система управління базами даних, програма, що входить до складу пакету офісних програм *Microsoft Office*. Має широкий спектр функцій, включаючи зв'язані запити, сортування по різних полях, зв'язок із зовнішніми таблицями і базами даних. Завдяки вбудованій мові VBA, в самому Access можна писати підпрограми.

Основні компоненти *MS Access*:

- конструктор таблиць;

- конструктор екранних форм;
- конструктор *SQL*-запитів;
- конструктор звітів, що виводяться на друк;
- макроси.

Таблиця – це основний об'єкт бази даних, призначений для збереження даних, документів та інших облікових записів.

Запит – вибирає дані з таблиць згідно з умовами, що задаються.

Форма – відображає дані з таблиць або запитів відповідно до форматів, описаних користувачем. Форма дозволяє переглядати, редагувати та друкувати дані.

Звіт – відображає і друкує дані з таблиць або запитів згідно з описаним користувачем форматом. У звіті дані редагувати не можна.

Макроси – інструменти, за допомогою яких можна автоматизувати завдання та розширити можливості форм, звітів і елементів керування без та з написанням коду в модулі *Visual Basic for Applications (VBA)*.

Порядок виконання.

Завдання 1. Створити таблиці згідно варіанту в режимі конструктора таблиць, визначити типи даних полів. Встановити властивості полів, де це необхідно (маска вводу, Майстер підстановки, обов'язковість, ключове поле тощо).

Варіанти завдань:

Таблиця 8.1 – **Варіант 1. Передплата на періодичні видання**

Видання	Видавництва	Передплатник	Передплата
Індекс	Код видавництва	Код передплатника	Код квитанції
Назва	Назва	Назва	Код передплатника
Вартість номеру	Адреса	Адреса	Індекс видання
Періодичність	Електронна пошта		Термін підписки
Код видавництва	ПІБ головного редактора		Дата підписки

Таблиця 8.2 – Варіант 2. Футбольна ліга

Гравці	Команди	Персонал	Посади
Код гравця	Код команди	Код	Код посади
Прізвище	Назва	Прізвище	Назва
Ім'я	Стадіон	Ім'я	
Дата народження	Місто	По-батькові	
Зріст		Дата народження	
Амплуа		Код команди	
Код команди		Код посади	
Номер			

Таблиця 8.3 – Варіант 3. Вироби народної творчості

Вироби	Кваліфікація	Майстри	Промисли
Код виробу	Код кваліфікації	Код майстра	Код промислу
Назва виробу	Назва	Прізвище	Назва промислу
Код промислу		Ім'я	
Код майстра		По-батькові	
		Адреса	
		Дата народження	
		Код кваліфікації	

Таблиця 8.4 – Варіант 4. Розклад занять викладачів

Предмети	Заняття	Викладачі	Кафедри
Код предмету	Код	Код викладача	Код кафедри
Назва	Код предмету	Прізвище	Назва
Скорочена назва	Код викладача	Ім'я	
	День тижня	По-батькові	
	Номер пари	Посада	
	№ аудиторії	Код кафедри	
	Група		

Таблиця 8.5 – **Варіант 5. Вироби кондитерських фабрик**

Вироби	Види виробів	Виробники	Форми власності
Код виробу	Код виду	Код виробника	Код форми власності
Назва	Назва виду	Назва	Назва
Код виду		Адреса	
Одиниця вимірювання		Код форми власності	
Ціна за одиницю		Відповідальна особа	
Код виробника		Телефон	
		Електронна пошта	

Таблиця 8.6 – **Варіант 6. Склад**

Споживач	Товари	Накладні	Постачання
Код споживача	Код товару	№ накладної	№ накладної
Назва	Назва	Дата	Код товару
Адреса	Одиниця вимірювання	Код споживача	Кількість
Телефон	Ціна за одиницю		
Примітки	Виробник		

Таблиця 8.7 – **Варіант 7. Будівельна організація**

Об'єкти	Типи об'єктів	Матеріали	Витрати
Код об'єкту	Код типу об'єкту	Код матеріалу	Код об'єкту
Назва об'єкту	Назва типу	Назва матеріалу	Код матеріалу
Код типу об'єкту		Одиниця вимірювання	Витрата матеріалу
Адреса		Ціна за одиницю	

Таблиця 8.8 – **Варіант 8. Видавництво**

Автори	Книги	Замовники	Замовлення
Код автора	Код книги	Код замовника	Код замовлення

Автори	Книги	Замовники	Замовлення
Прізвище	Тип видання	Назва	Код замовника
Ім'я	Назва книги	Адреса	Дата замовлення
По-батькові	Код автора	Телефон	Код книги
Дата народження	Вартість	Примітки	Тираж
Країна			Сплачено

Таблиця 8.9 – **Варіант 9. Відділ кадрів**

Співробітники	Кваліфікація	Посади	Відділи
Код співробітника	Код кваліфікації	Код посади	№ відділу
Прізвище	Назва кваліфікації	Назва посади	Назва відділу
Ім'я		Мінімальний оклад	Кількість штатних одиниць
По батькові		Максимальний оклад	
Адреса			
Код кваліфікації			
Код посади			
Дата зарахування на роботу			
№ відділу			
Освіта			

Таблиця 8.10 – **Варіант 10. Спортивне товариство**

Спортсмени	Тренери	Види спорту	Тренування
Код спортсмена	Код тренера	Код виду спорту	Код тренера
Прізвище	Прізвище	Назва	Код спортсмена
Ім'я	Ім'я		
По-батькові	По-батькові		
Дата народження	Дата народження		
Звання	Звання		
	Код виду спорту		

Завдання 2. Визначити зв'язки між таблицями, отримати схему даних.

Завдання 3. Заповнити таблиці даними. Кожна таблиця повинна містити не менш 5 записів.

Завдання 4. Створити екранні форми для роботи з таблицями бази даних Вашого варіанту.

Завдання 5. Створити запит на вибірку даних з різних таблиць.

Завдання 6. Створити звіт на основі запиту.

Завдання 7. Створити кнопкову форму для швидкого відкриття екранних форм і звіту.

Завдання 8. Створенну базу даних надіслати на перевірку.

Завдання 9. Письмово відповісти на контрольні запитання.

Контрольні запитання

1. Що таке база даних?
2. Що містить сучасна база даних?
3. Що таке реляційна база даних?
4. Що називають СУБД?
5. Яке призначення СУБД?
6. Назвіть та поясніть функції сучасних СУБД.
7. Що таке *Microsoft Access*?
8. Назвіть основні компоненти *MS Access*.
9. Що називають таблицею в *MS Access*?
10. Що таке запит в *MS Access*?
11. Яке призначення форми в *MS Access*?
12. Яке призначення звіту в *MS Access*?
13. Що таке макроси в *MS Access*?

Навчальне видання

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт та самостійних робіт
з дисципліни «Інформаційні системи і технології»
для студентів усіх спеціальностей.

Розділ «Система управління базами даних *Microsoft Access*»

Укладачі: ШЕВЧЕНКО Вікторія Олександрівна
КАРПІШЕН Богдан Сергійович
КОСТИКОВА Марина Володимирівна

Відповідальний за випуск *Г. А. Плехова*

В авторській редакції

Комп'ютерна верстка *Н.В. Ольховської*

План 2024 р. Поз. 52

Підписано до друку 24.05.2024 р. Формат 60×84 1/16.

Гарнітура Times New Roman Cyr.

Ум. друк. арк. 4,3. Обл.-вид. арк. 4,8

Зам. № 33/24-В. Наклад сайт.

ВИДАВНИЦТВО

Харківського національного автомобільно-дорожнього університету
Видавництво ХНАДУ, 61002, Харків-МСП, вул. Ярослава Мудрого, 25.

Тел. /факс: (057)700-38-64; 707-37-03,
e-mail: rio@khadi.kharkov.ua

Свідоцтво Державного комітету інформаційної політики, телебачення та радіомовлення
України про внесення суб'єкта видавничої справи до Державного реєстру
видавців, виготівників і розповсюджувачів видавничої продукції,
серія № ДК №897 від 17.04 2002 р.