

## Лабораторна робота № 8

# СТВОРЕННЯ МАКРОСІВ

### Навчальні питання

1. Засоби програмування в Access.
2. Специфіка використання макросів та засоби їх створення.
3. Створення командної кнопки з макросом.
4. Створення самостійного (глобального) макросу.
5. Створення складного макросу.

### Завдання

1. Запустити базу даних MS Access з ім'ям *ПрізвищеБД*, створену Вами на попередній лабораторній роботі.

2. За запитом *Ювіляри* створити однойменну форму та додати до неї командну кнопку з макросом, при натисканні на яку відкриватиметься форма *Працівники* з докладними анкетними даними.

3. Створити самостійний (глобальний) макрос, який виконуватиме дві дії (макрокоманди): 1) запускатиме вікно з попередженням, наприклад таким: "Буде відкрита таблиця, в яку можна вносити змінення"; 2) відкриватиме таблицю *Працівники*.

4. Застосувати щойно створений макрос до форми *Ювіляри* (у вигляді пов'язаної з цим макросом кнопки, клацання по якій запускатиме створений макрос) і до звіту *Працівники* (у вигляді пов'язаного з цим макросом елемента керування *Підпис*).

5. Створити форму з переліком існуючих посад на підприємстві (за таблицею *Посади*), а далі надати можливість за назвою посади віднайти прізвища працівників, їхнє місце роботи та інші відомості, створивши взаємодію між формами за допомогою макросу. Для розв'язання поставленої задачі створити не лише форму *Посади*, а ще й підформу з анкетними даними працівників. На підформі мають відображатися дані, які задовольнятимуть умові, заданій на головній формі.

### Теоретичні відомості

#### 1. Засоби програмування в Access

Створення бази даних зазвичай починається зі створення декількох об'єктів, наприклад таблиць, форм і звітів, але рано чи пізно доводиться вдаватися до засобів програмування, щоб автоматизувати деякі процеси та пов'язати об'єкти бази даних.

В Access програмуванням називається процес розширення функціональності бази даних з використанням *макросів* або *модулів*. **Макроси** – це невеликі програми *мовою макрокоманд* (мовою сценаріїв), інакше кажучи, це набір готових

дій, які можна вибирати і призначати тим або іншим об'єктам на формах і звітах за допомогою конструктора макросів. **Модулі** – це набори процедур *мовою програмування VBA* (Visual Basic for Applications), тобто це об'єкти, які містять програми мовою Visual Basic. Припустимо, що користувач створив форму і звіт та прагне додати на форму кнопку, при натисканні якої буде відкриватися звіт. У цьому разі програмування буде включати створення макросу або процедури VBA і налаштування події *OnClick* для кнопки таким чином, щоб при її натисканні запускався відповідний макрос або процедура. Для реалізації простих операцій, таких як відкривання звіту, можна використовувати майстер кнопок, але можна відключити його і самостійно запрограмувати все на VBA. Відтак, на відміну від інших програм Microsoft Office, таких як Word чи Excel, в яких макросом називається код VBA, в Access існує відмінність між поняттями макросу і модуля.

Основним призначенням макросів і модулів в Access є створення зручного інтерфейсу програми, в якому форми і звіти відкривалися б при натисканні кнопок на формах або на панелях інструментів.

Модулі є більш потужним засобом створення програмних розширень у середовищі Microsoft Office. Застосування модулів вимагає від користувачів знань основних принципів об'єктно-орієнтованого програмування. У свою чергу, конструктор макросів має більш структурований інтерфейс, ніж редактор VBA, дозволяючи додавати програмний код до елементів керування та об'єктів без знань особливостей програмування VBA, оскільки кожна дія (команда макросу) просто вибирається зі списку. Саме тому більшість користувачів надають перевагу створенню саме макросів, аніж написанню коду VBA.

Прийняття рішення щодо використання макросів, VBA-коду або і того, й іншого залежить в першу чергу від того, де буде використовуватися база даних, і рішення слід приймати на основі вимог до безпеки і функціональності додатка. Наприклад, якщо база даних буде використовуватися одним користувачем на одному комп'ютері, можна виконати більшість завдань програмування за допомогою VBA. Якщо ж планується розмістити базу даних на сервері і спільно використовувати її з іншими користувачами, краще відмовитися від застосування VBA з міркувань безпеки. Також треба враховувати ту особливість, що код VBA несумісний з функцією веб-публікації, і, якщо планується публікація бази даних у якості веб-дodatка Access, для виконання завдань програмування доцільно використовувати макроси замість коду VBA.

В якості рекомендацій забезпечення безпеки бази даних можна поради розробнику БД віддавати перевагу використанню макросів і вдаватися до коду VBA для виконання лише тих операцій, які неможна виконати за допомогою макрокоманд. Крім того, слід намагатися використовувати тільки ті дії, для виконання яких не потрібно висловлювати довіру до БД. Таке обмеження на використання макрокоманд дозволяє користувачам бути впевненими в тому, що БД не містить програмні засоби, які можуть зашкодити даним та іншим файлам на комп'ютерах.

У свою чергу, без модулів VBA не обійтися при виконанні деяких дій на рівні системи, наприклад, за потреби запуску з Access іншої програми – Word чи Excel, оскільки макрос неможна використовувати для виконання операцій

поза Access. А за допомогою VBA можна не лише запустити відповідну програму, а ще й перевірити наявність файлу на комп'ютері, використовувати автоматизацію або динамічний обмін даними (DDE) для взаємодії або викликати функцію з бібліотеки динамічного компонування (DLL) Windows.

## 2. Специфіка використання макросів та засоби їх створення

**Макрос** – іменованій набір макрокоманд для виконання певних операцій, що дозволяє автоматизувати роботу додатка. Використовуючи макроси для виконання рутинних операцій, можна істотно заощадити час і сили. Крім того, оскільки всякий раз при запуску макросу буде здійснюватися одна і та сама послідовність дій, макрос зробить роботу з базою даних більш ефективною і акуратною. Макроси можуть бути як простими, що складаються з однієї дії, так і складними, що включають логічні операції і кілька взаємопов'язаних дій.

Об'єктам (наприклад, формам, звітам) і елементам керування на них (наприклад, кнопкам, текстовим полям) можна на виконання різних подій призначати різні макроси. Прикладами подій можуть бути клацання кнопки миші, відкривання або закривання форми, запуск звіту або змінення даних у текстовому полі тощо. Наприклад, якщо до форми додати кнопку, подія *OnClick* кнопки пов'язуватиметься з макросом, що містить команди, які мають виконуватися при кожному натисканні кнопки.

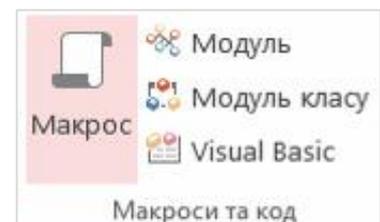
Події можуть запускатися зовнішніми по відношенню до Access факторами, такими як системні події, макроси або процедури, пов'язані з іншими подіями. Якщо додати значну кількість макросів до подій різних об'єктів на формах і звітах, база даних може стати вельми складною, але в більшості випадків можна досягти необхідного результату майже без програмування.

Призначати макроси, пов'язані з подіями, можна за допомогою вікна властивостей форм та звітів (до речі, таблиці і запити подіями не володіють).

Починаючи з версії Access 2010, в програму включено багато нових дій (макрокоманд), які дозволяють створювати більш ефективні макроси, ніж у попередніх версіях Access. Наприклад, за допомогою макрокоманд можна створювати і використовувати тимчасові глобальні змінні, а також більш гнучко обробляти помилки. У попередніх версіях Access ці можливості були доступні тільки при використанні VBA. Крім того, можна вбудовувати макроси безпосередньо у властивості подій об'єктів або елементів керування. Вбудований макрос стає частиною об'єкта або елемента керування і залишається з ним при його переміщенні або копіюванні.

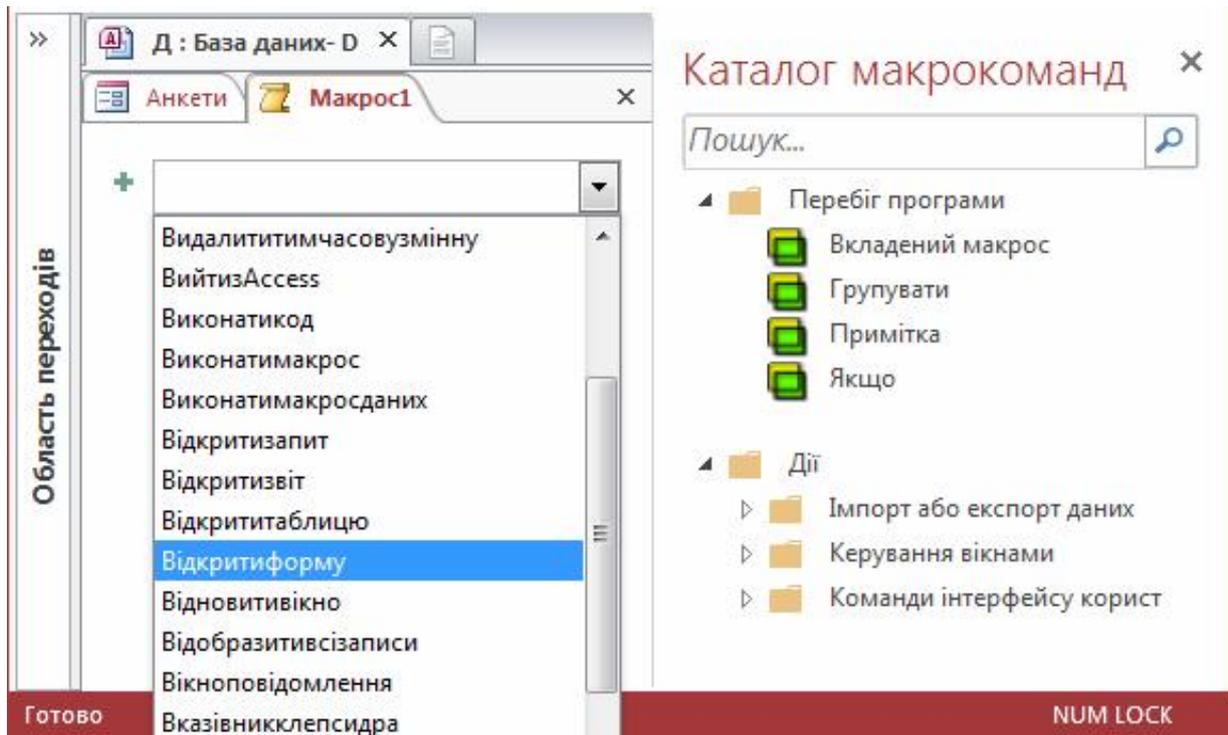
Для створення макросів за допомогою *Конструктора макросів*, який можна запустити командою *Макрос* на вкладці *Створення* у групі *Макроси та код*.

Вибрати ту чи іншу макрокоманду можна з розкритого списку



Створювані макроси можуть бути від початку пов'язані з якимось елементом, наприклад, з кнопкою на формі, або бути самостійними об'єктами, багато-

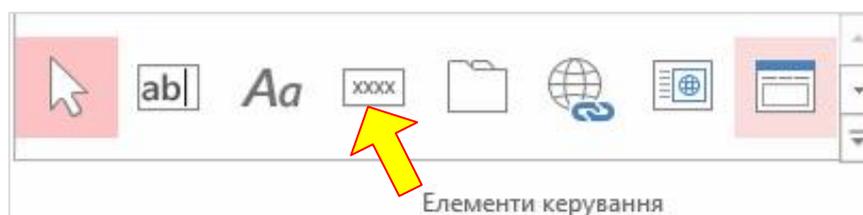
разово використовуваними у різних об'єктах на різних формах і звітах, наприклад, один і той самий макрос може виконуватися при виборі пункту меню на формі або при натисканні кнопки на звіті.



### 3. Створення командної кнопки з макросом

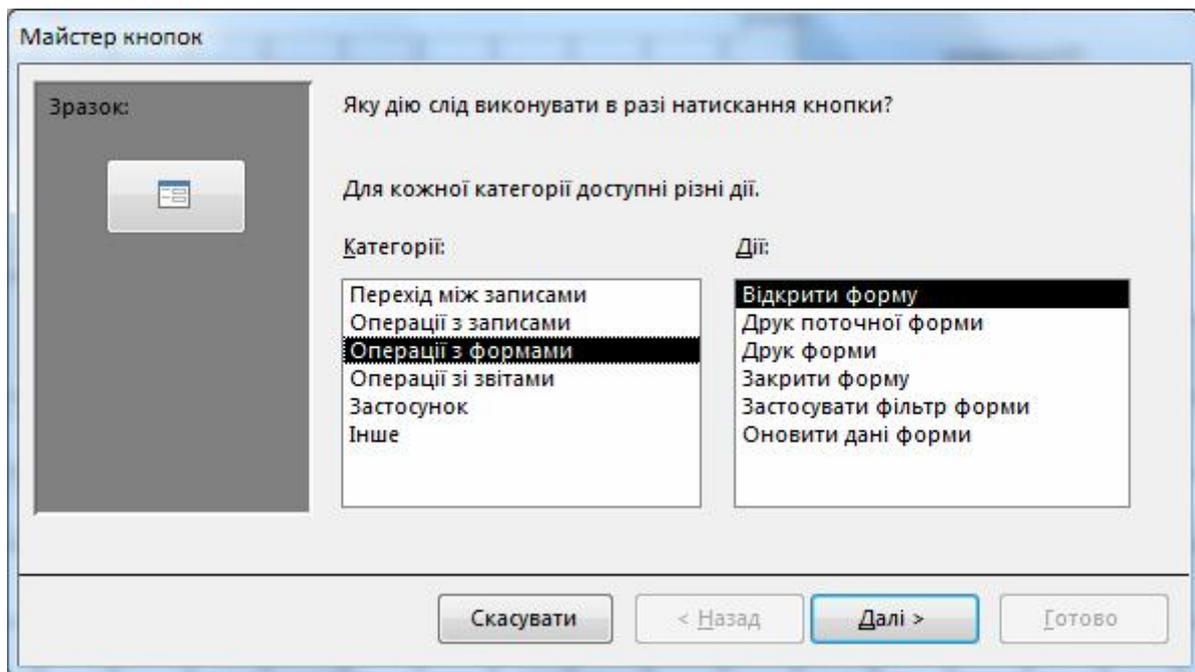
Послідовність дій для створення макросу розглянемо на прикладі створення форми *Ювіляри* на основі однойменного запиту та додавання на цю форму кнопки, при натисканні на яку відкриється форма *Працівники* з докладними анкетними даними працівника:

1. В області переходів клацнути один раз (без відкривання) на запиті *Ювіляри* та виконати команду *Форма* на вкладці *Створення*. Після цього створиться та відкриється форма, яку треба зберегти з ім'ям *Ювіляри* (не закриваючи її).
2. Перейти до режиму *Конструктор* та збільшити ширину форми.
3. На вкладці *Конструктор* у групі *Елементи керування* клацнути стрілку вниз, щоб відобразити колекцію елементів керування, і вибрати елемент *Кнопка*, після чого розмістити кнопку на вільному місці форми.



При цьому відкриється вікно майстра створення кнопок.

4. На першій сторінці майстра кнопок в області *Категорії* вибрати пункт *Операції з формами*, а в області *Дії* вибрати *Відкрити форму* і натиснути кнопку *Далі*.



5. Вибрати назву відкритої форми *Працівники* і натиснути кнопку *Далі*.

6. На наступному кроці встановити опцію *Відкрити форму та знайти певні дані для відображення* і натиснути кнопку *Далі*.

7. Оскільки запит *Ювіляри* і таблиця *Працівники*, на основі яких створені обидві форми, не були безпосередньо пов'язані, треба вибрати поля, які слід використовувати для відбирання даних. Для цього вибрати поле *КодПрацівника* в області *Ювіляри* й однойменне поле в області *Працівники*, після чого натиснути кнопку і кнопку *Далі*.

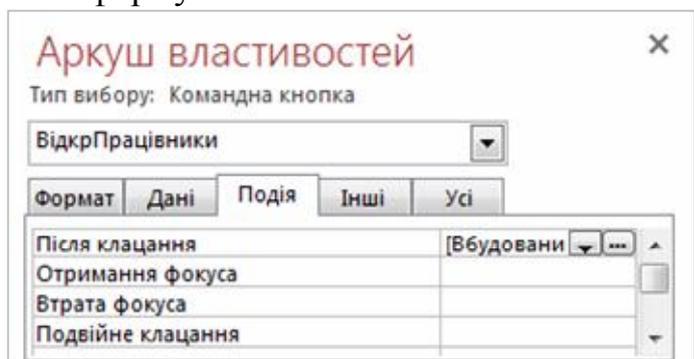
8. На наступному кроці вибрати опцію *Текст* та вписати текст, який буде відображатися на створюваній кнопці, наприклад: "Показати анкетні дані співробітника". Натиснути кнопку *Далі*.

Проте можна скористатися варіантом *Зображення*, який дозволить вибрати і відобразити на кнопці файл-малюнок формату *bmp* або *ico*.

9. Ввести ім'я кнопки *ВідкрПрацівники*<sup>1</sup> і натиснути кнопку *Готово*.

Після цього кнопка буде вставлена на форму.

10. Для додавання ще однієї дії, яка має виконуватися при натисканні цієї кнопки, треба в режимі *Конструктор* натиснути [F4] для відображення вікна властивостей. На вкладці *Подія* аркуша властивостей для події *Після клацання* натиснути кнопку побудови , що призведе до відкриття конструктора



макросу. Далі треба за допомогою списку *Додати нову дію* вибрати

<sup>1</sup> Можна погодитися з ім'ям, що пропонується за замовчуванням, оскільки ім'я не відображається на кнопці. Проте має сенс ввести значуще ім'я, щоб за потреби послатися на кнопку (наприклад, щоб при визначенні послідовності переходів між елементами керування на формі при натисканні клавіші [Tab], було набагато простіше розрізняти кнопки).

подальшу дію, наприклад *Закритивікно*, і заповнити рядки у бланку макросу: у рядку *Тип об'єкта* вибрати *Форма*, а в рядку *Ім'я об'єкта* вибрати *Ювіляри*.

11. Залишилося закрити макрос, перейти до режиму форми *Ювіляри* і натиснути на створену кнопку, щоб переконатися, що вона працює саме так, як очікувалось.

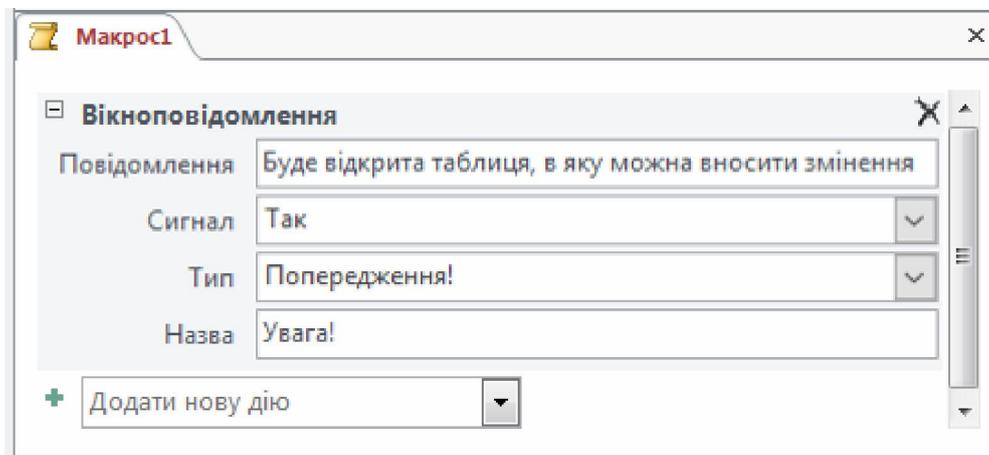
#### 4. Створення самостійного (глобального) макросу

Макрос є об'єктом бази даних, який можна використовувати неодноразово в різних варіантах. До того як створити макрос, слід подумати щодо його подальше використання. Наприклад, перед тим, як приступити до змінення даних в таблиці *Працівники*, слід попередити про це користувача, а потім відкрити потрібну таблицю. Отже, треба виконати дві дії (дві макрокоманди): 1) запустити вікно з попередженням; 2) відкрити потрібну таблицю.

Розглянемо порядок створення такого макросу, а потім використаємо цей макрос для форми *Ювіляри* і звіту *Працівники*:

1. На вкладці *Створення* у групі *Макроси та код* натиснути на значок  *Макрос*.

2. У вікні *Макрос1* розкрити список  і вибрати макрокоманду *Вікноповідомлення*, що призведе до відкривання бланка складання повідомлення. Наприклад, у рядку *Повідомлення* ввести текст "Буде відкрита таблиця, в яку можна вносити змінення" (без лапок), у рядку *Тип* вибрати зі списку *Попередження!*, у рядку *Назва* ввести "Увага!" (перша частина макросу готова).



3. Щоб додати до макросу ще одну макрокоманду, треба у вікні макросу зі списку  *Додати нову дію* вибрати дію *Відкрититаблицю*, після чого вибрати зі списку ім'я таблиці *Працівники*.

Як видно на рисунку вище, бланк макросу складається з окремих областей для кожної макрокоманди.

Сам макрос можна відкрити в режимі *Конструктор* і відкоригувати його.

4. Зберегти макрос з ім'ям *ВідкрТаблПрацівники*, після чого в області переходів з'явиться зазначене ім'я створеного макросу.

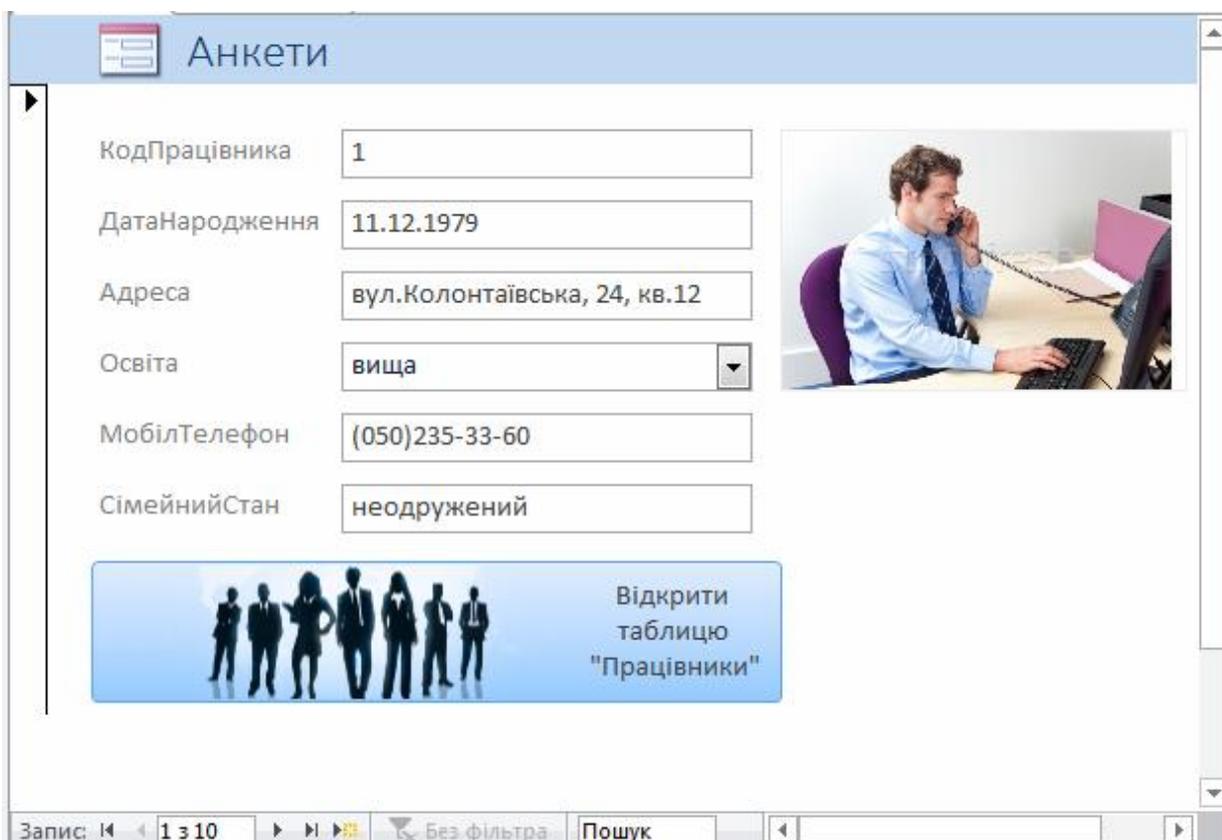
Залишилося найголовніше, прив'язати макрос до якогось об'єкта (або об'єктів).

5. Створити спеціальний підпис у звіті *Працівники*, клацання по якому запустить створений макрос. Для цього треба відкрити цей звіт у режимі *Конструктор*, розмістити у верхньому колонтитулі звіту (заголовку) (чи іншому місці) новий елемент керування **Aa** *Підпис*. Відкрити вікно властивостей цього елемента, у полі *Підпис* ввести текст "Перегляд таблиці "Працівники"". Також можна змінити інші властивості, наприклад, колір фону, розмір шрифту тощо.

Для прив'язки макросу до цього підпису треба на вкладці *Подія* у рядку *Після клацання* вибрати зі списку ім'я потрібного макросу *ВідкрТаблПрацівники*. Зберегти звіт, перейти до режиму *Подання звіту* та перевірити працездатність макросу.

6. Для ще одного використання цього макросу і для форми *Анкети*, слід відкрити форму в режимі *Конструктор* і розмістити на формі елемент керування **XXXX** *Кнопка*, що призведе до відкривання майстра створення кнопок. На першій сторінці майстра в області *Категорії* клацанням вибрати пункт *Інше*, зі списку *Дія* вибрати *Запустити макрос* і натиснути кнопку *Далі*. Після цього вибрати ім'я макросу *ВідкрТаблПрацівники* і натиснути кнопку *Далі*. Вибрати відповідний рисунок і/або вписати текст "Відкрити таблицю "Працівники"", який буде відображатися на створюваній кнопці. Натиснути кнопку *Готово*. Залишилося перейти до режиму форми і перевірити правильність роботи макросу.

Отже, зручністю використання макросу є ще й те, що він поширюється на всі типи об'єктів бази даних.



The screenshot shows a form titled "Анкети" (Surveys) with the following fields and a button:

КодПрацівника	1
ДатаНародження	11.12.1979
Адреса	вул.Колонтаївська, 24, кв.12
Освіта	вища
МобілТелефон	(050)235-33-60
СімейнийСтан	неодружений

Below the fields is a button with a silhouette of a group of people and the text "Відкрити таблицю 'Працівники'" (Open table 'Employees').

The bottom status bar shows: Запис: 1 з 10, Без фільтра, Пошук.

## 5. Створення складного макросу

Оскільки макроси в Access складаються зі стандартних дій (макрокоманд), то можна перекласти весь пошук необхідних даних на макроси замість того, щоб створювати складні умови відбору даних у запитах. Припустимо, керівництво організації цікавлять конкретні дані про працівників, які займають певні посади. Швидше за все, керівництво вже заплуталось у назвах посад на підприємстві, це пов'язано з постійною реорганізацією і створенням наказів щодо внесення змінень до штатного розкладу. Напевно, керівнику слід надати форму з переліком існуючих посад на підприємстві, а далі дати можливість за назвою посади віднайти прізвища працівників, їхнє місце роботи та інші відомості. Спробуємо створити взаємодію між формами за допомогою макросу.

Для розв'язання поставленої задачі потрібно дві форми. Перша форма (головна) міститиме список посад на певний момент часу в організації, а друга форма (підпорядкована форма, підформа) – дані, які цікавлять керівника. З підформи мають вибиратися дані, які задовольнятимуть умові, заданій керівником. Отже, зв'язок між формами покладається на програму, яка буде формуватися за допомогою макросів. Виконаємо кілька кроків для розв'язання поставленої задачі.

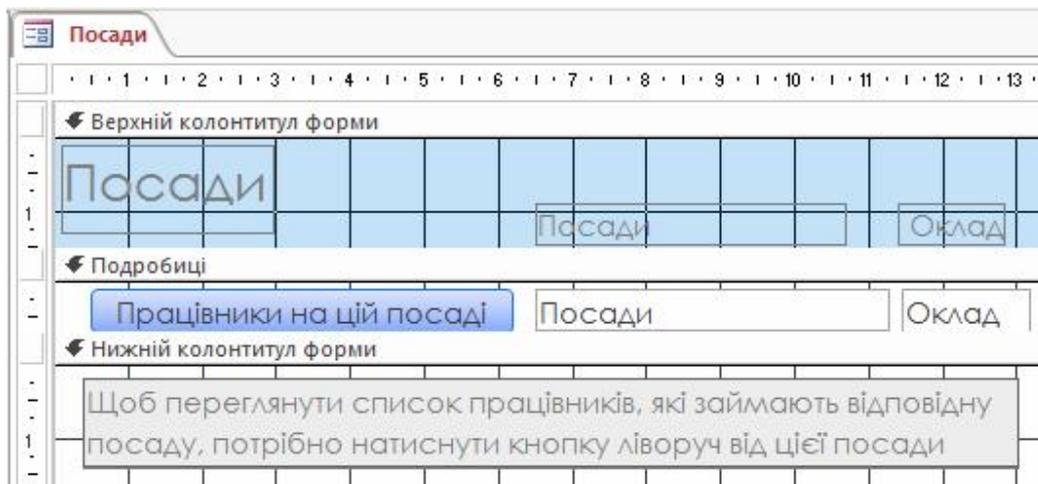
### Крок 1. Розробка головної форми *Посади*:

1. Створити форму за таблицею *Посади*, виділивши відповідну таблицю в області переходів (не відкриваючи її) та вибравши на вкладці *Створення* у групі *Форми* команду *Майстер форм*. Далі треба вибрати обидва поля та на наступному кроці вибрати макет *Таблиця* (у раніших версіях Access – вид форми *Стрічковий* (рос. *Ленточный*)), а на останньому кроці задати ім'я форми – *Посади*. Ширину полів налаштувати власноруч у режимі розмітки або конструктора. При відкриванні форма матиме такий вигляд.

Посади	Оклад
головний спеціаліст	4 300,00
інспектор	3 500,00
начальник відділу	5 000,00
провідний юрист	4 500,00
програміст	4 550,00
сисадмін	4 700,00
спеціаліст	4 000,00
юрист	4 000,00

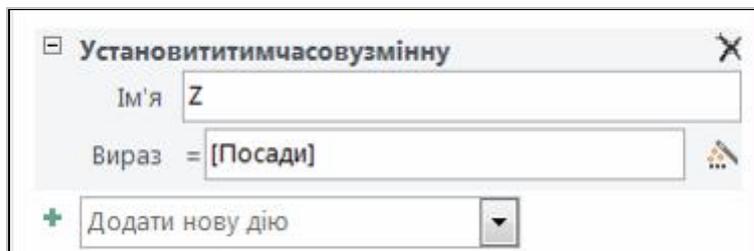
2. Для внесення змінень до створеної форми потрібно перейти до режиму *Конструктор*, посунути праворуч поля в області *Подробиці* (даних), розмістити на вільному місці елемент керування  *Кнопка*. На запуск майстра створення кнопок слід відреагувати натисканням кнопки *Скасувати*. У вікні властивостей ввести у властивості *Підпис* текст "Працівники на цій посаді".

В області форми *Нижній колонтитул* (або *Примітка*) розташувати елемент керування **Aa Підпис**, у властивості *Підпис* якого ввести текст "Щоб переглянути список працівників, які займають відповідну посаду, потрібно натиснути кнопку ліворуч від цієї посади". При бажанні можна задати інші параметри форматування форми і полів на ній, після чого зберегти готову форму.



3. Для створення макросу, який буде виконуватися при натисканні щойно створеної кнопки, треба для форми *Посади* у режимі *Конструктор* клацнути правою кнопкою миші на кнопці *Працівники на цій посаді* і вибрати з контекстного меню команду *Обробка подій*. У вікні вибрати параметр *Побудовник макросів* і натиснути кнопку *ОК*. Далі у вікні конструктора макросу треба зі списку  *Додати нову дію* вибрати дію *УстановитиТимчасовуЗмінну*.

У відкритому бланку макросу в рядку *Ім'я* ввести ім'я тимчасової змінної *Z*, за допомогою якої будуть передаватися дані з головної форми до підформи. Щоб це відбувалося, у рядку *Вираз* бланка слід вписати ім'я поля [*Посади*], значення якого будуть поєднувальною ланкою між головною формою і підформою. Причому, ім'я цього поля можна не вписувати, а вибрати зі списку полів, скориставшись кнопкою .



Закрити і зберегти бланк макросу.

**Крок 2.** Вибір підформи і пов'язування її з головною формою:

1. Вибір підформи можна здійснити з уже наявних форм або створити нову форму. У даному прикладі доцільно створити спеціальну форму з найменуванням *Анкетні дані*, яка міститиме поля з таблиць *Працівники* та *Анкети*. До речі, у ній обов'язково має бути поле *Посада*, оскільки у подальшому саме за його значенням буде відбуватися відбір записів для відображення.

Перейти до режиму форми, зберегти та закрити її.

2. Тепер можна продовжити процес створення макросу. Треба форму *Посади* відкрити у режимі *Конструктор*, натиснути кнопку *Працівники на цій посаді* й у вікні властивостей на вкладці *Подія* у рядку *Після клацання* натиснути значок . Це призведе до повторного відкривання бланка макросу.

3. Для додавання нової дії треба зі списку  *Додати нову дію* вибрати рядок з найменуванням *ВідкритиФорму*. У бланку макросу треба зі списку вибрати ім'я форми *Анкетні дані*, а в рядку *Умова відбору* ввести вираз:

$$[Посада]=[TempVars]![Z]$$

У рядку *Режим вікна* вибрати зі списку значення *Діалогове вікно*. Це робиться для того, щоб користувач міг переглядати дані, зібрані з підпорядкованої форми, у відповідності до поставленої умови.

4. Зберегти макрос і закрити його. Перевірити правильність роботи кнопки *Працівники на цій посаді* на формі *Посади*.

Так, при клацанні по цій кнопці біля посади *Юрист* відкриється підформа *Анкетні дані* з відображенням даних про працівників саме на цій посаді.

The screenshot shows two overlapping windows from a Microsoft Access application. The top window, titled "Посади", displays a table with three rows of employee data. The bottom window, titled "Анкетні дані", shows a form for a lawyer named Antonov Igor, including fields for position, surname, name, department, date of birth, date of hire, education, mobile phone, and marital status. A small photo of a man at a desk is also visible in the form.

Посади	Оклад
Інспектор	3 500,00
начальник відділу	5 000,00
провідний юрист	4 500,00

Посада	юрист	Дата Народження	11.12.1979
Прізвище	Антонов	МобільТелефон	(050)235-33-60
Ім'я	Ігор	СімейнийСтан	неодружений
Відділ	юридичний		
ДатаПрийому	05.06.2013		
Освіта	вища		

### Питання для самостійної роботи

1. Які засоби програмування підтримує MS Access?
2. Що являє собою макрос в Access?
3. Що являє собою модуль в Access?
4. Чим відрізняється модуль від макросу?
5. В яких випадках в Access доречніше використовувати макроси, а в яких – модулі?
6. Як створити макрос і через нього створити зв'язок з об'єктами бази даних?
7. Яка мова програмування використовується в MS Access для написання програмного коду?
8. Як створити на формі кнопку з макросом?
9. Як створити самостійний (глобальний) макрос, який можна використовувати неодноразово у різних об'єктах керування?
10. Де відображаються глобальні макроси, створені у БД?
11. Що треба зробити, щоб запустити глобальний макрос?