

Українська державний університет залізничного транспорту  
ФАКУЛЬТЕТ АВТОМАТИКИ, ТЕЛЕМЕХАНІКИ ТА ЗВ'ЯЗКУ

Кафедра спеціалізованих комп'ютерних систем

## **МЕТОДИЧНІ ВКАЗІВКИ**

до лабораторних робіт  
з дисциплін

*«ОРГАНІЗАЦІЯ БАЗ ДАНИХ»*  
*«АВТОМАТИЗАЦІЯ БІЗНЕС-ПРОЦЕСІВ»*  
*«ОПЕРАЦІЙНІ СИСТЕМИ ТА БАЗИ ДАНИХ»*

Частина 2

Харків 2016

Методичні вказівки розглянуто і рекомендовано до друку на засіданні кафедри спеціалізованих комп'ютерних систем 22 лютого 2016 р., протокол № 7.

У методичних вказівках приводяться приклади розв'язання типових задач, що розглядаються на лабораторних заняттях з навчальних дисциплін «Організація баз даних», «Автоматизація бізнес-процесів» та «Операційні системи та бази даних». Також наведено варіанти завдань для самостійного вирішення. Рекомендуються для студентів всіх форм навчання факультету АТЗ з напрямів підготовки «Автоматизація та автоматизація на транспорті» та «Спеціалізовані комп'ютерні системи» і слухачів НН ІППК.

Укладачі:

доценти О.Ю. Гуль,  
М.Р. Терованесов,  
асист. Т.О. Чуян,  
старш. викл. М.С. Курцев

Рецензент

доц. К.А. Трубчанінова

## ЗМІСТ

Розділ 1. Використання запитів .....	4
1.1 Запит на вибірку .....	4
1.2 Типи запитів на зміну.....	20
1.3 Властивості запитів .....	29
Розділ 2. Форми.....	32
2.1 Види форм .....	32
2.2 Створення форми.....	35
2.3 Підлеглі форми .....	47
2.4 Використання кнопок.....	49
2.5 Умовне форматування.....	52
Розділ 3. Звіти.....	55
3.1 Базові відомості про звіти.....	55
3.2 Створення звіту.....	55
Розділ 4. Друк в Access .....	65
4.1 Друк опису об'єктів БД .....	65
4.2 Друк даних у режимі таблиці .....	66
4.3 Друк форм .....	70
4.4 Друк звітів .....	71
Розділ 5.Завдання на виконання лабораторних робіт .....	73
Список літератури .....	85

## Розділ 1. ВИКОРИСТАННЯ ЗАПИТІВ

### 1.1 Запит на вибірку

Як правило, нема необхідності у відображенні всієї інформації, що зберігається в тій або іншій таблиці БД. У розділі 3 частини 1 методичних вказівок були розглянуті можливості з забезпечення необхідного режиму обробки даних, зокрема фільтрація, сортування, приховування стовпців, застосування підстановок, використання підтаблиць. Однак є більш потужний засіб обробки даних в одній або декількох таблицях, що дозволяє відображувати потрібну інформацію, що відповідає зазначеним вимогам. Наприклад, у БД **Освіта** необхідно знайти всіх студентів, які мають відмінні оцінки, або відобразити прізвища всіх студентів, які склали фізику. Для цього в Access використовуються *запити*.

Існує кілька типів запитів: на вибірку, на відновлення, на додавання, на видалення, перехресний і створення таблиці.

Найпоширенішим є запит на вибірку. При його виконанні створюється безліч записів, у яких утримуються зазначені дані, наприклад певні поля з однієї або декількох таблиць. При цьому інформацію, яка відображується в запиті, можна змінювати так само, як і при роботі зі звичайною таблицею. Однак слід зазначити, що названий набір записів створюється тимчасово, тільки під час виконання запиту, і не зберігається в БД, на відміну від вмісту таблиць.

Щоб створити новий запит, необхідно натиснути кнопку *Запити* у вікні БД **Освіта** і після цього вибрати режим створення запиту: з використанням конструктора або майстра. Також для створення запиту можна скористатися кнопкою *Створити*, після чого вибрати режим створення нового запиту. Існує кілька режимів створення запиту. Нижче буде докладно розглянуто кожний з них.

*Створення запиту в режимі конструктора.* Після вибору режиму конструктора буде відображено діалогове вікно *Додавання таблиці* для вибору таблиці (або таблиць), що буде служити джерелом даних при виконанні цього запиту. Як приклад буде розглянуто створення запиту на основі даних з

таблиці **Оцінки**. Після додавання таблиці **Оцінки** потрібно закрити вікно додавання таблиці, після чого на екрані буде відображено вікно конструктора нового запиту (рисунок 1), що зовні нагадує розглянуте раніше вікно розширеного фільтра.

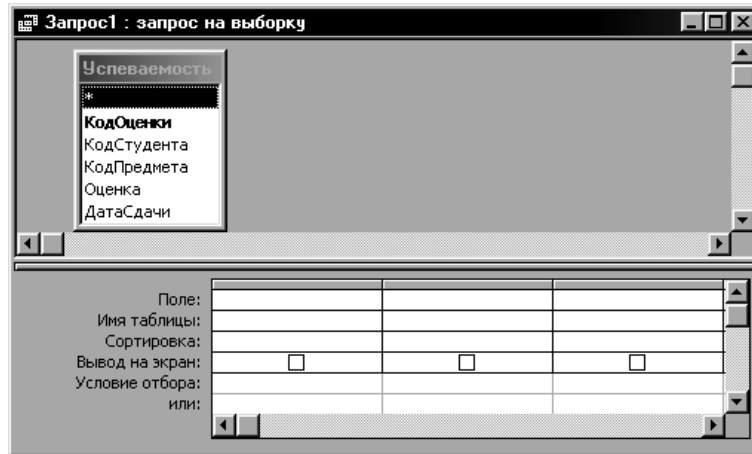


Рисунок 1 – Конструктор запитів

У верхній частині вікна конструктора розташовуються таблиці (або таблиця), що використовуються як джерело даних. У нижній частині вікна вказуються параметри даних, що відбирають у вибірку.

*Поле* – вказується назва поля, вміст якого буде додано в результати запиту. Щоб додати поле в запит, можна перетягнути його в рядок *Поле* з макета таблиці, розташованого у верхній частині вікна, або вибрати необхідне поле у списку, що випадає, який можна розкрити у рядку *Поле*. Щоб додати в запит всі поля з таблиці, необхідно двічі натиснути в таблиці на рядку з символом “\*”.

*Ім'я таблиці* – за допомогою списку вибирається назва таблиці, у якій перебуває поле, що використовується як джерело даних. Якщо вихідна таблиця одна, або назва поля є унікальною для всіх таблиць, то у використанні цього параметра нема необхідності, тому що його значення буде встановлюватися автоматично.

*Сортування* – вказується напрямок сортування для даного поля – за зростанням або за зменшенням.

*Виведення на екран* – задається режим, при якому на екрані буде відображуватися вміст обраного поля. На практиці можливі


випадки, коли певне поле бере участь у запиті, наприклад для виконання щодо нього якої-небудь умови, однак у результатах запиту це поле може і не відображатися.

*Умова відбору, або* – вказуються одна або кілька умов, відповідно до яких записи будуть додаватися у вибірку.

Наприклад, необхідно відобразити тільки оцінки “4” і “5”, отримані студентами (без зазначення прізвищ), і дату складання. При створенні такого запиту джерелом даних буде служити таблиця **Оцінки** (рисунок 1). У запиті будуть використовуватися поля **Оцінка** й **ДатаСкладання**, назви яких необхідно вибрати у рядку *Поле*. Назва таблиці в поле *Ім'я таблиці* буде встановлено автоматично. Для поля **ДатаСкладання** можна встановити напрямок сортування за *зростанням*. Обидва зазначених поля необхідно виводити на екран, тому для кожного з них потрібно встановити прапорець *Виведення на екран*. У рядку *Умова відбору* для поля **Оцінка** потрібно вказати таку умову: 4 OR 5 (рисунок 2) або вказати перше значення (4) у поле *Умова відбору*, а друге (5) – у поле *Або*.

Поле:	Оценка	ДатаСдачи	
Имя таблицы:	Успеваемость	Успеваемость	
Сортировка:		по возрастанию	
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Условие отбора:	4 Or 5		
или:			

Рисунок 2 – Запит на вибірку з таблиці Оцінки


Створений у такий спосіб запит необхідно виконати. Для цього потрібно натиснути кнопку **Запуск**  або виконати команду *Запит | Запуск*. У результаті буде отримано набір записів з необхідним набором полів, значення яких відповідають зазначеній умові (рисунок 3). Для збереження запиту необхідно виконати команду *Файл | Зберегти*, після чого вказати в діалоговому вікні ім'я нового запиту, наприклад **ОцінкаДата**. У результаті в списку запитів БД **Освіта** з'явиться запит **ОцінкаДата**, який можна відкрити для редагування кнопкою *Відкрити* або змінити запит за допомогою кнопки *Конструктор* вікна БД.

Запрос1 : запрос на выборку		
	Оценка	ДатаСдачи
▶	5	15.01.99
	4	15.01.99
	5	15.01.99
	4	20.01.99
	4	20.01.99
	5	20.01.99
	5	22.01.99
	4	25.01.00

Рисунок 3 – Результат виконання запиту

Розглянутий запит створювався на основі однієї таблиці. Існують також запити, які використовують дані з декількох таблиць. Такі запити називаються *багатотабличними*.

Наприклад, необхідно змінити запит **ОцінкаДата** таким чином, щоб відображувалися не тільки оцінки, але і прізвища студентів, які їх одержали, а також назви відповідних предметів. При цьому в запиті повинні відображатися тільки оцінки з предмета Фізика.

Для цього необхідно відкрити запит **ОцінкаДата** в режимі конструктора і додати в макет запиту ще дві таблиці, які містять необхідну інформацію: **Студенти** і **Предмети**. Щоб додати таблицю в макет запиту, необхідно виконати команду *Запит | Додати таблицю* або натиснути кнопку *Відобразити таблиці* , і за допомогою діалогового вікна *Додавання таблиці* по черзі додати таблиці **Студенти** і **Предмети** в макет. При цьому на схемі будуть автоматично відображені зв'язки між зазначеними таблицями. Слід зазначити, що в макеті запиту можна визначити зв'язки між таблицями, якщо вони не були раніше встановлені. Для цього потрібно виділити назву поля зв'язку в одній таблиці і перетягнути його на макет іншої таблиці, аналогічно роботі зі схемою даних.

Після додавання всіх таблиць у нижній частині вікна запиту можна перед полем **Оцінка** вставити два стовпці – для прізвищ студентів і назв предметів. При перетаскуванні поля з таблиці, розташованої в макеті, у рядок *Поле* автоматично з'являється новий стовпчик, у якому можна вказувати параметри даного поля. Для поля **Прізвище** з таблиці **Студенти** необхідно вказати напрямок сортування за зростанням, а для поля **Назва** з таблиці

**Предмети** – умова “Фізика”. При цьому сортування для поля **ДатаСкладання** можна скасувати, вибравши в поле *Сортування* значення (*відсутній*) (рисунк 4).

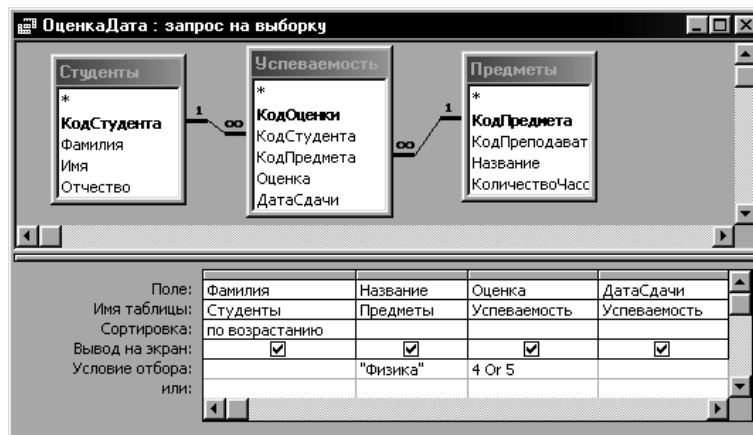



Рисунок 4 – Запит на вибірку з таблиць Студенти, Предмети і Оцінки

Створений запит необхідно виконати, скориставшись кнопкою *Запуск* . У результаті будуть отримані записи з зазначеним набором полів з обраних таблиць. При цьому дані відображуються тільки для предмета Фізика (рисунк 5).

	Фамилия	Название	Оценка	ДатаСдачи
▶	Жукова	Физика	5	15.01.99
	Игнатов	Физика	5	15.01.99
	Морозов	Физика	4	15.01.99

Рисунок 5 – Результат виконання запиту

Отриманому багатотабличному запиту можна дати назву *ОцінкиЗФізики*, скориставшись для цього командою *Файл |Зберегти як*.

*Створення запиту за допомогою майстра.* Щоб запустити майстер запитів, треба у вікні БД виконати подвійне натискання на пункті *Створення запиту за допомогою майстра* в розділі *Запити*. При цьому першим кроком у створенні запиту є вибір полів, які будуть у ньому відображуватися, з наявних у БД таблиць і запитів. Наприклад, необхідно створити запит, що містить прізвища викладачів і назви предметів, які вони



викладають. Для цього потрібно спочатку у списку, що розкривається, *Таблиці і запити* вибрати таблицю **Викладачі**, і додати поле **Прізвище** в розділ *Вибрані поля*. Потім необхідно вибрати таблицю **Предмети** і додати в запит поле **Назва** (рисунок 6).

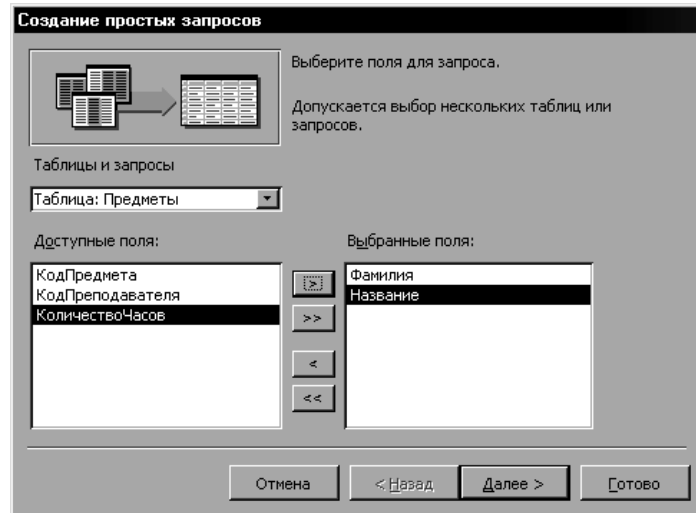


Рисунок 6 – Майстер запитів. Вибір полів

Після натискання кнопки *Далі >* буде відображено завершальне вікно майстра, у якому необхідно вказати назву створеного запиту і визначити подальший режим роботи з ним – перегляд даних або зміну макета. Запиту можна дати назву **ВикладачіПредмети**, і після вибору режиму перегляду даних і натискання кнопки *Готово* будуть відображені записи створеного запиту, що містять прізвища викладачів і назви предметів (рисунок 7).

ПреподавателиПредметы : запрос на выборку		
	Фамилия	Название
▶	Викулина	Физика
	Костыркин	Химия
	Казанко	Математика
	Позднякова	Экономика
	Загарийчук	Философия

Рисунок 7 – Запит ВикладачіПредмети

*Поля, що обчислюються.* При створенні запитів часто виникає необхідність не тільки використати наявні поля таблиць, але й створювати на їхній основі інші поля, які називаються полями, що обчислюються. Наприклад, якщо в таблиці будь-якого магазину є поле ціни на товар і поле кількості цього товару, то виходячи з цього можна створити поле, що обчислюється, у якому буде підраховуватися загальна вартість для кожного товару шляхом перемножування значень ціни і кількості. Інакше кажучи, в полі, що обчислюється, можуть використовуватися арифметичні оператори.

З іншого боку, при створенні поля, що обчислюється, на основі вмісту текстових полів, як правило, використовується операція об'єднання текстових значень, що називається *конкатенацією*. У таблиці 1 наведено оператори, які використовуються в Access при побудові виразів.

Таблиця 1 – Основні оператори, що використовуються при створенні виразів

Оператор	Опис
+	Додавання чисел
–	Віднімання чисел
*	Множення чисел
/	Ділення чисел
\	Ділення першого числа на друге з округленням результату
^	Зведення в степінь
MOD	Обчислення залишку від цілочисельного ділення
&	Конкатенація рядків

Очевидно, що розглянуті оператори надають досить вузький спектр можливостей для створення виразів. Щоб створити складний вираз, використовують потужний засіб, що називається *Будівник виразів*. Ця програма дозволяє побудувати вираз будь-якої складності з використанням найрізноманітніших функцій (рисунок 8).

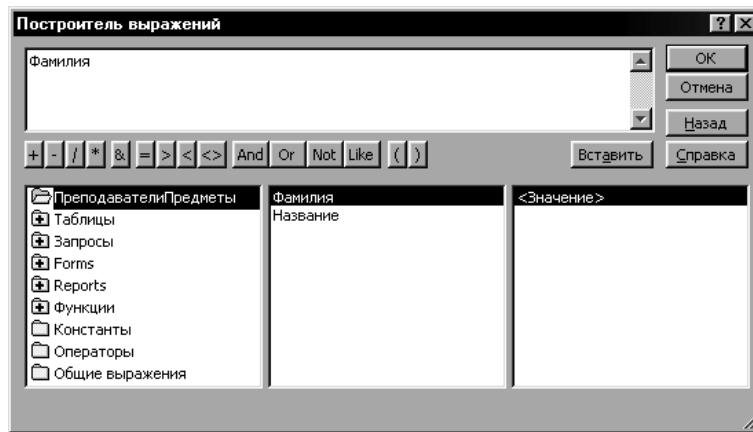


Рисунок 8 – Будівник виразів

За допомогою *Будівника виразів* можна виконувати різні обчислення з використанням даних, що перебувають у БД – полів таблиць, запитів, форм і звітів. Всі перераховані об'єкти БД перебувають відповідно у папках *Таблиці*, *Запити*, *Форми та Звіти* у лівій частині *Будівника*. Щоб вибрати необхідний об'єкт, наприклад таблицю, необхідно виконати подвійне натискання на папці *Таблиці*, у результаті чого розкриється список всіх таблиць БД. Після вибору потрібної таблиці (або іншого об'єкта) у середній частині *Будівника* відображується список її полів, значення яких можна включити у створюваний вираз. Для цього необхідно вибрати необхідне поле й натиснути у вікні *Будівника* кнопку *Вставити*.

При побудові виразів, крім назв полів, можуть використовуватись різні оператори (таблиця 1), додавання яких виконується при натисканні відповідних кнопок у вікні *Будівника*. Також існує можливість використання у виразі різних функцій Access. Для цього в лівій частині вікна необхідно відкрити папку *Функції* і вибрати пункт *Вбудовані функції*. Потім треба в середній частині *Будівника* визначити категорію, у якій перебуває необхідна функція, і вибрати після цього у правій частині вікна назву потрібної функції (рисунки 9, 10). Створюваний у такий спосіб вираз відображується у верхній частині вікна *Будівника виразів*.

Наприклад, необхідно в розглянутому вище запиті **ВикладачіПредмети** замість поля **Прізвище** додати поле, що обчислюється, у якому містились би прізвище та ініціали викладача.

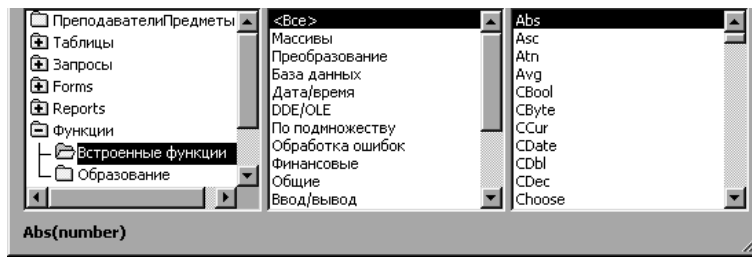


Рисунок 9 – Будівник виразів. Вбудовані функції

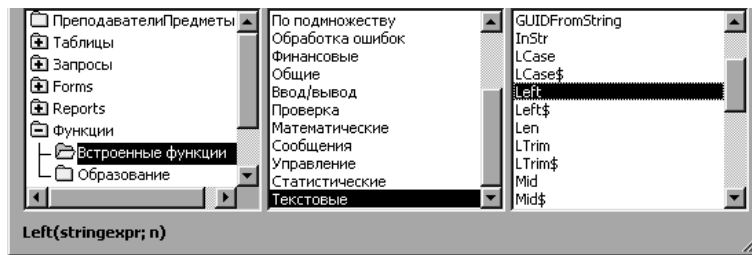



Рисунок 10 – Будівник виразів. Текстова функція Left

Для цього необхідно відкрити запит у режимі конструктора. Потім встановити курсор у рядок *Поле* першого стовпчика, тобто в поле *Прізвище*. Після цього необхідно скористатися кнопкою *Побудувати* , у результаті чого буде відображено вікно *Будівника виразів* (рисунок 8). Необхідно відзначити той факт, що за допомогою кнопки *Побудувати* можна як створити новий вираз, так і змінити вже існуючий, якщо він перебуває в поточному полі.

Назви полів у виразі повинні вказуватися у квадратних дужках, однак якщо поле буде зазначено без дужок і в імені поля відсутні пробіли, знаки арифметичних операцій та деякі інші спеціальні символи, то Access ці дужки вставить автоматично. За наявності в іменах полів вищезазначених символів квадратні дужки необхідно вставляти вручну, бо інакше СУБД може розставити ці дужки некоректно.

Для створення текстового рядка з прізвищем та ініціалами необхідно кілька разів послідовно виконати операцію конкатенації такої структури: *Прізвище & Символ Пробілу&Ініціал імені&Символ крапки&Ініціал по батькові&Символ крапки*. Всі текстові символи вказуються в подвійних лапках, тому структура виразу набуде такого вигляду: *Прізвище & “ ” &Ініціал імені& “.” &Ініціал по батькові& “.”*.

Таким чином, початок виразу буде таким: Прізвище & " " &.

Для роботи з текстовими значеннями використовуються функції категорії *Текстові*. Щоб одержати ініціал імені, необхідно скористатися однією з функцій даної категорії, а саме функцією *Left*, що повертає перші *n* символів зазначеного рядка. Щоб додати функцію у вираз, необхідно двічі натиснути на папці *Функції* і вибрати після цього розділ *Вбудовані функції* (рисунок 9).

Потім у другому стовпчику необхідно вибрати необхідну категорію, у цьому випадку *Текстові*, і після цього в третьому стовпчику вибрати потрібну функцію, тобто *Left* (рисунок 10).

Після того як функція обрана, необхідно встановити курсор у те місце виразу, куди вона повинна бути вставлена, і натиснути кнопку *Вставити* у вікні Будівника виразів.

У результаті цих дій у вираз буде додано шаблон функції *Left: Left («stringexpr»; «n»)*, у якому необхідно вказати значення необхідних параметрів. Зокрема, замість *stringexpr* потрібно вказати поле *Ім'я*, а замість *n* – кількість символів, які необхідно одержати з рядка, тобто 1. Таким чином, текст виразу має набути такого вигляду: Прізвище & " " &Left (Ім'я; 1).

Слід зазначити, що для поля **Ім'я** не вказувалася назва таблиці **Викладачі**, тому що воно було обрано раніше в рядку *Ім'я таблиці* вікна конструктора запитів. З іншого боку, щоб додати у вираз ім'я поля конкретної таблиці, можна розкрити папку *Таблиці* і вибрати в таблиці **Викладачі** поле **Ім'я** (рисунок 11), після чого встановити курсор у відповідне місце виразу і натиснути кнопку *Вставити*. Таким чином, назву поля з зазначенням імені таблиці необхідно відображувати в такому форматі: *[Ім'я таблиці]![Назва поля]*.

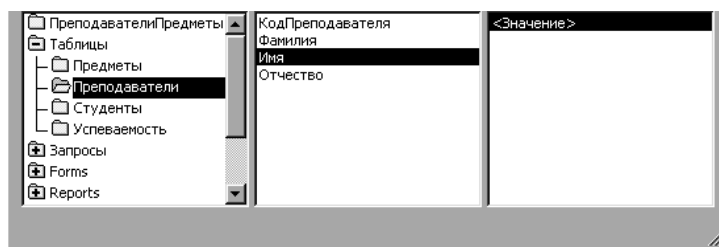



Рисунок 11 – Будівник виразів. Поле *Ім'я* таблиці *Викладачі*


Після визначення першого ініціала необхідно поставити крапку, тобто вказати у виразі рядок “.”. Потім аналогічно додати

ініціал **по батькові** і знову поставити крапку. Таким чином, буде отримано такий вираз:

Прізвище & " " & Left(Ім'я;1) & "." & Left (ПоБатькові; 1) & "."

Щоб повернутися у вікно конструктора запитів, необхідно натиснути кнопку *ОК*. У результаті створений вираз буде перебувати в рядку *Поле* першого стовпчика запиту. Для виконання запиту необхідно скористатися кнопкою *Запуск* , після чого в запиті буде створено поле, що обчислюється, *Вираз1*, яке міститиме прізвище та ініціали викладача.

Назви полів запиту формуються автоматично. Для полів таблиць назви збігаються з назвами відповідних полів запиту (наприклад, поле *Назва*). З іншого боку, назви полів, що обчислюються, формуються автоматично: *Вираз1*, *Вираз2* і т. д. Очевидно, що такі назви не є інформативними.

Щоб встановити підпис для поля в таблиці або запиті, необхідно в режимі конструктора встановити курсор у необхідне поле (зокрема в поле *Вираз1*) і натиснути кнопку *Властивості* . Після цього в діалоговому вікні *Властивості поля* можна на вкладці *Загальні* вказати в рядку *Підпис* і назва поля, що буде використовуватися замість вихідного в режимі редагування запиту (або таблиці), а також у формах і звітах. Зокрема для поля, що обчислюється, *Вираз1* можна вказати підпис ПІБ (рисунок 12).

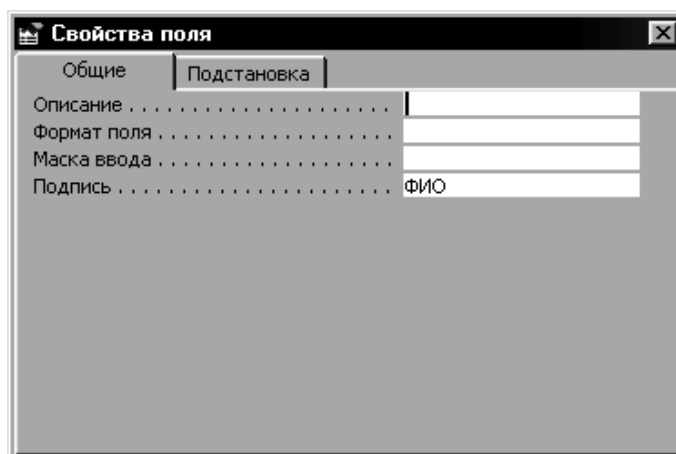




Рисунок 12 – Діалогове вікно *Властивості поля*

Таким чином, був створений і виконаний запит на вибірку з використанням поля, що обчислюється (рисунок 13).

ПреподавателиПредметы : запрос на выборку	
ФИО	Название
Викулина В.И.	Физика
Костыркин О.В.	Химия
Казанко В.В.	Математика
Позднякова Л.А.	Экономика
Загарийчук И.Д.	Философия

Рисунок 13 – Запит на вибірку з полем, що обчислюється

*Групові операції.* При обробці даних у таблицях найчастіше буває необхідно враховувати в запитах не окремі записи, а підсумки, які обчислюються або формуються певним чином для різних груп записів у таблиці. Наприклад, необхідно виконати запит, у якому для кожного студента була б підрахована кількість отриманих ним оцінок, а також обчислений середній бал. І потім на підставі середнього бала визначена винагорода, наприклад з розрахунку 10 грн за кожен бал. Інакше кажучи, для відмінника (середній бал = 5) розмір винагороди повинен скласти 50 грн.

Щоб виконати подібні обчислення, необхідно створити новий запит у режимі конструктора, додати в нього за допомогою кнопки *Відобразити таблицю*  таблиці **Студенти** й **Оцінки**, і потім скористатися кнопкою *Групові операції* . Після цього серед параметрів запити з'явиться рядок *Групові операції*. Щоб обчислити для кожного студента його середній бал, необхідно спочатку додати в запит поле **Прізвище** таблиці **Студенти**. При цьому в рядку *Групові операції* за замовчуванням встановлюється значення *Групування*.


Отже для кожного прізвища студента в другому стовпці необхідно розташувати загальну кількість оцінок, отриманих кожним студентом, у третьому – середній бал, і в четвертому – розмір нарахованої стипендії. Для цього у другий і третій стовпці запити необхідно додати поле **Оцінка** таблиці **Оцінки**, тому що на підставі саме цього поля буде визначатися загальна кількість оцінок, а також обчислюватись середній бал кожного студента.

При виконанні групових операцій можна використовувати так звані підсумкові функції (таблиця 2), які необхідно вибирати зі списку в доданому полі *Групові операції*.

Таблиця 2 – Основні підсумкові функції

Функція	Опис
Sum	Підсумовування значень у групі
Avg	Середнє значення для групи
Min	Мінімальне значення для групи
Max	Максимальне значення для групи
Count	Підрахунок числа значень у групі
First	Значення поля в першому записі групи
Last	Значення поля в останньому записі групи

Для підрахунку кількості отриманих оцінок необхідно вибрати функцію Count, а для визначення середнього бала – функцію Avg. Потім створений запит необхідно зберегти, вказавши для нього ім'я **СереднійБал**.

Щоб обчислити розмір винагороди, необхідно створити вираз за допомогою *Будівника виразів*. Для цього спочатку необхідно вибрати в рядку *Групові операції* четвертого стовпчика запиту за допомогою списку, що розгортається, пункт *Вираз*. Потім потрібно, перебуваючи в цьому полі, натиснути кнопку *Побудувати*  і вказати в *Будівнику виразів* формулу для обчислення розміру винагороди. У даній формулі буде використовуватися поле групової операції з функцією Avg, отримане раніше. Слід зазначити, що подібно до полів у запитах полів, що обчислюються, Access автоматично присвоює імена. Зокрема, поле, що містить кількість оцінок, було названо **Count-Оцінка**, а поле з середнім балом було названо **Avg-Оцінка**. Отже, у вікні *Будівника виразів* необхідно у формулі для стипендії використати поле **Avg-Оцінка** збереженого запиту **СереднійБал**. Тобто потрібно вставити у формулу поле **Avg-Оцінка** з папки **СереднійБал** і помножити це значення на 10 для визначення розміру винагороди (рисунок 14). Однак необхідно мати на увазі, що якби розглянутий запит не був збережений, то в *Будівнику виразів* не з'явилися б поля **Count-Оцінка** і **Avg-Оцінка**.

Після установаження всіх описаних параметрів макет запиту буде мати вигляд, який подано на рисунку 15.



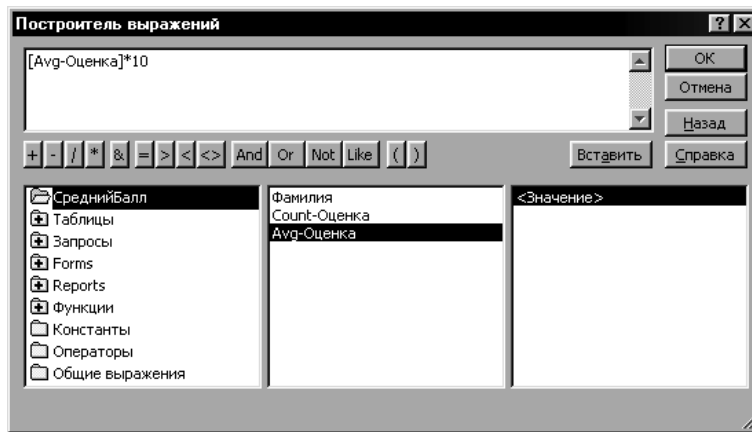



Рисунок 14 – Будівник виразів. Визначення розміру винагороди



Рисунок 15 – Макет запису з використанням групових операцій

При виконанні створеного в такий спосіб запису буде визначена кількість оцінок, середній бал і розмір винагороди для кожного студента. Однак назви полів із груповими операціями (кількість оцінок і середній бал) і виразу (винагорода) були створені автоматично і не є достатньо інформативними. Тому необхідно змінити властивості полів, встановивши їм підписи.

Для цього необхідно відкрити запит у режимі конструктора і у вікні властивостей (кнопка ) для кожного з цих полів вказати в рядку підпис назви **КількістьОцінок**, **СереднійБал** і **Винагорода** відповідно.

Результатом виконання запису **СереднійБал** буде набір записів, поданий на рисунку 16.

СреднийБалл : запрос на выборку				
	Фамилия	КоличествоОценок	СреднийБалл	Стипендия
▶	Жукова	2	5	50
	Игнатов	2	4,5	45
	Морозов	2	4	40
	Петров	3	4	40

Рисунок 16 – Результат виконання запиту СереднійБалл

*Перехресні запити.* В Access є можливість створення запитів, у яких використовуються групові операції, у вигляді електронних таблиць. Тобто як назви рядків і стовпців такої «таблиці» використовуються ті або інші поля таблиць, а в її «чарунках» будуть відображуватися значення, що обчислюються, які відповідають значенням, зазначеним у назві рядка й стовпчика. Як приклад подібної електронної таблиці можна розглянути набір оцінок, які одержав кожний студент з усіх предметів. Інакше кажучи, заголовками рядків такої таблиці будуть прізвища студентів, заголовками стовпців – назви предметів, а на перетині рядків і стовпців буде відображуватися оцінка того або іншого студента з даного предмета аналогічно розташуванню даних у робочих аркушах Excel. При цьому слід зазначити, що значення, які обчислюються, не можуть бути відредаговані, тому використання перехресного запиту з метою зміни даних не має рації. Однак перехресні запити можуть виявитися досить корисними при створенні різноманітних звітів.

Для створення перехресного запиту, що відображує таблицю оцінок студентів, необхідно спочатку створити новий запит у режимі конструктора. Потім потрібно в макет запиту додати таблиці, у яких містяться вихідні дані запиту: **Студенти**, **Оцінки** і **Предмети**. Після цього необхідно виконати команду *Запит | Перехресний*, у результаті чого в макеті запиту серед параметрів з'явиться рядок *Перехресна таблиця*, у якій для полів запиту можна за допомогою списку, що розгортається, вибрати одне з таких значень:

*Заголовки рядків* – вибирається для поля запиту, значення якого потрібно використати як заголовки рядків. Як заголовки рядків необхідно визначити хоча б одне поле, при цьому в рядку *Групові операції* необхідно вибрати значення *Групування*, одну з підсумкових функцій або *Вираз*.

*Заголовки стовпців* – вибирається для поля запиту, значення якого потрібно використати як заголовки стовпців. З цією метою використовується тільки одне поле, вимоги до якого аналогічні вимогам до поля, у якому встановлено значення *Заголовки рядків*.

*Значення* – вибирається для поля запиту, у якому обчислюється підсумкове значення, відображуване в “чарунках” перехресного запиту. Таке поле має бути єдиним, при цьому в рядку *Групові операції* для нього необхідно вибрати одну з підсумкових функцій або задати вираз, у якому використовуються підсумкові функції.

*(не відображується)* – вибирається для того поля, значення якого не повинні відображатися в перехресному запиті (використовується для задавання деяких умов відбору даних).

У першому стовпці запиту потрібно вибрати поле **Прізвище** таблиці **Студенти**, при цьому необхідно задати для нього значення *Групування* в полі *Групова операція*, а також *Заголовки рядків* у полі *Перехресна таблиця*. Також можна вибрати напрямок сортування за зростанням.

Для другого стовпчика запиту необхідно вибрати поле **Назва** таблиці **Предмети** і встановити для нього значення *Групування* в полі *Групова операція* і *Заголовки стовпців* у полі *Перехресна таблиця*.

У третьому стовпці запиту потрібно вибрати поле **Оцінка** таблиці **Оцінки** і задати для нього функцію *Sum* у полі *Групова операція*, а також *Значення* в полі *Перехресна таблиця*.

Створений запит можна зберегти під ім'ям **ОцінкиЗаПредметами** (рисунок 17).



Рисунок 17 – Макет перехресного запиту **ОцінкиЗаПредметами**

У результаті виконання перехресного запиту **ОцінкиЗаПредметами** буде отримана електронна таблиця, яка містить оцінки студентів з усіх предметів (рисунок 18). Даний запит може бути використаний, наприклад, у звіті, що повинен відображувати підсумки минулої сесії.

ОценкиПоПредметам : перекрестный запрос						
	Фамилия	Математика	Физика	Философия	Химия	Экономика
▶	Жукова		5		5	
	Игнатов		5		4	
	Морозов		4		4	
	Петров	5		3		4

Рисунок 18 – Результат виконання запиту **ОцінкиЗаПредметами**

## 1.2 Типи запитів на зміну

Як правило, при роботі з таблицями використовується запит на вибірку. Однак у багатьох випадках необхідно за допомогою запитів виконувати різні зміни в БД: створювати нові таблиці, оновлювати, додавати і видаляти записи в існуючих таблицях. Для цього використовуються *запити на зміну*, які можна розділити відповідно на такі категорії: *створення таблиці*, *оновлення*, *додавання* і *видалення*. Слід зазначити, що перед створенням будь-якого запиту на зміну спочатку створюється запит на вибірку з необхідною структурою, після чого вибирається необхідний тип запиту.

*Створення таблиці.* У тому випадку, коли при багаторазовому виконанні запиту на вибірку відбувається відбір тих самих даних, то їх доцільно оформити у вигляді нової таблиці за допомогою запиту на створення. Такий запит буде особливо ефективним, якщо процес одержання даних досить трудомісткий, тобто коли доводиться виконувати велику кількість обчислень або працювати з великими обсягами інформації.

Щоб створити за допомогою запиту нову таблицю, необхідно за допомогою конструктора побудувати запит на вибірку, після чого виконати команду *Запит | Створення таблиці* або скористатися кнопкою *Тип запиту*, що відображує останній обраний тип запиту. Вибір типу запиту виконується у списку, що розгортається, кнопки, зображеної на рисунку 19.

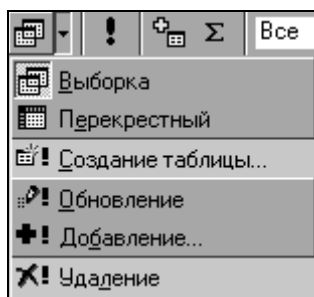


Рисунок 19 – Кнопка *Тип запиту*

Якщо обрано режим *Створення таблиці*, то в цьому випадку з'явиться діалогове вікно *Створення таблиці* (рисунок 20), у якому необхідно вказати ім'я нової таблиці, а також її розташування, тобто таблиця буде створена в поточній БД або в тій, яка зазначена в рядку *Ім'я файлу*.

Наприклад, необхідно створити нову таблицю на підставі розглянутого раніше запиту на вибірку **ВикладачіПредмети**. Для цього необхідно відкрити даний запит у режимі конструктора і вибрати тип запиту *Створення таблиці*. Потім необхідно вказати ім'я створюваної таблиці, наприклад **ПредметиВикладачі**, і натиснути кнопку *ОК*.

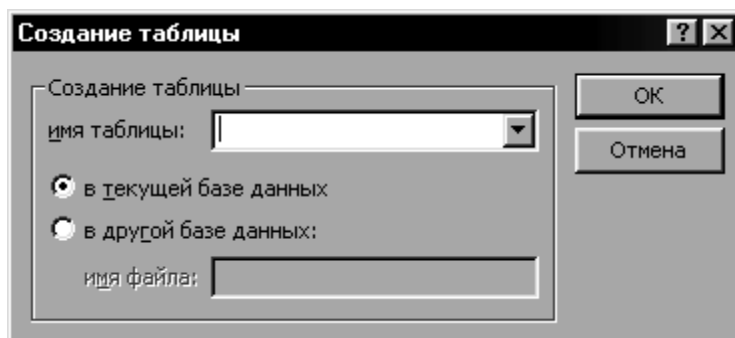



Рисунок 20 – Діалогове вікно *Створення таблиці*

При виконанні даного запиту за допомогою кнопки *Запуск*  буде відображено вікно попередження про створення нової таблиці і розташування в ній обраних записів. Після підтвердження, тобто вибору кнопки *Так*, у БД **Освіта** буде створена нова таблиця **ПредметиВикладачі** (рисунок 21).

ПредметыПреподаватели : таблица		
	Выражение1	Название
▶	Викулина В.И.	Физика
	Костыркин О.В.	Химия
	Казанко В.В.	Математика
	Позднякова Л.А.	Экономика
	Загарийчук И.Д.	Философия

Рисунок 21 – Нова таблиця ПредметиВикладачі

Слід зазначити, що в БД не може бути двох об'єктів з однаковими іменами (це стосується тільки таблиць і запитів). Тому якщо в описаному прикладі новій таблиці було б присвоєно ім'я **ВикладачіПредмети**, що збігається з ім'ям запиту, то було б відображено попередження про те, що об'єкт з таким ім'ям у цій базі вже є.

Оновлення. Даний тип запиту на зміну використовується в тих випадках, коли необхідно виконувати відбір записів з наступною зміною для них значення певного поля.

Як приклад можна розглянути таблицю **Студенти** в БД **Освіта**. Кожний студент у цій БД може перебувати в одному з чотирьох станів: *навчається, в академічній відпустці, відрахований, закінчив*. Щоб мати можливість змінювати стан студента, тобто відраховувати, відновлювати, відправляти в академічну відпустку або “повертати” з відпустки, можна в таблицю **Студенти** додати поле **Стан**. Якщо студент навчається, то це поле має бути порожнім. Якщо ж студент відрахований, то необхідно в полі **Стан** вказати ознаку «*в*», якщо в академічній відпустці – ознаку «*а*», якщо закінчив – ознаку «*з*».

Для додавання нового поля необхідно відкрити таблицю **Студенти** в режимі конструктора. Потім задати нове поле – **Стан**, для якого необхідно вибрати тип *Текстовий* і визначити такі властивості:

*Розмір поля:* 1;

*Умова на значення:* "" Or "а" Or "в" Or "з";

*Порожні рядки:* Так.

У полі **Стан** можуть перебувати як символи “*а*”, “*в*”, або “*з*”, так і порожній рядок, тобто “”. Тому у властивості *Умова на значення* необхідно вказати, що в дане поле може бути введено тільки одне з цих значень (або не введено нічого). Крім того, у

властивості *Порожні рядки* необхідно вказати, що в полі **Стан** можуть зберігатися порожні рядки.

Після установавлення властивостей і переходу в режим таблиці буде відображено попередження про можливе недотримання цілісності даних, установленої при створенні зв'язків між таблицями. У цьому випадку можна не виконувати перевірку на цілісність, тому що її порушення відбулося через додавання нового поля, що не повинно привести до небажаних змін у БД.

У режимі таблиці можна внести деякі зміни в поле **Стан** для окремих студентів (таблиця 3). При цьому слід зазначити, що при спробі введення в поле **Стан** значення, що не відповідає зазначеній умові, буде показано вікно попередження. Також необхідно мати на увазі, що при перевірках і обчисленнях букви, які виглядають однаково в латиниці і кирилиці, розрізняються. Інакше кажучи, якщо ввести в поле **Стан** латинську букву “а”, то Access це сприйме як спробу запису в поле помилкового значення.

У подальших міркуваннях можна зробити допущення, що на початку кожного навчального року всі “академісти” повертаються до навчання, тому для них необхідно в полі **Стан** видаляти ознаку “а”. У подібних ситуаціях зручно використовувати запит на оновлення, у якому для всіх записів, у яких значення поля **Стан** дорівнює “а”, буде виконано видалення цієї ознаки.

Таблиця 3 – Таблиця Студенти з новим полем Стан

КодСтудента	Прізвище	Ім'я	По батькові	Стан
1	Жукова	Марина	Володимирівна	
2	Морозов	Ігор	Олегович	
3	Петров	Михайло	Олександрович	
4	Ігнатов	Максим	Валерійович	
5	Волков	Юрій	Олексійович	а
6	Волошина	Вікторія	Іванівна	
7	Литвинов	Євгеній	Валерійович	в
8	Гущина	Олена	Валеріївна	а
9	Кузнецов	Руслан	Олексійович	в
10	Семененко	Світлана	Станіславівна	в

Для виконання описаних дій необхідно створити новий запит у режимі конструктора і додати в макет таблицю **Студенти**. Потім по черзі додати в запит всі поля цієї таблиці, причому для поля **Стан** вказати параметр *Умова відбору*: "а". Після цього можна виконати даний запит, щоб переконатися в його функціональності. Заключним етапом у створенні запиту буде зміна його типу з *Вибірка* на *Оновлення* кнопкою *Тип запиту* (рисунок 19), після чого в макеті з'явиться новий параметр – *Оновлення*. Для поля **Стан** необхідно вказати *Оновлення*: "" (порожній рядок), щоб для всіх студентів-академістів була вилучена ознака "а". Отриманий у такий спосіб запит (рисунок 22) можна зберегти під ім'ям **Академісти**.

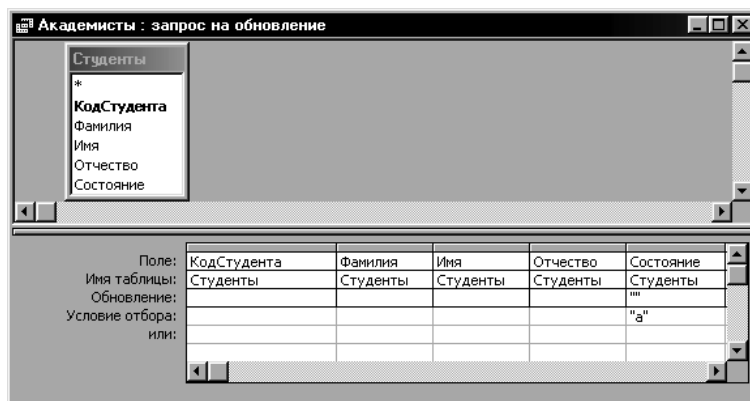


Рисунок 22 – Макет запиту Академісти

При виконанні запиту **Академісти** в цьому випадку буде відображено повідомлення про зміну 2 записів, що відповідає поточній кількості академістів. Після підтвердження оновлення записів всі значення, рівні "а", будуть вилучені з поля **Стан** (рисунок 23).

Слід також зазначити, що в макеті запиту **Академісти** після того, як його тип був змінений на *Оновлення*, автоматично будуть вилучені всі поля, що не оновлюються, тобто залишиться тільки поле **Стан**, яке буде оновлюватись.



Студенты : таблица				
КодСтудента	Фамилия	Имя	Отчество	Состояние
1	Жукова	Марина	Владимировна	
2	Морозов	Игорь	Олегович	
3	Петров	Михаил	Александрович	
4	Игнатов	Максим	Валерьевич	
5	Волков	Юрий	Алексеевич	
6	Волошина	Виктория	Ивановна	
7	Литвинов	Евгений	Валерьевич	о
8	Гущина	Елена	Валерьевна	
9	Кузнецов	Руслан	Алексеевич	о
10	Семенов	Светлана	Станиславовна	о

Рисунок 23 – Оновлена таблиця Студенти (без академістів)

*Додавання.* Іноді виникає необхідність тимчасового зберігання якихось даних з можливістю їхнього відновлення. Наприклад, з таблиці **Студенти** необхідно час від часу видаляти відрахованих студентів. Однак відрахований студент може бути відновлений, і тому дані про всіх відрахованих студентів можна зберігати в архівній таблиці **СтудентиВідраховані**, і якщо виникне потреба у відновленні інформації в БД про того або іншого студента, можна буде просто скопіювати відповідний запис із таблиці **СтудентиВідраховані** у таблицю **Студенти**.

Для обробки подібного архіву спочатку необхідно створити в БД **Освіта** нову таблицю **СтудентиВідраховані**, структура якої буде збігатися зі структурою **Студенти**. Тому найкраще в даній ситуації скористатися операцією копіювання. Тобто потрібно у вікні БД **Освіта** виділити таблицю **Студенти**, натиснути кнопку *Копіювати*, і потім натиснути кнопку *Вставити*. Після цього буде відображено вікно *Вставка таблиці*, у якому необхідно вказати ім'я нової таблиці: **СтудентиВідраховані**, і вибрати режим вставки *тільки структура* (рисунок 24).

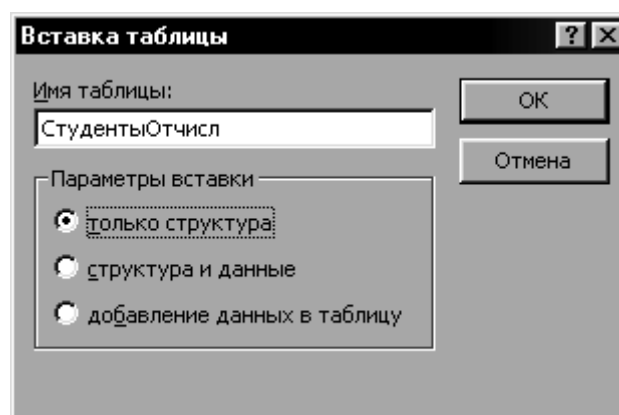


Рисунок 24 – Діалогове вікно *Вставка таблиці*

Потім необхідно створити новий запит у режимі конструктора, додати до його макета таблицю **Студенти** і по черзі додати в структуру запиту всі поля, крім **КодСтудента**, тому що значення поля **КодСтудента** формуються автоматично, і їх не потрібно переносити з таблиці **Студенти** в таблицю **СтудентиВідраховані**. З іншого боку, через те, що поле **КодСтудента** є первинним ключем, то при спробі повторного копіювання тих самих студентів буде виникати помилка, тому що значення первинного ключа повинне бути унікальним для кожного запису.

Після цього необхідно вибрати тип запиту *Додавання*, у результаті чого буде відображено діалогове вікно *Додавання*, у якому необхідно в якості “приймача” даних вибрати таблицю **СтудентиВідраховані** (рисунок 25).

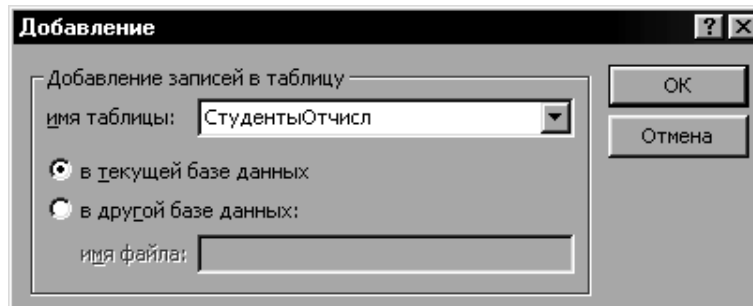


Рисунок 25 – Діалогове вікно *Додавання*

При цьому в макеті запиту з'явиться новий рядок *Додавання*, у який автоматично будуть додані імена відповідних полів таблиці **СтудентиВідраховані**. Потім треба для поля **Стан** вказати параметр *Умова відбору*: "в" і зберегти створений запит під ім'ям **ДодаванняВідрахованих** (рисунок 26).

Тепер при кожному виконанні запиту **ДодаванняВідрахованих** у таблицю **СтудентиВідраховані** будуть додаватися відраховані студенти.

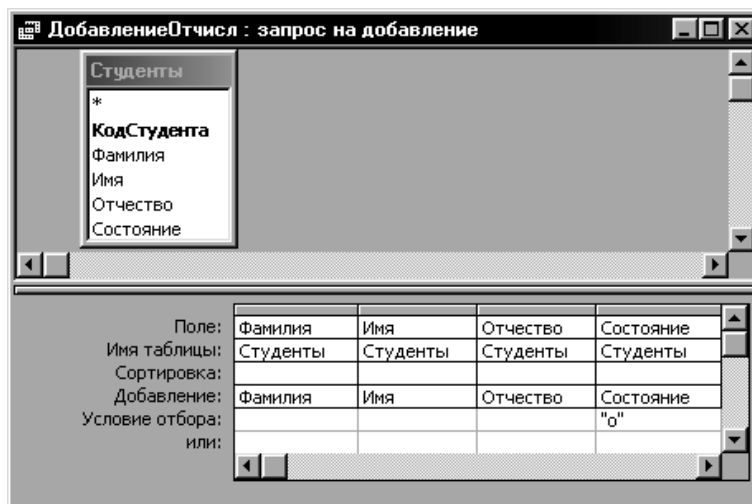


Рисунок 26 – Макет запиту ДодаванняВідрахованих

*Видалення.* Поряд з додаванням записів часто виникає необхідність у видаленні записів з таблиці при виконанні запиту. Наприклад, у розглянутому випадку з відрахованими студентами при кожному новому виконанні запиту **ДодаванняВідрахованих** будуть додаватися вже скопійовані раніше записи, тому що умовою для запиту є тільки наявність ознаки “**в**” у полі **Стан**. Тому зручним доповненням до запиту на додавання є *запит на видалення*, за допомогою якого в цьому випадку можна після додавання в таблицю **СтудентиВідраховані** видалити всіх відрахованих студентів з таблиці **Студенти**.

При створенні запиту на видалення необхідно мати на увазі, що з записами, що видаляються, можуть бути зв'язані дані з іншої таблиці. Наприклад, студенти **Жукова** і **Морозов** мають кілька оцінок, тобто існує безпосередній зв'язок між даними в таблицях **Студенти** і **Оцінки**. Інакше кажучи, неможливо просто видалити записи для цих студентів з таблиці **Студенти**, тому що в цьому випадку для декількох записів з таблиці **Оцінки** у полі **КодСтудента** буде утримуватися посилання на неіснуючі записи в таблиці **Студенти**.

Для запобігання подібної ситуації необхідно перевірити, який вид зв'язку є між полями **КодСтудента** таблиць **Студенти** і **Оцінки**, і переконатися, що для цього зв'язку в діалоговому вікні *Зміна зв'язків* (рисунок 27) встановлено режим *каскадне видалення зв'язаних записів*.

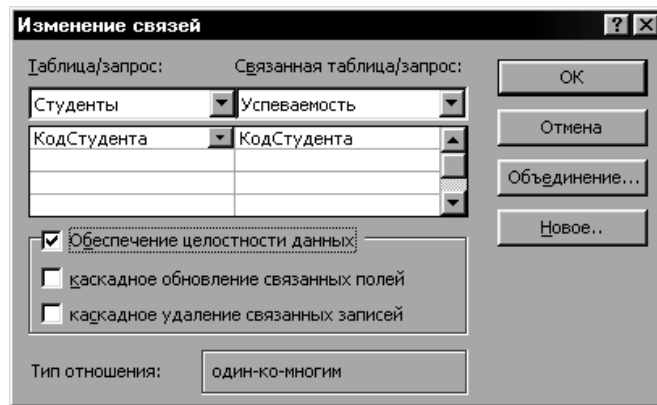


Рисунок 27 – Діалогове вікно *Зміна зв'язків*

Створення запиту виконується так само, як і в попередніх випадках. Для простоти можна в цьому випадку скористатися готовим запитом **ДодаванняВідрахованийих**, зберігши його під новим ім'ям **ВидаленняВідрахованийих** (рисунок 28). При цьому необхідно встановити тип запиту *Видалення* за допомогою кнопки *Тип запиту* (рисунок 19).

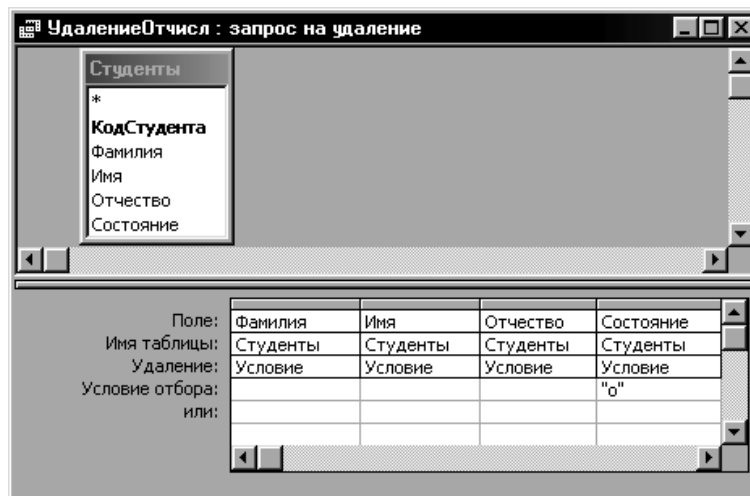



Рисунок 28 – Макет запиту **ВидаленняВідрахованийих**

Тепер при кожному запуску даного запиту з таблиці **Студенти** будуть видалятися всі відраховані студенти. Крім того, будуть також видалятися всі пов'язані з ними записи з таблиці **Оцінки**. Інакше кажучи, із БД будуть видалятися і студент, і його оцінки.

### 1.3 Властивості запитів

Для будь-якого запиту можуть бути задані певні параметри, якими визначаються особливості його виконання. Інакше кажучи, для кожного запиту існує набір властивостей, які використовуються за замовчуванням, і заміна значень цих властивостей тим або іншим способом буде впливати на процес виконання запиту.

Щоб змінити будь-яку властивість, необхідно відкрити запит у режимі конструктора і, не встановлюючи курсор на жодне поле в макеті запиту, скористатися командою *Вид / Властивості* або натиснути кнопку Властивості . Після цього буде відображено вікно *Властивості запиту* (рисунок 29). Далі будуть розглянуті основні властивості запитів.

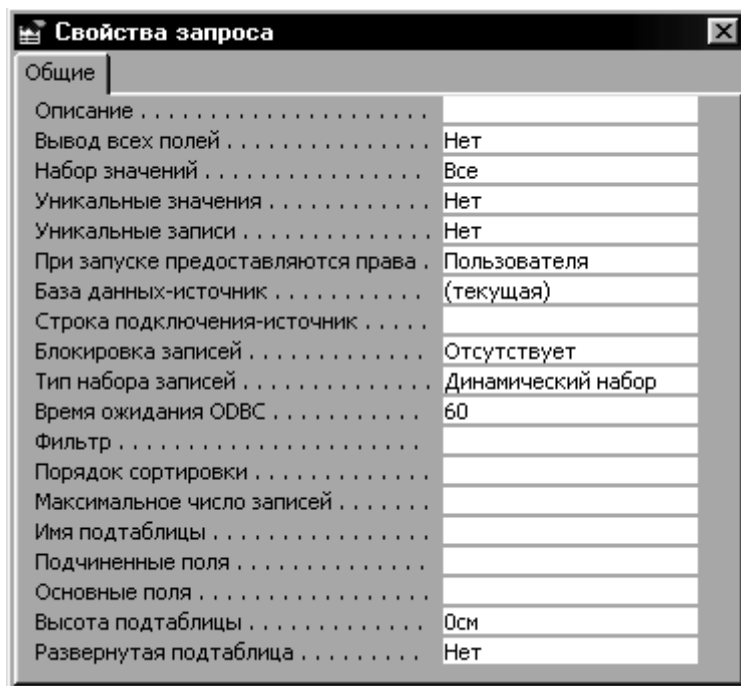


Рисунок 29 – Вікно *Властивості запиту*

*Виведення всіх полів* – відображення в запиті всіх полів з таблиць, які використовуються як джерело даних. Як правило, для даної властивості встановлюється значення *Немає*.

*Набір значень* – виведення на екран перших n записів (5, 25 або 100) або n % загальної кількості записів (5 %, 25 % або 100 %). Цю властивість корисно використовувати в тих випадках,

коли кількість знайдених записів у виконаному запиті буде достатньо великою, тому можна відображувати вже відібрані записи, не очікуючи на закінчення виконання запиту.

*Унікальні значення* – використовується для відображення в запиті тільки унікальних значень серед знайдених записів. Інакше кажучи, якщо для даної властивості встановлено значення *Так*, то в запиті будуть відображатися тільки записи з унікальними поєднаннями значень у відображуваних полях.

*Унікальні записи* – застосовується для відображення унікальних записів тільки в багатотабличних запитах. При створенні багатотабличного запиту може виникнути ситуація, коли у відібраних записах повинно відображатися поле (або кілька полів) однієї з таблиць, у той час як поле пов'язаної з нею таблиці використовується в структурі запиту тільки для зазначення умови для відбору записів. Тобто поля другої таблиці не будуть включені в результат виконання запиту.

*Фільтр* – застосування зазначеного фільтра для знайдених записів, тобто відображення тільки тих записів, які будуть відповідати зазначеній умові.

*Порядок сортування* – визначення напрямку сортування: за зростанням або за зменшенням.

Як приклад використання властивостей *Унікальні значення* і *Унікальні записи* можна розглянути два запити. Перший запит буде включати тільки поле **Оцінка** таблиці **Оцінки**. Якщо в цьому випадку для властивості *Унікальні значення* буде встановлене значення *Так*, то в результаті виконання даного запиту будуть відібрані тільки ті записи, для яких значення в поле **Оцінка** не повторюються (рисунок 30).



Унікальність	
	Оценка
▶	3
	4
	5

Рисунок 30 – Використання властивості *Унікальні значення*

Другий запит буде являти собою доповнення першого. У структуру запиту необхідно додати поле **Назви** таблиці

**Предмети**, при цьому необхідно встановити для нього прапорець *Виведення на екран*, а для поля **Оцінка** даний прапорець потрібно прибрати, щоб після виконання запиту у знайдених записах відображувалися тільки назви предметів. Якщо властивості *Унікальні записи* для даного запиту присвоєно значення *Ні*, то в результаті його виконання будуть знайдені всі записи, у тому числі і з повторюваними назвами (рисунок 31). Якщо ж встановити для властивості *Унікальні записи* значення *Так*, то будуть відображені тільки записи з унікальними значеннями в поле **Предмет** (рисунок 32).

Унікальність	
	Название
▶	Физика
	Химия
	Физика
	Химия
	Философия
	Экономика
	Математика
	Физика
	Химия

Рисунок 31 – *Унікальні записи* не встановлено

Унікальність	
	Название
▶	Физика
	Химия
	Математика
	Экономика
	Философия

Рисунок 32 – *Унікальні записи* встановлено

Необхідно відзначити той факт, що властивість *Унікальні значення* використовується тільки для полів, відображуваних у запиті, при цьому джерелом даних служить одна таблиця. З іншого боку, властивість *Унікальні записи* встановлюється для багатотабличних запитів, у яких поля деяких таблиць присутні в запиті, але не відображуються в знайдених записах. Тому через розходження у сфері дії даних властивостей для того самого запиту неможливо одночасно встановити значення *Так* і для властивості *Унікальні значення*, і для властивості *Унікальні записи*.

## Розділ 2. ФОРМИ

### 2.1 Види форм

Робота з даними в БД безпосередньо здійснюється в таблицях і запитах. При цьому пропонувані можливості зі зміни структур таблиць і даних, що містяться в них, корисні скоріше розробнику, ніж користувачеві. В Access є широкі можливості зі створення інтерфейсу користувача за допомогою *форм*.

Створюючи форму в БД, розробник може переслідувати одну з декількох цілей:

1 *Відображення і зміна даних*. Найчастіше форми створюються саме для виведення і зміни даних у БД. При цьому розробник може задати будь-який вид відображення інформації, тобто довільний набір полів з однієї або декількох таблиць або запитів (якщо поля перебувають у різних таблицях, то між цими таблицями має бути зв'язок). Також можна встановити будь-які обмеження на редагування даних у формі, тобто дозволити редагування одних полів і заборонити зміну інших.

2 *Введення даних*. Форми можуть використовуватися для введення нових даних у таблицю, при цьому введення буде виконуватися користувачем у тому форматі, що заданий розробником. Інакше кажучи, користувач може вводити значення тільки для тих полів, які представлені у формі.

3 *Друк*. Незважаючи на те, що для друку даних у БД використовуються звіти, вміст форми також можна роздрукувати.

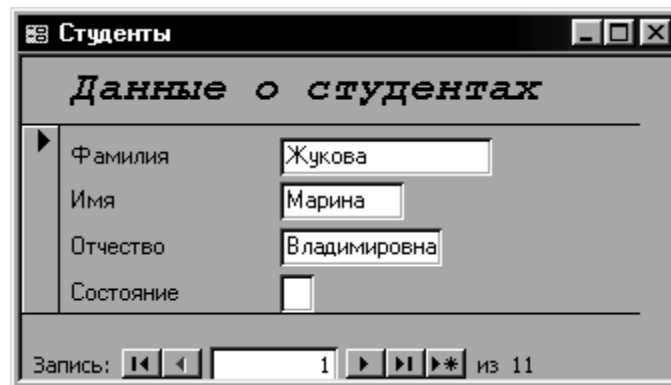
4 *Повідомлення*. Додаткова можливість використання форм полягає у створенні за їх допомогою різних повідомлень, які відображатимуться згодом у тій або іншій ситуації для полегшення роботи користувача. У такому вигляді форми використовуються в додатках.

Таким чином, можна зробити висновок, що форми в Access є головним засобом розроблення користувальницького інтерфейсу. Залежно від того, з якою метою створюється форма, розробник визначає параметри її зовнішнього вигляду. Існує кілька основних видів форм:

*Багатосторінкова (у стовпчик)* – поля запису виводяться в один стовпчик, при цьому в цей самий момент у формі



відображується вміст тільки поточного запису. На рисунку 33 зображена багатосторінкова форма **Студенти**, у якій відображені дані про студентів з таблиці **Студенти**.



The screenshot shows a window titled "Студенты" with a subtitle "Данные о студентах". It displays a form with four fields: "Фамилия" (Жукова), "Имя" (Марина), "Отчество" (Владимировна), and "Состояние" (empty). At the bottom, there is a navigation bar with "Запись:" followed by navigation icons and the text "1 из 11".

Рисунок 33 – Багатосторінкова форма

*Стрічкова* – всі поля запису виводяться в один рядок, при цьому у формі відображуються всі записи. У стрічковій формі **Студенти**, яка подана на рисунку 34, відображені всі записи з таблиці **Студенти**.



The screenshot shows a window titled "Студенты" with a subtitle "Данные о студентах". It displays a striped form with a table of student records. The columns are "Фамилия", "Имя", "Отчество", and "Состояние". The records are as follows:

Фамилия	Имя	Отчество	Состояние
Жукова	Марина	Владимировна	
Морозов	Игорь	Олегович	
Петров	Михаил	Александрович	
Игнатов	Максим	Валерьевич	
Волков	Юрий	Алексеевич	
Волошина	Виктория	Ивановна	
Литвинов	Евгений	Валерьевич	о
Гущина	Елена	Валерьевна	
Кузнецов	Руслан	Алексеевич	о
Семененко	Светлана	Станиславовна	о
*			

At the bottom, there is a navigation bar with "Запись:" followed by navigation icons and the text "1 из 10".

Рисунок 34 – Стрічкова форма

*Таблична* – відображення записів здійснюється в режимі таблиці. На рисунку 35 подана таблична форма **Студенти**, у якій всі записи таблиці **Студенти** виведені в табличному режимі.

Фамилия	Имя	Отчество	Состояние
Жукова	Марина	Владимиров	
Морозов	Игорь	Олегович	
Петров	Михаил	Александров	
Игнатов	Максим	Валерьевич	
Волков	Юрий	Алексеевич	
Волошина	Виктория	Ивановна	
Литвинов	Евгений	Валерьевич	о
Гущина	Елена	Валерьевна	
Кузнецов	Руслан	Алексеевич	о
Семенов	Светлана	Станиславов	о

Запись: 1 из 11

Рисунок 35 – Таблична форма

*Підлегла* – використовується за необхідності виведення даних, зв'язаних відношенням «один-до-багатьох», при цьому в підлеглий формі відображаються записи з таблиці, що відповідає слову «багатьох». У формі **ОцінкиСтудентів** (рисунок 36), крім даних з таблиці **Студенти**, є також підлегла форма, у якій відображаються записи з таблиці **Оцінки**.

**Оценки по предметам**

**Студенты**

Фамилия: Жукова  
 Имя: Марина  
 Отчество: Владимировна  
 Состояние:

**Оценки**

Предмет	Оценка	ДатаСдачи
Физика	5	15.01.99
Химия	5	20.01.99
*	0	

Запись: 1 из 2

Запись: 1 из 11

Рисунок 36 – Підлегла форма

Необхідно звернути увагу на той факт, що у стовпчику **Предмет** цієї форми виводяться назви предметів, у той час як у таблиці **Оцінки**, що служить джерелом даних для підлеглої форми, є тільки коди предметів. Подібний ефект досягається за допомогою майстра підстановок для таблиці **Оцінки**, причому як підстановка для поля **КодПредмета** було використано поле **Назва** з таблиці **Предмети**.

## 2.2 Створення форми

Щоб створити форму, необхідно у вікні БД вибрати у списку *Об'єкти* категорію *Форми* і скористатися кнопкою *Створити* або виконати команду *Вставка | Форма*. Після цього в діалоговому вікні *Нова форма* (рисунок 37) необхідно вибрати режим створення форми, а також таблицю (у списку, що розкривається), що буде використовуватися як джерело даних. Основними режимами створення будь-якої форми є *Конструктор*, *Майстер форм* і *Автоформа*.

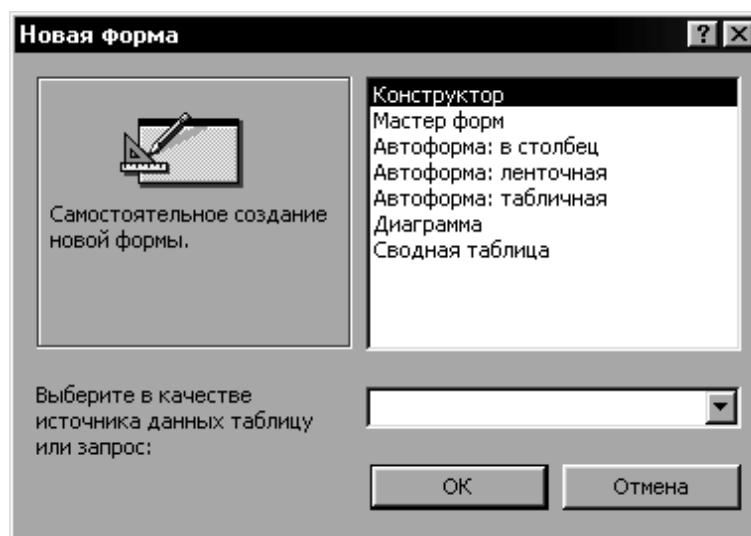


Рисунок 37 – Діалогове вікно *Нова форма*

*Майстер форм*. Багато форм можуть бути створені в режимі конструктора, який буде розглянуто нижче, не вдаючись до допомоги додаткових засобів побудови. Однак більш простим і наочним методом розроблення форм є використання майстра, за допомогою якого можна створити основну частину форми (зокрема задати набір полів для відображення), після чого в

режимі конструктора зробити “косметичні” зміни (наприклад змінити розташування полів на формі).

Якщо в якомусь полі таблиці використовується підстановка, то при додаванні такого поля у форму майстер створення форм автоматично встановлює режим відображення таким чином, щоб замість реального значення в даному полі використовувалась підстановка. При розробленні форми в режимі конструктора подібне використання підстановки буде складно реалізувати, тому що в цьому випадку доведеться змінювати значну кількість властивостей того елемента, у якому повинно відображатися значення підстановки.

Як приклад роботи майстра буде розглянуто створення форми **Студенти**, у якій буде відображатися інформація про студентів. Після вибору режиму *Майстер форм* і таблиці-джерела **Студенти** (у списку, що розкривається) необхідно *натиснути* ОК.

На першому етапі роботи майстра необхідно вибрати поля, які будуть використовуватися в створюваній формі. У цьому випадку необхідно вибрати поля **Прізвище, Ім'я, ПоБатькові та Стан** (рисунок 38), і потім натиснути *Далі*>. Слід зазначити, що поля можуть додаватися у форму з декількох таблиць – для цього необхідно вибирати потрібну таблицю у списку, що розкривається, *Таблиці та запити* вікна майстра форм.

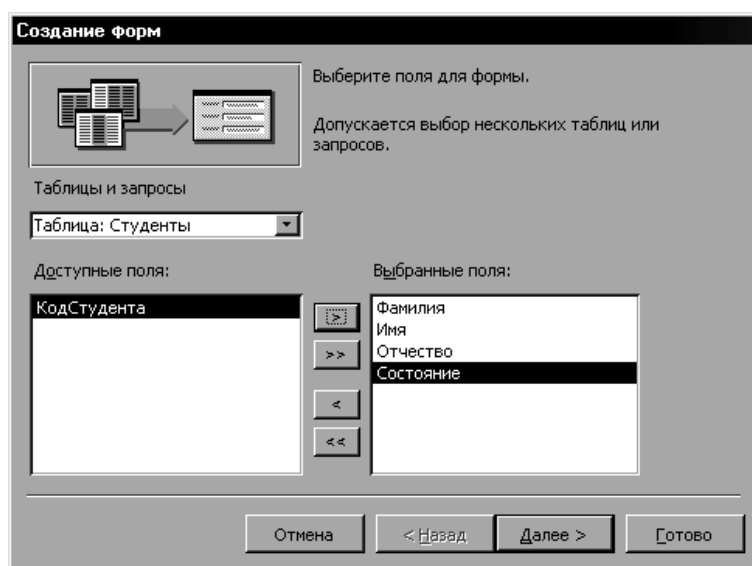


Рисунок 38 – Перший етап роботи майстра форм.  
Вибір полів

Другий етап передбачає вибір зовнішнього вигляду форми. У нашому випадку можна вибрати режим *в один стовпчик* і натиснути кнопку *Далі* >.

На третьому етапі роботи майстра необхідно вибрати стиль дизайну форми, що створюється (рисунок 39). Для форми **Студенти** можна скористатися стилем *Стандартний* і потім натиснути *Далі* >.

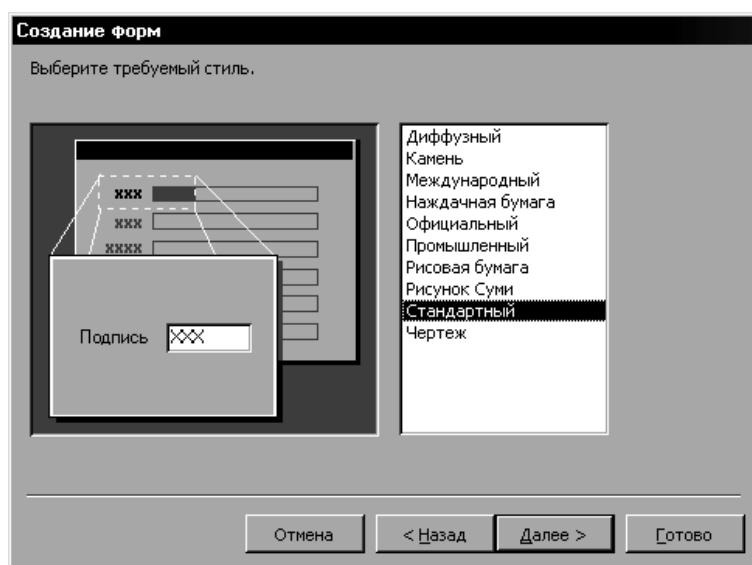


Рисунок 39 – Третій етап роботи майстри форм.  
Вибір стилю

Останній етап припускає задавання імені новій формі. Тут необхідно ввести ім'я **Студенти** і як подальші дії вибрати пункт *Відкрити форму для перегляду та введення даних*. Для завершення роботи майстра потрібно натиснути кнопку *Готово*.

У результаті буде отримано найпростіший варіант багатосторінкової форми **Студенти**, у якій відображується інформація про студентів (рисунок 40).

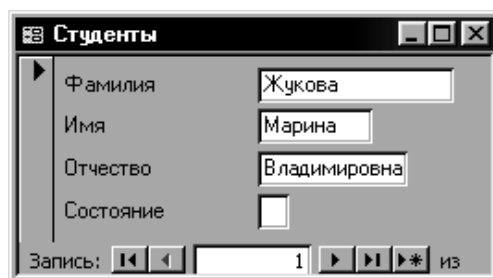







Рисунок 40 – Форма Студенти



Слід зазначити, що переміщення між записами, а також додавання нових записів виконується за допомогою засобу *Запис* (рисунок 40) аналогічно роботі в режимі таблиці (таблиця 4).

Таблиця 4 – Кнопки в області виділення рядка

Кнопка	Опис
	поточний запис, уже збережений
	новий запис, у який можна вводити дані
	запис, що змінюється, ще не збережений
	запис у цей момент заблокований

*Автоформа.* У випадку, коли джерелом даних для форми є одна таблиця або запит, можна скористатися режимом Автоформа. У цьому випадку Access автоматично обирає поля і розміщує їх на формі. Для створення форми в автоматичному режимі можна скористатись описаним раніше діалоговим вікном *Нова форма* (рисунок 37) або ж спеціальною кнопкою на панелі інструментів *Автоформа* . Форма **ОцінкиСтудентів** (рисунок 36) була створена саме так.

*Конструктор.* Як правило, режим конструктора використовується для вдосконалення зовнішнього вигляду форми, створеної за допомогою майстра. Робота з конструктором форм буде розглянута на прикладі створеної форми **Студенти**.

Щоб викликати режим конструктора для відкритої форми, необхідно виконати команду *Вид | Конструктор* або скористатися кнопкою *Вид*, вибравши у списку, що розкривається, *режим*  *Конструктор*. З іншого боку, щоб вибрати режим перегляду форми, перебуваючи в конструкторі, можна виконати команду *Вид | Режим форми* або задати цей режим за допомогою кнопки *Вид* . Структура будь-якої форми (рисунок 41) являє собою набір таких компонентів:

*Заголовок форми* – область у верхній частині форми, у якій міститься інформація, що не залежить від вмісту відображуваних записів. Зокрема в заголовку форми може бути розташований напис із назвою форми (наприклад “Дані про студентів”), а також назви стовпців, тобто полів записів (**Прізвище, Ім'я, ПоБатькові і Стан**). Приклад заголовка з зазначенням назви форми і підписів

до стовпців представлений у стрічковій формі Студенти (рисунок 34).

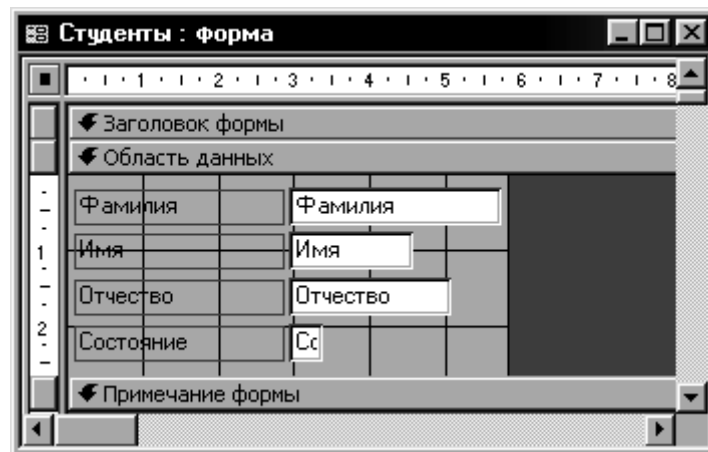



Рисунок 41 – Форма Студенты в режимі конструктора

*Область даних* – центральна частина форми, у якій відображується інформація з таблиць або запитів, а також пояснювальні написи (зокрема назви полів і їхній вміст).

*Примітка форми* – область у нижній частині форми, що містить не залежну від записів інформацію, аналогічно заголовку.

Вся інформація відображується у формі за допомогою так званих *елементів управління*. Щоб додати у форму новий елемент управління, необхідно скористатися *Панеллю елементів* (рисунок 42) яка відображується командою *Вид | Панель елементів* або кнопкою *Панель елементів* .

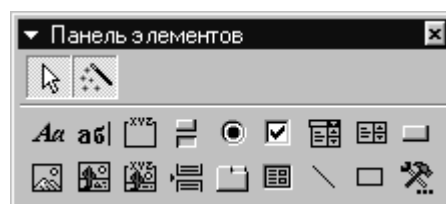




Рисунок 42 – Панель елементів

Перелік всіх елементів управління, а також їхній опис представлені в таблиці 5.

Слід зазначити, що на панелі елементів є також дві кнопки (рисунок 42), які визначають режим роботи з формою:

 – *вибір об'єктів*. Якщо кнопка перебуває в натиснутому стані, то це означає, що жоден елемент управління на панелі не обраний, і можна виконувати зміни об'єктів, які вже є на формі. З іншого боку, якщо вибрати на панелі будь-який елемент управління, то кнопка набуде вигляду «відтиснутої». Це буде означати, що обрано об'єкт (тобто елемент управління) для додавання на форму. Щоб скасувати вибір елемента, необхідно знову натиснути кнопку *Вибір об'єктів*;



 – *майстри*. Натиснутий стан кнопки означає, що в цей момент встановлено режим автоматичного використання майстрів, які будуть запускатися у випадку створення нового об'єкта на формі. Рекомендується в процесі розроблення макета форми тримати цю кнопку в натиснутому стані.

Таблиця 5 – Список елементів управління



Кнопка	Назва	Опис
1	2	3
	Напис	Містить текст для опису елемента управління або іншого об'єкта (заголовки форми, підписи до полів)
	Поле	Відображення будь-яких значень (як правило, полів з таблиць і запитів)
	Група перемикачів	Використовується для об'єднання в групу декількох елементів вибору (вимикачів, перемикачів або прапорців)
	Вимикач	Застосовується для вибору користувачем значення типу Так/Ні або Вкл/Викл
	Перемикач	Використовується в тому випадку, коли з декількох значень або параметрів необхідно вибрати тільки один
	Прапорець	Застосовується для вибору декількох значень або параметрів з наявного набору
	Поле зі списком	Використовується для роботи з усіма значеннями певного поля таблиці або запиту. При цьому в полі відображується поточне обране значення, у той час як за допомогою списку, що розкривається, можна вибрати інше значення поля з наявного набору записів



Продовження таблиці 5

1	2	3
	Список	Застосовується для відображення всіх значень певного поля таблиці або запити, при цьому в списку виділено поточне значення даного поля. Вибір іншого значення виконується за допомогою виділення відповідного елемента списку
	Кнопка	Використовується для виконання будь-якої дії, наприклад відображення іншої форми або звіту
	Малюнок	Дозволяє розмістити на формі малюнок (у якості ілюстрації або фону)
	Вільна рамка об'єкта	Застосовується для додавання у форму зовнішнього об'єкта ActiveX (діаграма, малюнок або звуковий файл), які не знаходяться безпосередньо в таблиці БД
	Приєднана рамка об'єкта	Використовується для додавання у форму об'єкта ActiveX, що перебуває в таблиці БД. Для об'єктів, які не можуть зберігатися в таблиці, у формі відображується іконка (значок) додатка, за допомогою якого обробляється даний об'єкт
	Розрив сторінки	Застосовується з метою вставлення розривів у багатосторінковій формі
	Набір вкладок	Служить для створення у формі декількох вкладок (або сторінок), на яких можуть розташовуватися інші елементи
	Підлегла форма/звіт	Дозволяє додати у форму іншу форму або звіт. При цьому додається форма, яка називається «вкладеною» або підлеглою
	Лінія	Призначена для відображення у формі розділових ліній між іншими елементами, розділяючи їх у такий спосіб на логічні категорії

## Продовження таблиці 5

1	2	3
	Прямокутник	Служить для додавання у форму прямокутників з метою поліпшення сприйняття інформації, яка відображується на формі
	Інші елементи	Використовується для відображення повного списку елементів ActiveX, які можуть бути додані на форму

Щоб додати будь-який елемент управління на форму (наприклад напис), необхідно виконати таку послідовність дій:

- 1 Вибрати кнопку *Напис* на панелі елементів.
- 2 Помістити покажчик миші на форму. При цьому замість стрілки як покажчик з'явиться хрестик з буквою *A*, що зображує напис.
- 3 Натиснути ліву кнопку й перетягнути покажчик миші таким чином, щоб зображення напису досягло необхідних розмірів, після чого відпустити кнопку миші.
- 4 У полі, що з'явилося, ввести напис, після чого натиснути лівою кнопкою миші за її межами.

У результаті на формі буде відображено зазначений напис. На рисунку 43 подано виділений напис *Прізвище*.

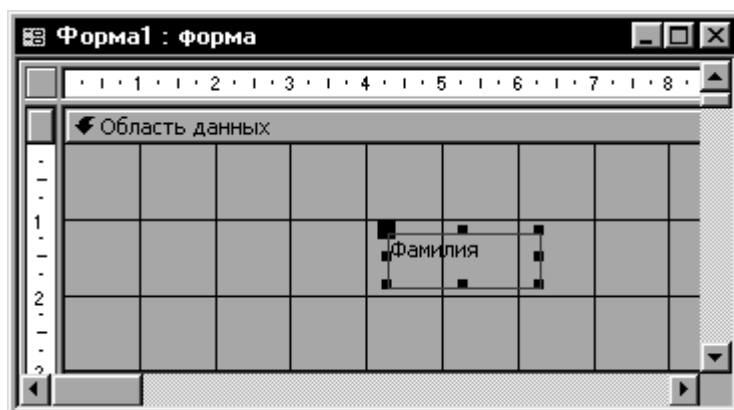



Рисунок 43 – Виділений напис *Прізвище*

Щоб змінити розміри об'єкта, наприклад напису, необхідно цей об'єкт виділити, потім встановити покажчик миші на його границю, обумовлену чорними квадратами, щоб з'явилася

подвійна стрілка. Після цього необхідно перетягнути границю, збільшивши або зменшивши об'єкт до необхідних розмірів.

За необхідності переміщення об'єкта треба його виділити, потім встановити на великому чорному квадраті в лівому верхньому куті покажчик миші таким чином, щоб з'явилося зображення розкритої долоні, після чого можна перетягнути об'єкт на потрібне місце.

Якщо об'єкт потрібно видалити, то його необхідно спочатку виділити, і після цього можна скористатися клавішею **Delete**.

Кожен елемент управління являє собою об'єкт з набором властивостей. Щоб змінити ті або інші властивості виділеного об'єкта, наприклад напису *Прізвище*, необхідно скористатися командою *Вид | Властивості* або кнопкою *Властивості* , у результаті чого з'явиться вікно властивостей даного напису (рисунок 44).

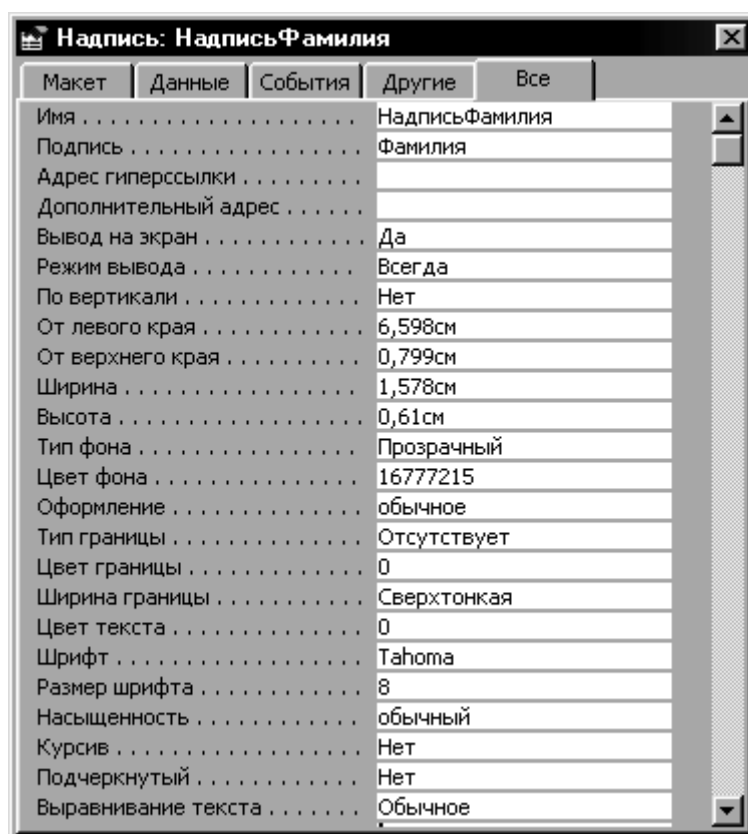


Рисунок 44 – Властивості напису *Прізвище*

Слід зазначити, що кожний об'єкт на формі має ім'я, яке використовується при створенні додатків на VBA. У

розглянутому випадку для напису було обрано ім'я НадписьФамилия.

Властивість *Підпис* містить значення, що відображується у вигляді напису на формі (у цьому випадку зазначене значення – *Прізвище*).

Для об'єкта може також бути визначено властивість виведення на екран у формі (властивість *Виведення на екран*), а також режим виведення (властивість *Режим виведення*), у якому можна визначити, коли буде відображуватися даний напис: *Завжди*, *Тільки при друці* або *Тільки на екран*.

Для оформлення зовнішнього вигляду об'єкта є ряд властивостей, які впливають на ті або інші особливості його відображення на формі: *Ширина*, *Висота*, параметри фону, параметри границі, а також властивості шрифту.

У вікні властивостей можуть відображуватися відразу всі властивості обраного об'єкта або тільки з певної категорії. Для вибору того або іншого режиму відображення властивостей використовуються вкладки *Макет*, *Дані*, *Події*, *Інші* й *Всі* (рисунок 37).

Якщо на форму додається поле **аб|**, то для нього автоматично “приєднується” пояснювальний напис. Наприклад, необхідно відобразити у формі вміст поля **Прізвище** таблиці **Студенти**. Для цього необхідно спочатку додати на форму нове поле (рисунок 45).

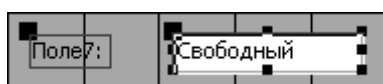


Рисунок 45 – Нове поле у формі

Потім необхідно змінити властивості поля і приєданого напису. Для властивості *Ім'я* можна встановити значення *НаписПрізвище*, а для властивості *Підпис* – значення *Прізвище*. Після цього необхідно закрити вікно властивостей, у результаті чого приєднаний напис буде виглядати так, як показано на рисунку 46.



Рисунок 46 – Приєднаний до поля напис

Останнім етапом у створенні поля буде зміна його властивостей. Для властивості *Ім'я* можна вказати значення *ПолеПрізвище*. Досить важливою властивістю для об'єкта Поле є властивість *Дані* (рисунок 47), у якому вибирається джерело даних для цього поля.

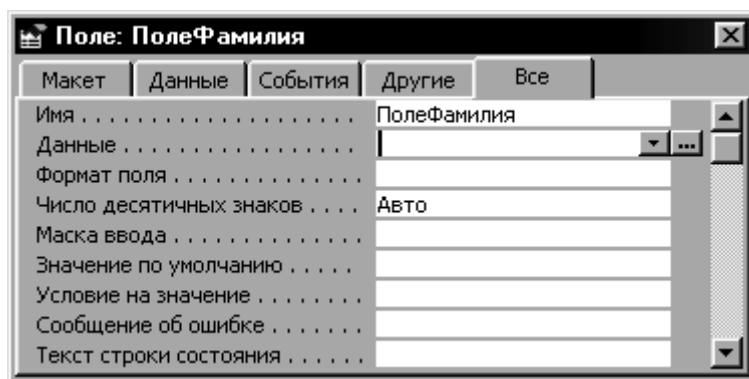




Рисунок 47 – Властивості поля ПолеПрізвище

За допомогою кнопки  можна відобразити вікно *Будівника виразів* у якому необхідно вказати вираз для виділеного поля.

За допомогою кнопки  у списку, що розкривається, можна вибрати поле таблиці або запиту, що буде служити джерелом даних для об'єкта **Поле**. Слід зазначити той факт, що в цьому списку відображуються тільки ті таблиці і запити, які встановлені у властивості *Джерело записів* для форми, тому в нашому випадку список, що розкривається, буде порожній.

Оскільки в якості джерела даних для поля *ПолеПрізвище* необхідно вибрати таблицю **Студенти**, то необхідно додати цю таблицю у властивість *Джерело записів* форми. Щоб вибрати той або інший об'єкт для зміни його властивостей, наприклад поле, напис або форму, можна або натиснути на необхідному об'єкті, або скористатися кнопкою *Об'єкт* панелі інструментів *Формат (форма/звіт)* (рисунок 48), після чого вибрати зі списку потрібний об'єкт (у нашому випадку – об'єкт *Форма*).

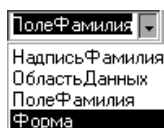




Рисунок 48 – Кнопка *Об'єкт*

Потім для властивості *Джерело записів* (вкладка *Дані* або *Усі*) необхідно вибрати таблицю **Студенти**. Для цього треба в рядку даної властивості скористатися кнопкою  й вибрати зі списку всіх таблиць і запитів БД **Освіта** таблицю **Студенти**. З іншого боку, якби треба було вибрати кілька таблиць для властивості *Джерело записів*, то необхідно було б скористатися кнопкою , після чого створити запит відповідної структури, додавши в нього потрібні поля необхідних таблиць.


Після установаження властивості *Джерело записів* форми треба для властивості *Дані* об'єкта *ПолеПрізвище* вибрати у списку, що розкривається, поле **Прізвище**. У результаті об'єкти форми, тобто напис і поле, будуть мати такий вигляд, як на рисунку 49.



Рисунок 49 – Об'єкти форми

Потім так само можна додати на форму об'єкти *ПолеІм'я*, *ПолеПоБатькові* і *ПолеСтан*, вибравши для кожного з них як джерело даних (властивість *Дані*) поля **Ім'я**, **ПоБатькові** і **Стан** відповідно.

На наступному етапі створення даної форми можна в заголовку форми відобразити напис *Дані про студентів*. Щоб одержати доступ до областей форми *Заголовок* і *Примітка*, необхідно виконати команду *Вид | Заголовок/примітка форми*. Після цього можна в область заголовка додати напис, указавши для її властивості *Підпис* значення *Дані про студентів*, а також встановивши параметри шрифту і границі.


Отриману форму необхідно зберегти під ім'ям **Студенти**. Для її перегляду необхідно скористатися кнопкою *Вид* , після чого на екрані з'явиться створена форма **Студенти**, що відображує значення записів у таблиці **Студенти** (рисунок 40).

## 2.3 Підлеглі форми

Форма з попереднього прикладу мала досить просту структуру, тобто джерелом даних для неї служила одна таблиця **Студенти**. У тому випадку, коли необхідно відображувати і змінювати дані відразу в декількох таблицях, необхідно використати підлеглі форми.

Наприклад, необхідно створити форму, у якій для кожного студента (таблиця **Студенти**) виводилися б назви предметів (таблиця **Предмети**), отримані оцінки й дата складання (таблиця **Оцінки**). Для реалізації описаного завдання можна створити форму такої структури: вмістом головної форми буде інформація про студентів, а в підлеглій формі будуть відображатися інші дані, які перебувають у таблиці **Оцінки**.

Першим кроком у створенні описаної складної форми буде створення нової форми **ОцінкиСтудентів** або використання вже наявної форми **Студенти**, яку можна буде зберегти під новим ім'ям – **ОцінкиСтудентів**. Якщо форма створюється за допомогою майстра, то як джерело даних необхідно вибрати таблицю **Студенти**, з якого потрібно вибрати поля **Прізвище**, **Ім'я**, **ПоБатькові** і **Стан** (рисунок 38).

На наступному етапі необхідно додати на створену форму об'єкт *Підлегла форма* за допомогою кнопки *Підлегла форма/звіт*, у результаті чого почне виконуватися майстер підлеглих форм, якщо попередньо кнопка  була встановлена в натиснутий стан.

На початку роботи майстра буде відображено вікно, у якому необхідно вибрати джерело даних для підлеглої форми: *таблиці і запити* або *форми*, які є в БД. Вибираємо пункт *Наявні таблиці і запити* і тиснемо *Далі*>.

Другий етап припускає вибір полів з однієї або декількох таблиць, які необхідно включити в підлеглу форму. У розглянутому прикладі потрібно з таблиці **Оцінки** вибрати поля **КодПредмета**, **Оцінка** і **ДатаЗдачі**. Слід зазначити, що замість назв предметів у форму були додані тільки їхні коди (поле **КодПредмета**). Однак у режимі форми замість кодів будуть відображатися назви предметів, тому що для поля

**КодПредмета** таблиці **Оцінки** була використана підстановка поля **Назва** з таблиці **Предмети**.

На наступному кроці роботи майстра необхідно визначити поле зв'язку між головною і підлеглою формою. У цьому випадку майстер автоматично визначив існуючий зв'язок за полем **КодСтудента** між таблицями **Студенти** і **Оцінки**. Таким чином, необхідно в режимі *Вибір зі списку* вибрати зв'язок 'Оцінки' для кожного запису 'Студенти', поле зв'язку 'КодСтудента' (рисунок 50).

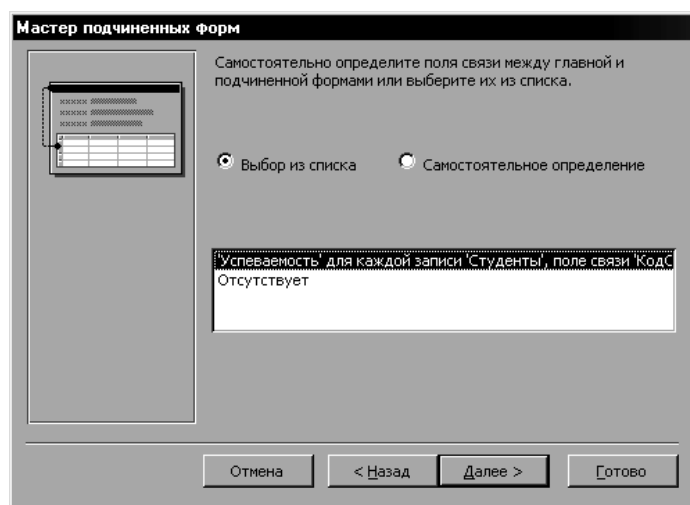


Рисунок 50 – Майстер підлеглих форм. Визначення зв'язків

На останньому етапі необхідно вказати ім'я для підлеглої форми, наприклад *підлегла форма Оцінки* і натиснути *Готово*.

У результаті отримана основа складної форми **ОцінкиСтудентів** з використанням підлеглої форми. Для завершення створення форми можна в область заголовка додати напис *Оцінки за предмети*, а також зробити підписи для даних: *Студенти* і *Оцінки*. Крім того, можна додати розділову лінію (об'єкт *Лінія*) між даними головної і підлеглої форм.

Створена в такий спосіб складна форма **ОцінкиСтудентів** наведена на рисунку 36.

Слід зазначити, що для відображення назв предметів було автоматично створене поле зі списком (рисунок 51), при цьому у його властивості *Дані* отримується значення **КодПредмета**.



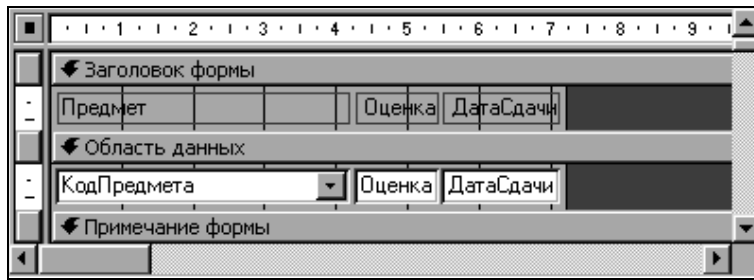


Рисунок 51 – Підлегла форма Оцінки

З іншого боку, у властивості *Джерело рядків* даного об'єкта перебуває запит, що містить два поля з таблиці **Предмети**: **КодПредмета** і **Назва**. Відображення у формі тільки назв предметів було досягнуто автоматичним установленням таких властивостей поля на вкладці *Макет* (рисунок 52):

*Кількість стовпців – 2;*

*Ширина стовпців – 0 см; 2,54 см.*




Рисунок 52 – Властивості поля зі списком *КодПредмета*.

Вкладка *Макет*

## 2.4 Використання кнопок

В Access існує можливість виконання тих або інших команд обробки записів, форм, звітів і т. д. за допомогою додавання на форму спеціального об'єкта – *кнопки*. Натискання на цю кнопку в режимі форми приводить до виконання певної дії.

Наприклад, у формі **ОцінкиСтудентів** можна додавати оцінки студентам, але не можна додавати нові предмети, тому що для поля зі списком встановлена властивість *Обмежитися списком* (вкладка *Дані*). Для додавання нових предметів у таблицю **Предмети** можна створити на формі кнопку, при натисканні на яку будуть відображатися записи таблиці **Предмети**. Таким чином, можна буде вносити додавання у список предметів.

При додаванні на форму об'єкта *Кнопка* буде запущений майстер створення кнопок, якщо кнопка  перебуває в натиснутому стані. На першому етапі його роботи необхідно вибрати категорію *Робота з формою*, у списку дій якої вибрати *Відкрити форму* (рисунок 53), щоб при натисканні на кнопку виконувалося відкриття певної форми.

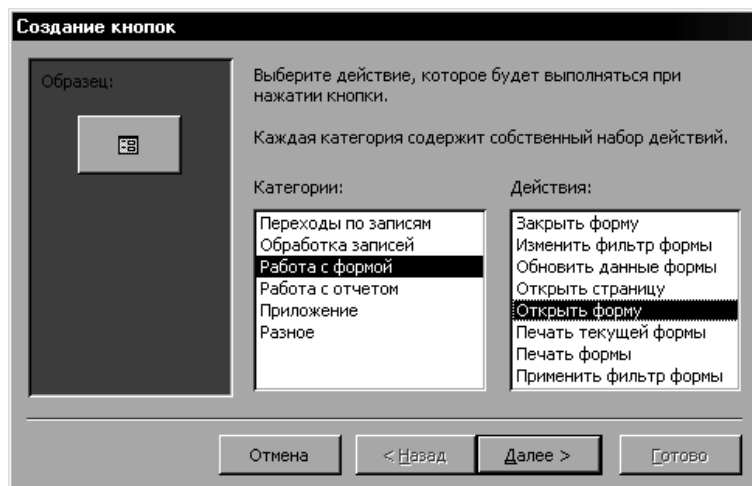


Рисунок 53 – Створення кнопок. Вибір дії

Слід зазначити, що для кнопки можна вибрати дію, розташовану в одній з таких категорій:

*Переходи по записам* – містить дії, пов'язані з переміщенням по записах таблиці (*Перший запис*, *Останній запис* і т. д.).

*Обробка записів* – пропонує набір дій, пов'язаних з обробкою записів (*Додати запис*, *Видалити запис* і т. д.).

*Робота з формою* – містить список дій, використовуваних при роботі з формою (*Відкрити форму*, *Закрити форму*, *Друк* і т. д.).

*Робота зі звітом* – пропонує перелік дій для обробки звітів (*Друк звіту, Перегляд звіту* і т. д.).

*Додаток* – містить набір дій для керування додатком (*Виконати додаток, Вийти з додатка* і т. д.).

*Різні* – містить корисні дії, які не стосуються жодної з описаних вище категорій (*Виконати запит, Друк таблиці* і т. д.).

Другий етап припускає вибір таблиці, що повинна бути відкрита. У нашому випадку необхідно вибрати таблицю **Предмети**.

На наступному етапі можна встановити режим, при якому у формі будуть відображатися тільки ті записи, які задовольняють певним умовам. Тому, якщо немає необхідності в установці обмежень на список предметів, то потрібно вибрати режим *Відкрити форму і показати всі записи*.

Черговий етап припускає вибір тексту або малюнка, який необхідно розмістити на кнопці. У даному прикладі можна в поле *Текст* вказати підпис **Предмети** (рис. 54).

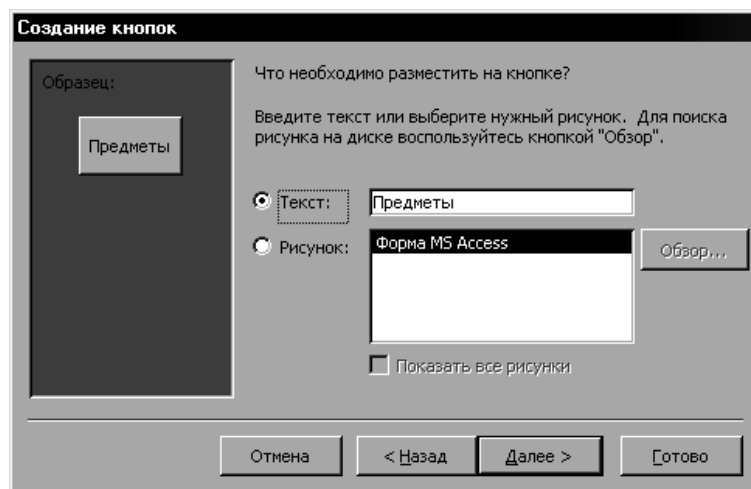


Рисунок 54 – Створення кнопок. Встановлення підпису

На останньому етапі необхідно вибрати ім'я для об'єкта *Кнопка*, наприклад **Предмети**.

У результаті усіх виконаних дій на формі **ОцінкиСтудентів** буде створена кнопка **Предмети**. Натискання на цю кнопку в режимі форми приведе до відображення форми **Предмети**, у якій буде виведено список наявних предметів з можливістю додавання нових назв.

Можна також для кнопки *Предмети* вказати текст спливаючої підказки. Для цього необхідно в режимі конструктора форми **ОцінкиСтудентів** виділити кнопку і встановити для неї у властивості *Спливаюча підказка* значення *Список предметів*.

Отримана в результаті усіх виконаних дій форма **ОцінкиСтудентів** наведена на рисунку 55.

Предмет	Оценка	ДатаСдачи
Физика	5	15.01.99
Химия	5	20.01.99
*	0	

Рисунок 55 – Форма *ОцінкиСтудентів*

## 2.5 Умовне форматування

Часто при відображенні даних у формі буває зручно тим або іншим способом виділяти значення в полях залежно від виконання певних умов. Подібні дії називають *умовним форматуванням*.

Наприклад, необхідно залежно від оцінки, що одержав студент, виводити її значення шрифтом встановленого формату, а саме:

якщо “5”, то жирним, чорного кольору;

якщо “4”, то жирним, сірого кольору;

якщо “3” або “2”, то жирним, з підкресленням, чорного кольору.

А також якщо за предмет отримана оцінка “3” або “2”, то назва предмета має відображуватись жирним підкресленим шрифтом.

Спочатку необхідно встановити умовне форматування полю *Оцінка*. Для цього необхідно його виділити, перебуваючи в режимі конструктора, після чого необхідно виконати команду *Формат | Умовне форматування*. Потім у діалоговому вікні в рядку *Умова 1* потрібно встановити параметрам такі значення: *Значення поля, дорівнює, 5*. Після цього необхідно вибрати необхідні параметри для шрифту, яким буде відображена оцінка (жирний, чорного кольору).

Щоб вказати наступну умову і відповідний формат, необхідно натиснути у вікні *Умовне форматування* кнопку *Додати>>*. З іншого боку, щоб видалити будь-яку зі встановлених умов, потрібно скористатися кнопкою *Видалити* і вибрати непотрібні умови. Слід зазначити, що для одного поля можна встановити до трьох умов.

Після того як у вікні *Умовне форматування* будуть зазначені всі умови і формати (рисунк 56), необхідно натиснути кнопку *ОК*.

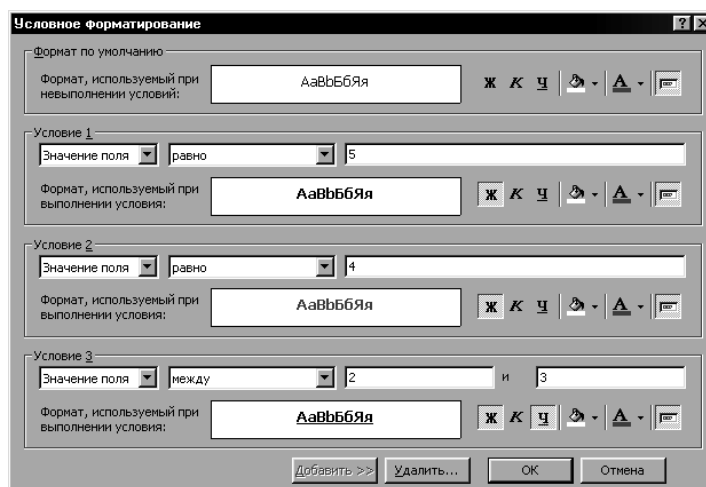


Рисунок 56 – Діалогове вікно *Умовне форматування*

При установленні формату для поля зі списком предметів необхідно вибрати параметр *Виразі* укапати умову

$$[Оцінка]=3 \text{ Or } [Оцінка]=2,$$

після чого потрібно встановити необхідний формат шрифту (рисунк 57).

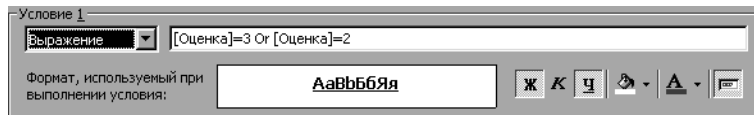


Рисунок 57 – Умовне форматування для поля зі списком предметів

Результати використання умовного форматування можна побачити в режимі форми. Зокрема на рисунку 58 подано оцінки і назви предметів для студента Петрова, при цьому дані відображаються з урахуванням умовного форматування.

Предмет	Оценка	ДатаСдачи
Философия	3	10.02.99
Экономика	4	25.01.00
Математика	5	22.01.99
*	0	

Запись: 1 из 3

Рисунок 58 – Результати умовного форматування

## Розділ 3. ЗВІТИ

### 3.1 Базові відомості про звіти

Щоб представити у друкованому вигляді інформацію із БД (таблиць або запитів), в Access використовуються *звіти*, які мають порівняно з іншими способами друку (з таблиць або форм) такі переваги:

1 Крім відображення вмісту полів з таблиць або запитів, у звітах також можна обчислювати різні підсумкові значення, а також групувати дані.

2 За допомогою звітів можна барвисто оформляти різноманітні документи, необхідні в тих або інших сферах професійної діяльності.

Для відображення основних даних і обчислених підсумкових значеннях у звітах можна використовувати такі можливості:

1) здійснення обчислень будь-якої складності з інформацією, що перебуває не тільки в одній, але також і в різних групах даних;

2) використання ієрархічної структури у відображенні груп даних (у структурі може перебувати до 10 рівнів);

3) друк верхнього і нижнього колонтитулів на кожній сторінці звіту;

4) для звіту можна встановити текст заголовку і приміток;

5) у кожній групі даних можуть перебувати власні заголовки і примітки.

Крім перерахованих можливостей, у звітах (аналогічно створенню форм) також застосовуються впроваджені рисунки, об'єкти і підлеглі звіти.

### 3.2 Створення звіту

Як приклад можна розглянути створення звіту **Оцінки** (рисунок 59), у якому для кожного студента відображуються його оцінки з усіх предметів, а також його середній бал.

Оценки					
28 ноября 2000 г.					
Фамилия	Имя	Отчество	Предмет	Оценка	Дата сдачи
Жукова	Марина	Владимировна	Физика	5	15.01.99
			Химия	5	20.01.99
			Средний балл		
Игнатов	Максим	Валерьевич	Физика	5	15.01.99
			Химия	4	20.01.99
			Средний балл		
Морозов	Игорь	Олегович	Физика	4	15.01.99
			Химия	4	20.01.99
			Средний балл		
Петров	Михаил	Александрович	Математика	5	22.01.99
			Философия	3	10.02.99
			Экономика	4	25.01.00
			Средний балл		

Рисунок 59 – Звіт Оцінки

Якщо в якості вихідних даних для звіту використовуються таблиці або запити, то необхідно створити запит на вибірку **Оцінки**, у якому повинні відображатися поля **Прізвище**, **Ім'я** та **ПоБатькові** таблиці **Студенти**, поле **Назва** таблиці **Предмети**, а також поля **Оцінка** і **ДатаЗдачі** таблиці **Оцінки**. У результаті запуску цього запиту буде отриманий набір записів, зображений на рисунку 60.

Фамилия	Имя	Отчество	Название	Оценка	ДатаСдачи
Жукова	Марина	Владимировна	Физика	5	15.01.99
Жукова	Марина	Владимировна	Химия	5	20.01.99
Морозов	Игорь	Олегович	Физика	4	15.01.99
Морозов	Игорь	Олегович	Химия	4	20.01.99
Петров	Михаил	Александрович	Философия	3	10.02.99
Петров	Михаил	Александрович	Экономика	4	25.01.00
Петров	Михаил	Александрович	Математика	5	22.01.99
Игнатов	Максим	Валерьевич	Физика	5	15.01.99
Игнатов	Максим	Валерьевич	Химия	4	20.01.99

Рисунок 60 – Запит Оцінки

Аналогічно створенню форм створення звітів можна виконувати як за допомогою майстра, так і за допомогою конструктора. Як правило, майстер використовується для створення основної частини звіту, тобто структури, у той час як конструктор застосовується для вдосконалення зовнішнього вигляду звіту.



Щоб створити звіт, необхідно у вікні БД у розділі Звіти скористатися кнопкою *Створити* і вибрати необхідний режим створення нового звіту.

*Майстер звітів*. На першому етапі роботи майстра необхідно вибрати у списку, що випадає, *Таблиці і запити* запит **Оцінки** і додати з нього усі поля у список *Вибрані поля* (рисунок 61). Після цього необхідно натиснути *Далі*>.

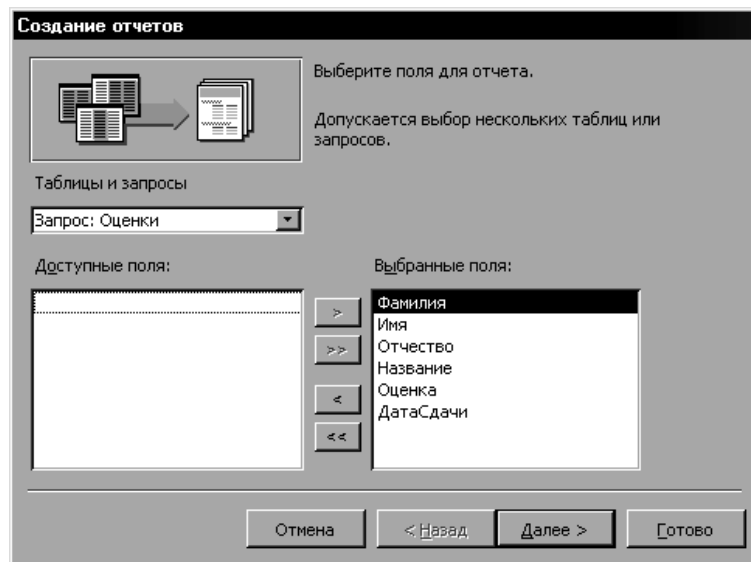


Рисунок 61 – Створення звіту. Вибір полів

Потім необхідно вибрати вид подання даних, тобто на які групи будуть розбиті дані у звіті. У розглянутому прикладі необхідно для кожного студента відображувати список оцінок, тобто ієрархія в цьому випадку буде наступною: **Студенти** → **Предмети**, **Оцінка** і **Дата**, тобто необхідно у списку вибрати пункт *Студенти* (рисунок 62). Після цього потрібно натиснути *Далі*>.

На наступному етапі, якщо буде потреба, можна додати рівні групування відображуваних даних. Наприклад, які оцінки одержав студент у той або інший день, тобто додати рівень **ДатаЗдачі**. Після цього стане доступною кнопка *Групування*, що дозволяє встановити інтервали групування для полів, включених в ієрархічну структуру з метою створенню нових рівнів групування. Наприклад, якщо додати рівень групування **ДатаЗдачі**, то натискання кнопки *Групування* дозволить вибрати в діалоговому вікні *Інтервали групування* за допомогою списку,

що розкривається, один з наявних інтервалів, наприклад за місяцями (рисунок 63).

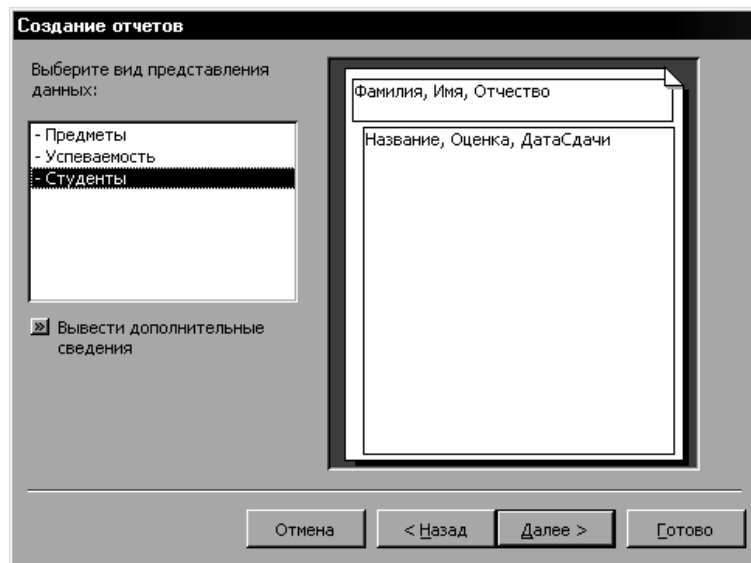


Рисунок 62 – Створення звіту. Вид подання даних

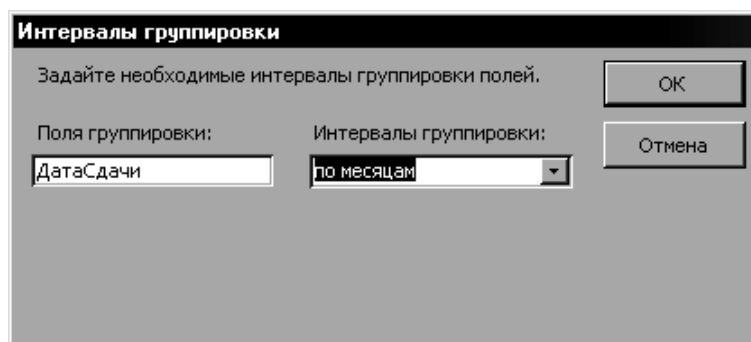


Рисунок 63 – Створення звіту. Інтервали групування

У розглянутому прикладі можна не додавати рівні групування, а відразу натиснути *Далі*>.

Черговий етап припускає визначення порядку сортування записів, що відображуються у звіті (рисунок 64), а також підсумкові обчислення, виконання яких можна встановити за допомогою кнопки *Підсумки*. У діалоговому вікні *Підсумки*, який викликається натисканням відповідної кнопки (рисунок 65) можна обчислити суму (*Sum*), середнє значення (*Avg*), а також мінімум (*Min*) або максимум (*Max*) для даних, що містяться в зазначеному полі.

Для поля *Назва* можна встановити сортування за алфавітом, тобто за зростанням, а у вікні *Підсумки* необхідно вибрати обчислення середнього бала (функція *Avg*).

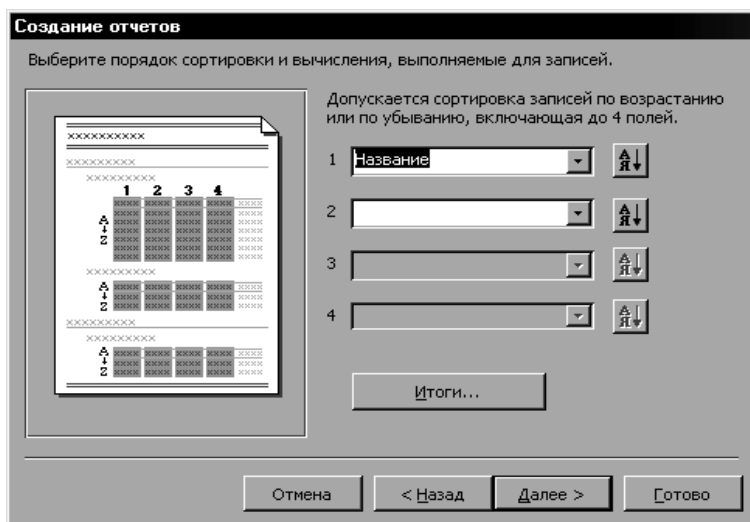


Рисунок 64 – Створення звіту. Порядок сортування

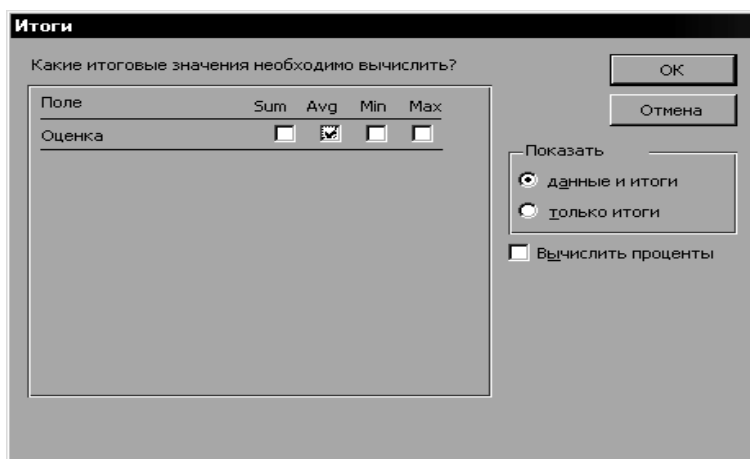


Рисунок 65 – Створення звіту. Підсумки

На наступному етапі можна вибрати вид макета, що буде використовуватися у звіті, а також орієнтацію сторінки (*книжкова* або *альбомна*). У списку *Макет* можна вибрати *східчастий* і встановити книжкову орієнтацію (рисунок 66).

Черговий крок роботи майстра дозволяє вибрати стиль для оформлення звіту, аналогічно тому, як визначається зовнішній вигляд при створенні форм. У нашому випадку можна вибрати стиль *Діловий* і натиснути *Далі >*.

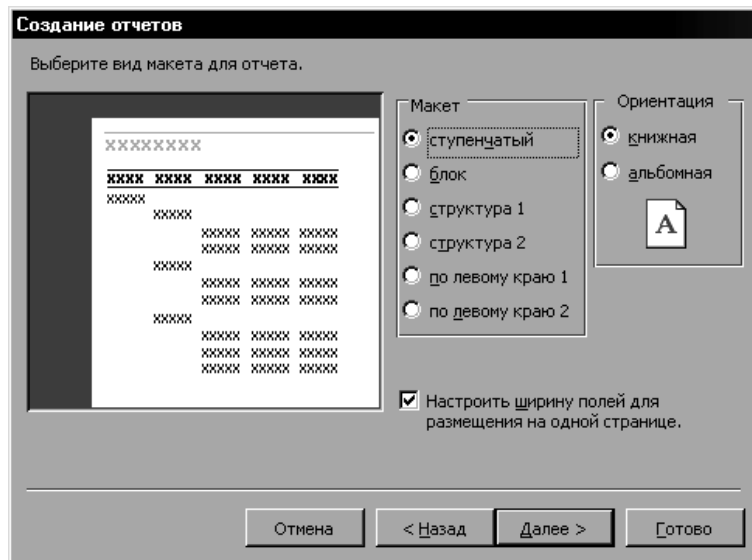



Рисунок 66 – Створення звіту. Вид макета

На останньому етапі необхідно вказати ім'я створеного звіту: **Оцінки**, а також вибрати подальші дії: *Переглянути звіт* (режим попереднього перегляду) або *Змінити макет звіту* (режим конструктора). Слід зазначити, що перемикання між режимом *Попередній перегляд* і *Конструктор* виконується за допомогою кнопки *Вид* , використання якої аналогічно раніше розглянутим об'єктам БД, тобто таблицям, запитам і формам.

Після того як за допомогою майстра створена структура звіту для внесення змін, пов'язаних з його оформленням, необхідно встановити для звіту **Оцінки** режим конструктора.

*Конструктор.* У режимі конструктора будь-який звіт (рисунок 67) являє собою набір таких розділів: верхній і нижній колонтитули (друкуються відповідно вгорі і внизу кожної сторінки звіту) і область даних. При цьому у звіт можна включити заголовок, що буде друкуватися тільки вгорі першої сторінки, а також примітка, що буде відображуватися наприкінці останньої сторінки звіту. Слід також зазначити, що область даних може перебувати всередині однієї або декількох ієрархічних груп, кожна з яких має свій заголовок і примітку. Наприклад, у звіті **Оцінки**, створеному за допомогою майстра (рисунок 67), є тільки одна група – *Прізвище*, для якого встановлено заголовок і примітку.

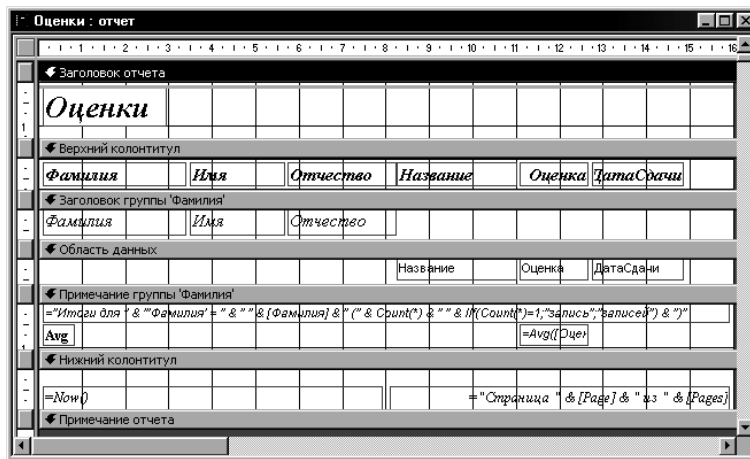


Рисунок 67 – Звіт Оцінки. Режим конструктора

Як ми вже зазначали раніше, за допомогою майстра зручно створювати звіти, а за допомогою конструктора звітів – виправляти помилки. Наприклад, можна поставити завдання внесення за допомогою конструктора ряду змін в отриманий завдяки майстрові звіт:

- 1 Вирівняти заголовок звіту по центру сторінки.
- 2 Замість підпису *Назва* встановити підпис *Предмет*.
- 3 Змінити Підпис стовпчика з *ДатаЗдачі* на *Дата здачі*.
- 4 Видалити в розділі приміток групи *Прізвище* рядок коментарю “*Підсумки для ...*”.
- 5 Замість напису *Avg*, що позначає обчислення середнього значення для оцінок студента, ввести напис *Середній бал* і розташувати його у стовпці *Предмет*.
- 6 Перемістити напис із поточною датою в лівий верхній кут сторінки і видалити напис у правому нижньому куті: “*Сторінка ...*”.
- 7 Додати поруч із датою номер сторінки.

Щоб перемістити будь-який об'єкт, наприклад напис *Оцінки*, у межах одного розділу звіту, достатньо встановити на нього покажчик миші і, натиснувши ліву кнопку, перетягнути об'єкт у необхідне місце. Якщо ж необхідно перемістити об'єкт з одного розділу в інший, наприклад із примітки в заголовок, необхідно виконати таку послідовність дій:

- 1 Виділити об'єкт.
- 2 Вирізати його в буфер командою *Виправлення | Вирізати*.
- 3 Натиснути в тому розділі, куди необхідно перемістити об'єкт (у нашому випадку на заголовку).

4 Вставити об'єкт з буфера командою *Виправлення / Вставити*.

5 Перемістити об'єкт на необхідне місце даного розділу (заголовка).


Зокрема описані дії необхідно виконати для напису з поточною датою. При цьому слід зазначити, що поточна дата визначається за допомогою вбудованої функції *Now* без параметрів, тобто значенням властивості *Дані* для відповідного поля в цьому випадку є значення = *Now()*. Загалом кажучи, якщо в полі необхідно обчислити будь-який вираз, то для властивості *Дані* необхідно вказувати перед цим виразом знак "=", щоб Access сприймав його не як текст, а як значення, що обчислюється, аналогічно редактору електронних таблиць Excel.

Щоб змінити написи *Назва*, *ДатаЗдачі* і *Avg*, необхідно вказати для кожної з них відповідні значення у властивості *Дані*.

Для відображення номерів сторінок у звіті необхідно скористатися функцією *Page*, вказавши її у властивості *Дані* для нового напису, який можна розташувати у верхньому або нижньому колонтитулі.

Крім усіх перерахованих змін, для поліпшення сприйняття даного звіту можна додати кілька ліній, що відокремлюють логічні розділи. Зокрема можна відокремити лініями всіх студентів, а також рядок із визначенням середнього бала кожного студента.

У результаті виконання всіх описаних дій буде отримано звіт **Оцінки**, зображений на рисунку 59.

На закінчення слід зазначити, що для створеного звіту можна встановлювати порядок групування і сортування даних у групах за допомогою кнопки *Сортування і групування* , при натисканні на яку відображається діалогове вікно *Сортування і групування* (рисунок 68). У даному вікні можна додавати, змінювати або видаляти рівні групування так само, які при роботі в режимі таблиці, а також встановлювати порядок сортування для того або іншого рівня.

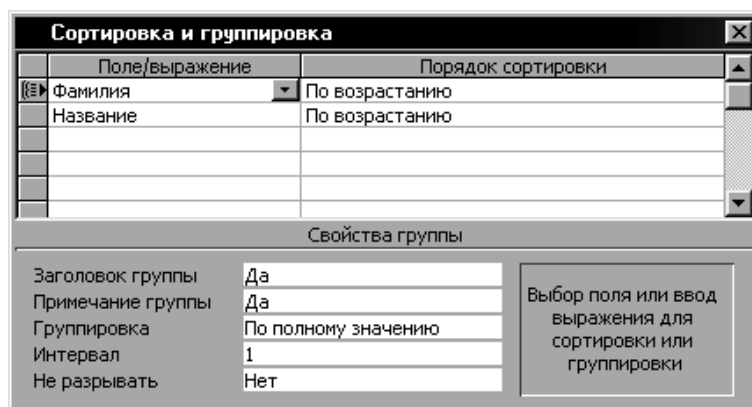


Рисунок 68 – Діалогове вікно *Сортування і групування*

Крім роботи з рівнями групування, для полів звіту можна також використати умовне форматування, аналогічно роботі з формами. Наприклад, якщо середній бал студента більше або дорівнює “4,5”, то в цьому випадку потрібно відображувати значення середнього бала, а також прізвище, ім'я та по батькові студента напівжирним шрифтом.

Щоб встановити умовне форматування для поля, необхідно його виділити і виконати після цього команду *Формат | Умове форматування*. Для поля з обчисленням середнього бала необхідно вказати як *Умова1* такі параметри: *Значення поля, більше або дорівнює, 4,5* і встановити після цього формат шрифту: напівжирний (рисунок 69). З іншого боку, для полів **Прізвище**, **Ім'я** та **ПоБатькові** потрібно вказати умову з використанням поля, у якому обчислюється середній бал студента. За замовчуванням даному полю було привласнено ім'я *Avg – оцінка*. Тому у вікні *Умове форматування* для полів **Прізвище**, **Ім'я** та **ПоБатькові** необхідно як параметр для *Умови1* вибрати *Вираз* і вказати умову *[Avg - оцінка]>=4,5*. Потім потрібно для даної умови також вибрати формат шрифту напівжирний.

На рисунку 70 наведено звіт Оцінки з застосуванням описаного вище умовного форматування.

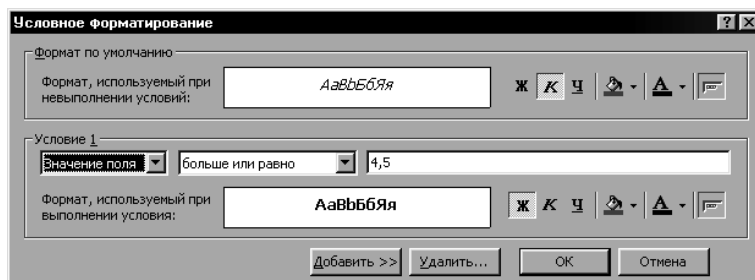


Рисунок 69 – Умовне форматування для середнього балу

<i>Оценки</i>					<i>28 ноября 2000 г.</i>
<i>Фамилия</i>	<i>Имя</i>	<i>Отчество</i>	<i>Предмет</i>	<i>Оценки</i>	<i>Дата сдачи</i>
<i>Жукова</i>	<i>Марина</i>	<i>Владимировна</i>	<i>Физика</i>	5	15.01.99
			<i>Химия</i>	5	20.01.99
			<b>Средний балл</b>	<b>5</b>	
<i>Игнатов</i>	<i>Максим</i>	<i>Валерьевич</i>	<i>Физика</i>	5	15.01.99
			<i>Химия</i>	4	20.01.99
			<b>Средний балл</b>	<b>4,5</b>	
<i>Морозов</i>	<i>Игорь</i>	<i>Стегович</i>	<i>Физика</i>	4	15.01.99
			<i>Химия</i>	4	20.01.99
			<b>Средний балл</b>	<b>4</b>	
<i>Петров</i>	<i>Михаил</i>	<i>Александрович</i>	<i>Математика</i>	5	22.01.99
			<i>Философия</i>	3	10.02.99
			<i>Экономика</i>	4	25.01.00
			<b>Средний балл</b>	<b>4</b>	

Рисунок 70 – Звіт Оцінки з застосуванням умовного форматування



## Розділ 4. ДРУК В ACCESS

### 4.1 Друк опису об'єктів БД

В Access існує можливість створення звіту з описом того або іншого об'єкта обраної БД, наприклад таблиць, запитів і т. д.

Щоб створити звіт з описом, необхідно скористатися командою *Сервіс | Аналіз | Архіваріус*, після чого буде відображено діалогове вікно *Архіваріус* (рисунок 71). У даному вікні можна, відкривши будь-яку вкладку (*Таблиці*, *Запити* і т. д.), вибирати об'єкти, для яких необхідно роздрукувати опис. Крім того, за допомогою кнопки *Параметри* можна встановлювати, які саме властивості об'єкта будуть відображені у звіті з описом.

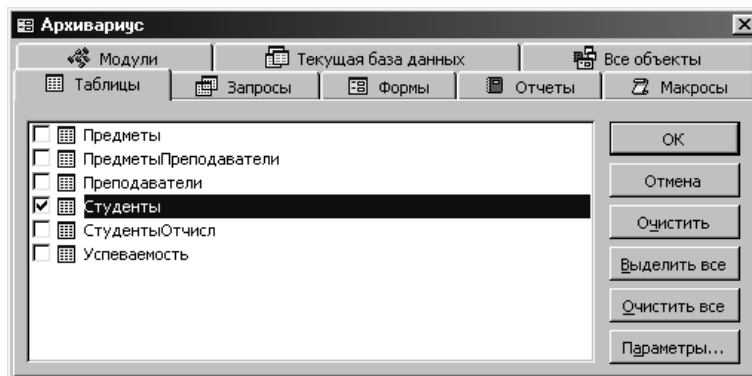


Рисунок 71 – Діалогове вікно *Архіваріус*. Вкладка *Таблиці*

Наприклад, необхідно створити звіт опису таблиці **Студенти**, у який необхідно включити тільки імена полів, типи даних і формати. Для цього треба у вікні *Архіваріуса* встановити прапорець поруч із таблицею **Студенти** (вкладка *Таблиці*) і натиснути кнопку *Параметри*. Потім у діалоговому вікні *Друк опису таблиці* потрібно в розділі *Включити в опис полів* вибрати пункт *імена, типи даних і формати* (рисунок 72) і натиснути *ОК*. Після цього, якщо буде потреба, можна вибрати ще будь-які об'єкти, для яких також необхідно вказати параметри опису. Якщо всі необхідні об'єкти вибрано, то для створення звіту з описом необхідно у вікні *Архіваріуса* натиснути *ОК*. У розглянутому прикладі створюється звіт з описом полів таблиці **Студенти** (рисунок 73).

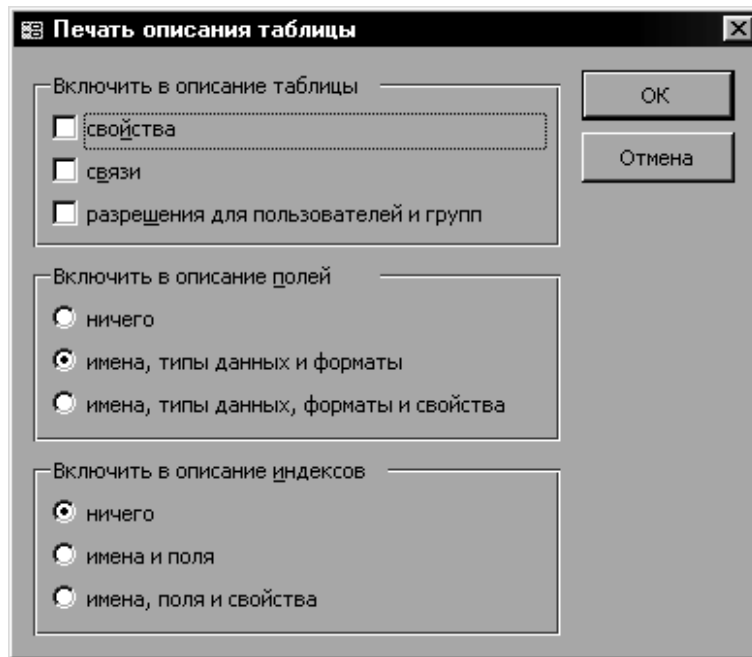


Рисунок 72 – Діалогове вікно *Друк опису таблиці*

Таблица Студенты			29 ноября 2000 г. Страница: 1
<b>Столбы</b>			
Имя	Тип	Размер	
КодСтудента	Длинное целое	4	
Фамилия	Текстовый	20	
Имя	Текстовый	10	
Отчество	Текстовый	15	
Состояние	Текстовый	1	

Рисунок 73 – Опис таблиці Студенти

## 4.2 Друк даних у режимі таблиці

Дані, які зберігаються в таблицях, можуть бути роздруковані. При цьому на інформацію, що друкується, впливають режими фільтрації і сортування, які встановлені для цієї таблиці, а також формат самої таблиці, тобто висота рядків, ширина стовпців тощо.

Наприклад, необхідно роздрукувати інформацію про студентів, тобто прізвище, ім'я та по батькові студента. Для цього необхідно відкрити таблицю **Студенти** і сховати в ній стовпці **КодСтудента** і **Стан**, які не повинні відображатися. Можна також за допомогою команди *Формат | Режим таблиці* встановити в діалоговому вікні *Формат таблиці* параметри для відображення таблиці (рисунок 74).

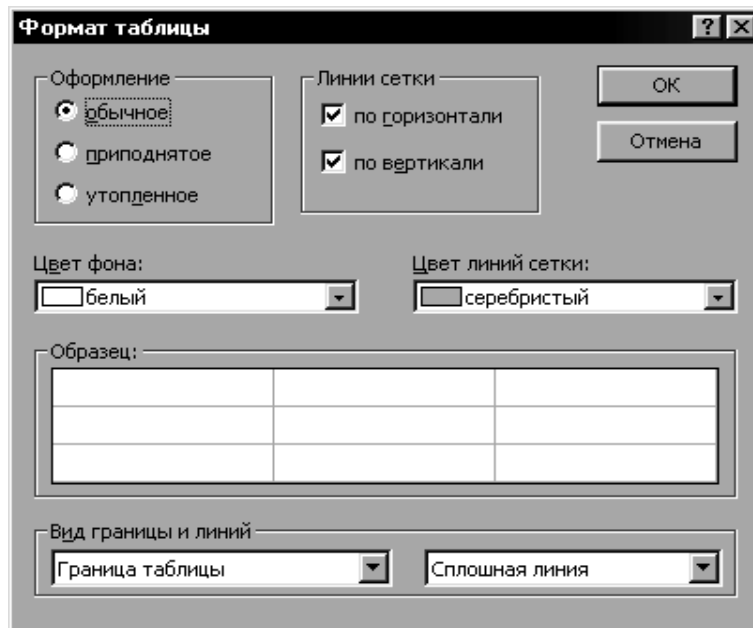


Рисунок 74 – Діалогове вікно *Формат таблиці*

Після того як встановлено формат таблиці, необхідно скористатися командою *Файл | Попередній перегляд*. У даному режимі можна побачити таблицю в тому вигляді, у якому вона буде надрукована.

В Access існує панель інструментів *Попередній перегляд*, що відображується у випадку перегляду даних перед друком (рисунок 75).

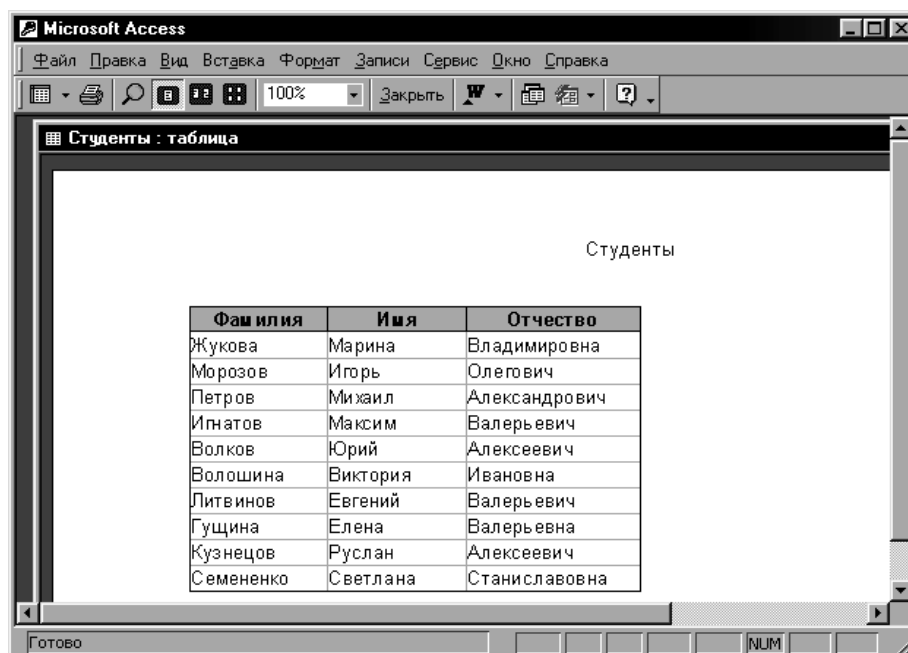




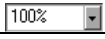



Рисунок 75 – Таблица Студенты. Режим *Попередній перегляд*

Опис основних кнопок цієї панелі наведено в таблиці 6.

Таблиця 6 – Основні кнопки панелі інструментів *Попередній перегляд*

Кнопка	Назва	Опис
	Друк	Відправлення документа на друк
	Одна сторінка	Відображення даних з розташуванням на екрані однієї поточної сторінки
	Дві сторінки	Відображення даних з розташуванням на екрані двох сторінок
	Кілька сторінок	Виведення на екран одночасно декількох сторінок
	Масштаб	Зміна масштабу сторінки
	Закрити	Закриття режиму <i>Попередній перегляд</i>

Слід також зазначити, що для сторінок, що друкують, можуть бути задані параметри: поля, розмір, орієнтація й т. д. Для установлення параметрів необхідно виконати команду *Файл / Параметри сторінки*, після чого буде відображено діалогове вікно *Параметри сторінки* (рисунок 76). На вкладці *Поля* для сторінок, що друкуються, можна встановити розмір полів, а на вкладці *Сторінка* – орієнтацію сторінки (книжкова або альбомна), розмір паперу тощо.

Щоб відправити дані на друк, можна скористатися кнопкою *Друк* панелі *Попередній перегляд*. У цьому випадку документ буде надруковано на поточному принтері з тими налаштуваннями, які для нього встановлені за замовчуванням. З іншого боку, щоб задати деякі параметри друку, наприклад кількість сторінок, що друкуються, або принтер, який необхідно використати для друку, необхідно скористатися командою *Файл | Друк*, після чого буде відображено діалогове вікно *Друк* (рисунок 77). У даному вікні необхідно встановити всі необхідні параметри для друку, після чого потрібно натиснути кнопку *ОК*. У результаті документ буде відправлено на друк.

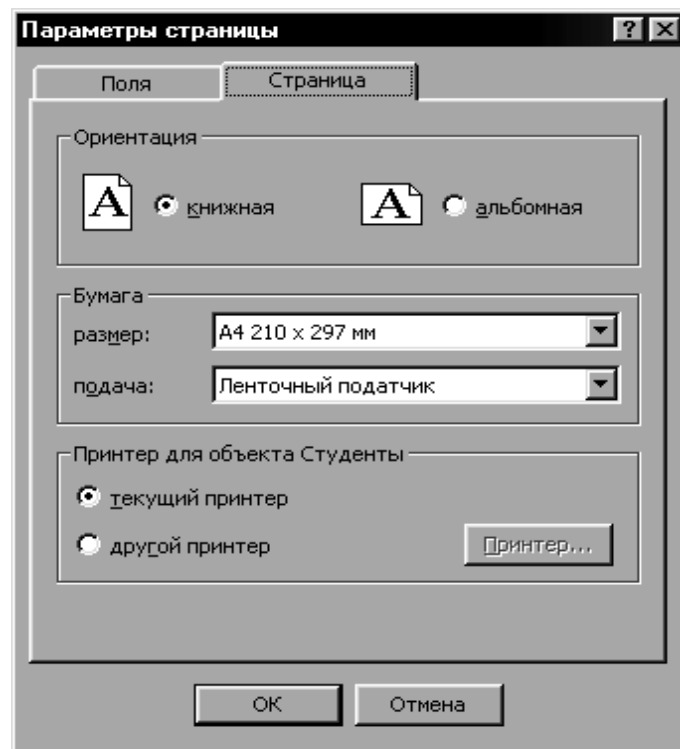


Рисунок 76 – Диалогове вікно *Параметри сторінки*.  
Вкладка *Сторінка*

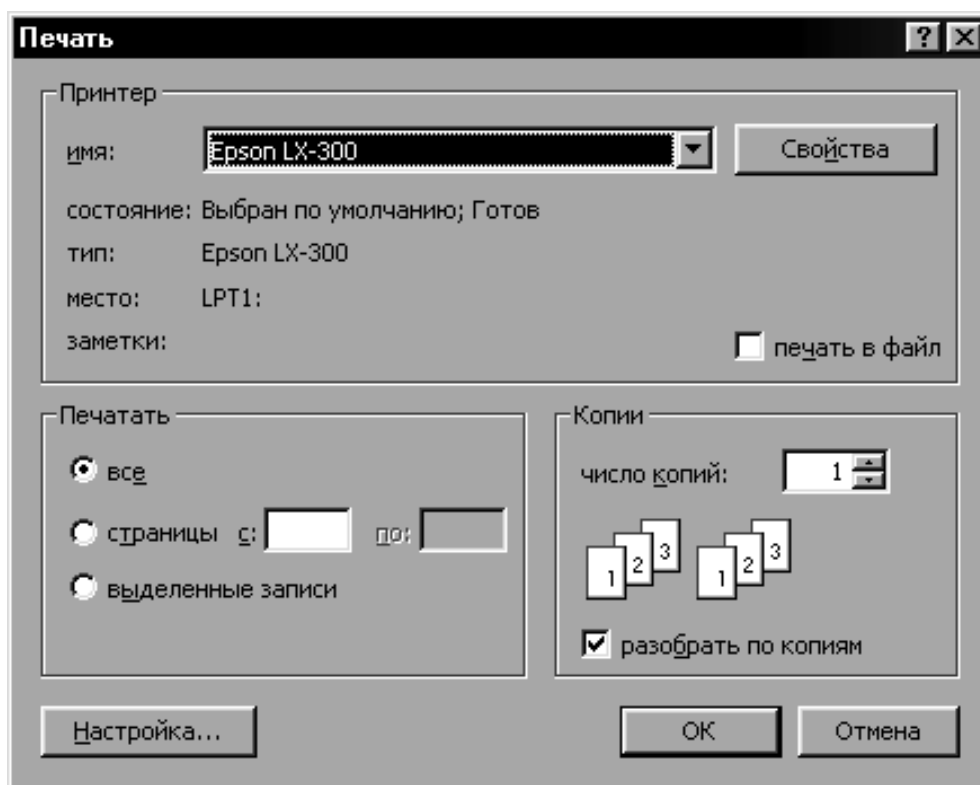


Рисунок 77 – Диалогове вікно *Друк*

### 4.3 Друк форм

Форма в Access може бути використана не тільки для введення і відображення даних, але і для друку тієї інформації, що представлена у формі.

Наприклад, необхідно вивести на друк інформацію про студентів, що відображується у формі **Студенти**. Для цього необхідно змінити форму відповідним чином, щоб кожен запис відображувався в одному рядку, і назви стовпців розмістити в заголовку форми. Після цього можна скористатися кнопкою *Попередній перегляд*, щоб побачити, як форма буде виглядати в друкованому вигляді (рисунок 78). Слід зазначити, що в даному режимі для форми, як і для таблиці, доступна панель інструментів *Попередній перегляд*.



The screenshot shows a window titled "Студенты" with a subtitle "Данные о студентах". It displays a table with four columns: "Фамилия", "Имя", "Отчество", and "Состояние". The data is as follows:

Фамилия	Имя	Отчество	Состояние
Жукова	Марина	Владимировна	<input type="checkbox"/>
Морозов	Игорь	Олегович	<input type="checkbox"/>
Петров	Михаил	Александрович	<input type="checkbox"/>
Игнатов	Максим	Валерьевич	<input type="checkbox"/>
Волков	Юрий	Алексеевич	<input type="checkbox"/>
Волошина	Виктория	Ивановна	<input type="checkbox"/>
Литвинов	Евгений	Валерьевич	<input checked="" type="checkbox"/>
Гущина	Елена	Валерьевна	<input type="checkbox"/>
Кузнецов	Руслан	Алексеевич	<input checked="" type="checkbox"/>
Семенов	Светлана	Станиславовна	<input checked="" type="checkbox"/>

At the bottom, there is a navigation bar with the text "Страница:" and navigation buttons. The current page is 1.

Рисунок 78 – Попередній перегляд форми Студенти

Якщо буде потреба, можна змінити параметри сторінки для форми, що друкується. Для цього, як і у випадку друку таблиці, необхідно скористатися командою *Файл | Параметри сторінки*. У діалоговому вікні *Параметри сторінки* для форми (рисунок 79) є розглянуті вище вкладки *Поля* і *Сторінка*. Однак є також вкладка *Стовпці*, у якій можна встановлювати кількість і

розмір стовпців для відображення даних, тобто в скількох стовпчиках буде розташована інформація на сторінці, а також макет стовпчика:

*зверху вниз* – дані спочатку виводяться в одному стовпці, і коли сторінка закінчується, то виведення здійснюється в наступному стовпці і т. д.;

*зліва направо* – дані виводяться в першому рядку першого стовпчика, потім у першому рядку другого стовпчика і т. д.

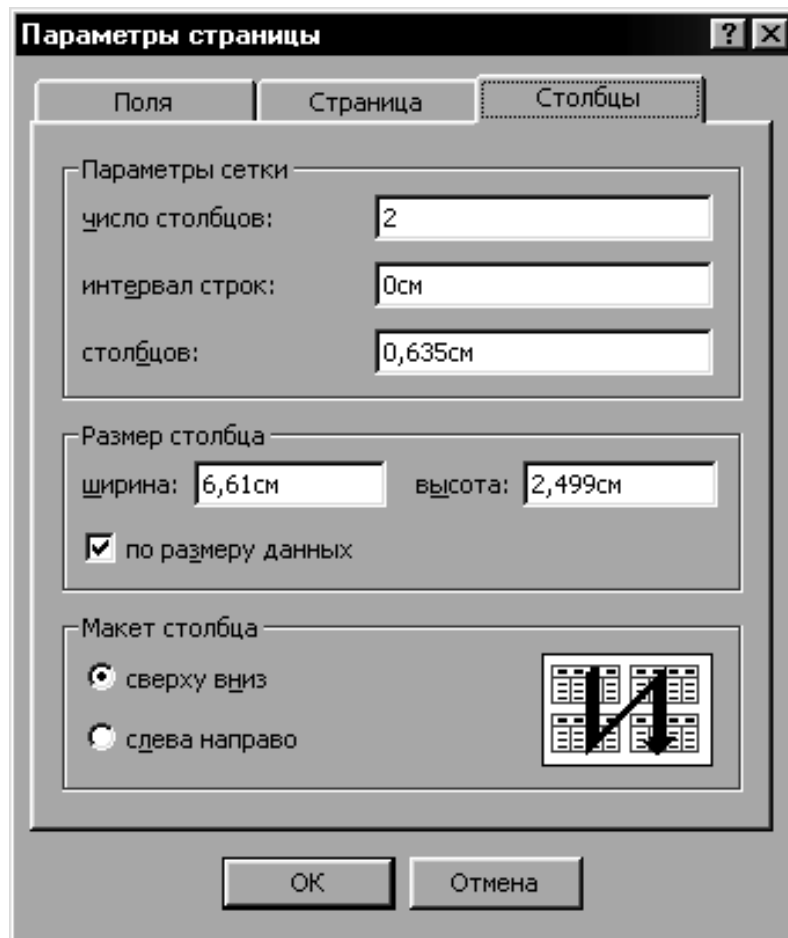


Рисунок 79 – Діалогове вікно *Параметри сторінки*.  
Вкладка *Стовпці*

#### 4.4 Друк звітів

Створені звіти можна переглянути перед друком за допомогою кнопки *Попередній перегляд* аналогічно роботі з таблицями і формами. При цьому також відображується панель

інструментів *Попередній перегляд*, кнопки якої дозволяють встановити параметри перегляду сторінок звіту.

Для зміни параметрів сторінок звіту необхідно скористатися командою *Файл | Параметри сторінки*, після чого необхідно встановити необхідні параметри. Вкладки діалогового вікна *Параметри сторінки* для звіту аналогічні вкладкам для форми (рисунок 79). Зокрема на вкладці *Стовпці* також можна встановити, яка кількість стовпців знадобиться для відображення інформації на сторінці, а також встановити макет стовпчика. З іншого боку, на вкладці *Поля* (рисунок 80) можна змінити розмір полів, які встановлені за замовчуванням.

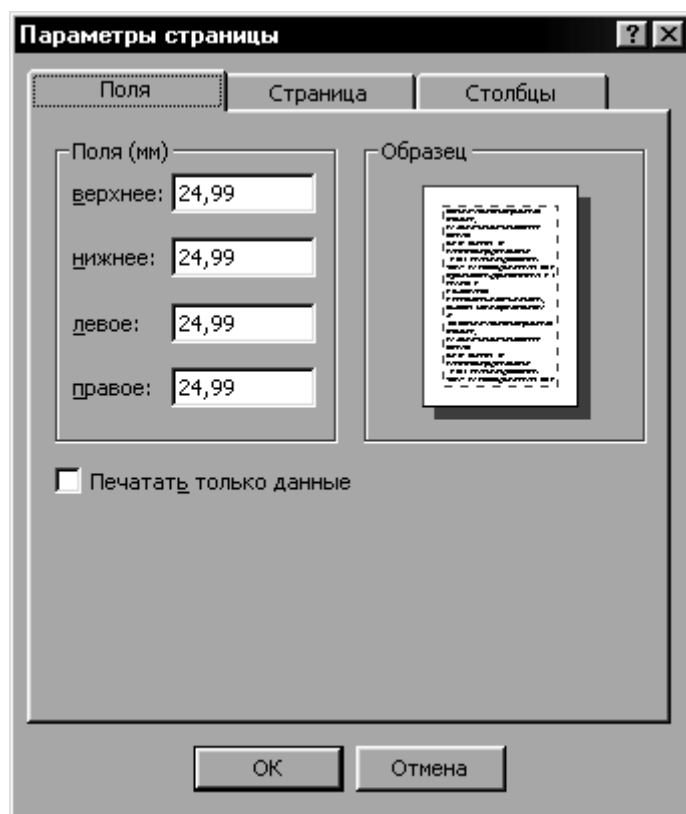


Рисунок 80 – Діалогове вікно *Параметри сторінки*.  
Вкладка *Поля*



## Розділ 5. ЗАВДАННЯ НА ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

### Лабораторна робота 3. СТВОРЕННЯ ЗАПИТІВ

*Вывод запроса на экран.*

1 В окне открытой БД выбираем в списке *Объекты* кнопку *Запросы*.

2 Выделяем *Создание запроса в режиме конструктора*. Нажимаем *ОК*.

3 Добавляем с помощью кнопки *Отобразить таблицу* ТБЛ *Студенты*.

4 Для вывода всех данных таблицы *Студенты* нажимаем на «\*» (в самом верхнем поле) двойным нажатием на левую кнопку мышки.

5 Отображаем запрос на экране с помощью кнопки *Запуск* «!».

6 Сохраняем наш первый запрос. *Файл, Сохранить как, «ЗПРСтуденты»*.

*Знакомство с конструктором.*

1 Создаем запрос в режиме конструктора.

2 Добавляем с помощью кнопки *Отобразить таблицу* ТБЛ *Студенты*.

3 Для работы нам понадобятся фамилия, имя и группа студента.

4 Существует три различных способа выбора нужных нам ячеек из той или иной таблицы:

1) двойное нажатие левой кнопкой мышки по нужной нам ячейке (добавим таким способом ячейку *ФамилияСтудента*);

2) зажав нужную ячейку левой кнопки мышки и перетаскив на свободное поле (добавим ячейку *ИмяСтудента*);

3) выбрав нужную нам ячейку из выплывающего списка (проведем данную процедуру над ячейкой *ГруппаСтудента*).

5 Запускаем запрос, нажав на кнопку *Запуск* «!».

6 Убеждаемся, что все три колонки с фамилиями, именами и группами студентов отобразились корректно.

7 Переходим в режим конструктора.

8 Ознакомимся с его структурой (рисунок 59):

В верхней части окна конструктора располагаются таблицы (или таблица), используемые в качестве источника данных. В нижней части окна указываются параметры отбираемых в выборку данных:

- 1) поле;
- 2) имя таблицы;
- 3) сортировка;
- 4) вывод на экран;
- 5) условие отбора.

Для детального объяснения последней ячейки, выполним следующие действия:

- 1) создаем многотабличный запрос;
- 2) добавляем с помощью кнопки *Отобразить таблицу* ТБЛ Успеваемость и ТБЛ Дисциплины;
- 3) выбираем ячейки *НазваниеДисциплины*, *Оценка*;
- 4) теперь у нас в нижней части окна располагаются 5 колонок: *ФамилияСтудента*, *ИмяСтудента*, *ГруппаСтудента*, *Дисциплина*, *Оценка*;

**Задание.** Вывести всех отличников на экран.

Вспоминаем параметры записи *расширенного фильтра*. Т. к. студенты оцениваются по стобальной шкале, отличники – это все студенты, что получил 90 баллов и выше. Поэтому, в колонке *Оценка* в ячейке *Условие отбора* записываем *Like «9\*»*. Но т. к. фильтр отсеет все значения, кроме диапазона от 90 до 99, нам нужно еще добавить значение 100. Вписываем в ячейку *Или* цифру *100*. Альтернатива записи: в ячейке *Условие отбора* вписываем *Like «9\*» or 100*.

- 1 Отображаем запрос на экране с помощью кнопки *Запуск «!»*.
- 2 Сохраняем запрос. *Файл*, *Сохранить как*, «ЗПРОтличники».
- 3 Не закрывая запрос, переходим в режим конструктора.
- 4 Проверим, вовремя ли студент сдал предмет и законно ли поставлена оценка *отлично*. Добавим ячейку *ДатаСдачи*.
- 5 Выводим запрос на экран с помощью «!».
- 6 Обращаем внимание, что фамилии расположены в произвольном порядке.

7 Для того чтобы упорядочить фамилии по алфавиту, переходим в режим конструктора.

8 В ячейке *Сортировка* выставляем *По возрастанию*.

9 Выводим запрос на экран и убеждаемся, что сортировка выполнена корректно.

10 Сохраняем запрос под именем ЗПРСтуденты\_Отличники.

11 Переходим в режим конструктора.

12 Добавляем в условие отбора конкретную группу студентов и выводим запрос на экран с помощью «!».

13 Сохраняем запрос ЗПРСтуденты\_Отличники\_(маркировка группы).

**Задание 1.** Посмотреть успеваемость студентов во всех группах при двух условиях:

1 Чтобы и фамилии студентов, и группы отсортировались по возрастанию.

2 Так сгруппировать студентов, чтоб они отобразились в алфавитном порядке в своих группах, а не единым алфавитным списком на весь ВУЗ. Пример:

Группа Студента	ФИО Студента
2-3-АТС	Иванов Иван Иванович
2-3-АТС	Лаврентьева Ольга Николаевна
2-3-АТС	Семенов Валентин Сергеевич
4-4-АТС	Антонов Михаил Олегович
4-4-АТС	Козлова Анна Николаевна
4-4-АТС	Петренко Екатерина Семеновна
4-4-АТС	Татарский Виктор Александрович
4-4-АТС	Якушев Леонид Данилович
4-5-УПП	Краснокутская Евгения Васильевна
4-5-УПП	Романенко Алексей Федорович
4-5-УПП	Шевченко Алексей Николаевич
7-2-ИЯ	Бондаренко Марина Анатольевна
7-2-ИЯ	Никулин Владимир Сергеевич

**Решение:**

1 Переходим в режим конструктора.

2 Добавляем с помощью кнопки *Отобразить таблицу* ТБЛ Студенты, ТБЛ Дисциплины, ТБЛ Успеваемость.

3 Для работы нам понадобятся ячейки: *ФамилияСтудента*, *ИмяСтудента*, *ОтчествоСтудента*, *ГруппаСтудента*, *Дисциплина*, *Оценка*.

4 В колонках *ФамилияСтудента* и *ГруппаСтудента* выставляем сортировку *По возрастанию*.

5 Выводим запрос на экран и видим, что все студенты ВНЗ у нас разместились в алфавитном порядке, но никак не по группам.

Переходим обратно в режим конструктора, чтобы решить данную проблему и справиться с поставленной задачей.

6 Переносим колонку *ГруппаСтудента* на первое место.

7 Таким образом, у нас в первую очередь отсортируются группы, а в них, уже во вторую очередь, будут сортироваться студенты по фамилиям.

8 Отображаем запрос на экране. Задание выполнено.

*Задание 2. Отобразить на экране успеваемость студентов с такой же сортировкой, как указано в задании 1, но при этом добиться того, чтобы на первом месте выводилась на экран колонка с фамилией студента, далее шла колонка с именем, потом с отчеством и только после этого шла колонка с группой студента.*

Задание решить самостоятельно. Сохранить запрос.

*Параметрический запрос с клавиатуры.*

1 Для работы будем использовать предыдущую форму запроса.

2 Для примера, создадим параметрический запрос с клавиатуры по фамилии студента.

3 В колонке *ФамилияСтудента*, в ячейке *Условие отбора* пишем [Введите фамилию студента].

4 Нажимаем кнопку *Запуск «!»*.

5 Теперь, чтобы найти конкретного студента в нашем списке, достаточно будет ввести лишь его фамилию и нажать кнопку *ОК*.

6 Также возможна манипуляция не с одним видом данных, а сразу с несколькими, для сужения круга поиска. Например,

поставить параметрический запрос с клавиатуры на Фамилию, имя и группу студента.

**Самостоятельная работа.** Создать запрос по выборке. Вывести информацию о нагрузке преподавателей с сортировкой по кафедре и фамилиям (вид сортировки аналогичный заданию 2). Создать параметрический запрос с вводом названия кафедры. Сохранить запрос.

#### **Лабораторна работа 4. РОБОТА З ГРУПОВИМИ ОПЕРАЦІЯМИ**

1 В окне открытой БД выбираем в списке *Объекты* кнопку *Запросы*.

2 Выделяем *Создание запроса в режиме конструктора*. Нажимаем *ОК*.

3 Добавляем с помощью кнопки *Отобразить таблицу* ТБЛ Студенты, ТБЛ Успеваемость.

4 Работать будем с ячейками КодСтудента, ФамилияСтудента, ИмяСтудента, ОтчествоСтудента, ГруппаСтудента, Дисциплина, Оценка.

5 Для работы с групповыми операциями нажимаем иконку  $\Sigma$  (Totals).

**Задание.** Узнать средний балл каждого студента.

1 В колонке Оценка выбрать в сплывающем меню Avg.

2 Запрашиваем информацию («!»).

3 Видим, что ничего не изменилось, т. к. БД вывела среднее значение (средний балл студента) по каждой дисциплине.

**Вопрос:**

*Как вывести общий средний балл по всем дисциплинам у каждого студента?*

Задание решить самостоятельно.

**Задание.** Подсчитать общее количество оценок каждого студента.

1 Добавляем еще одну колонку Оценка.

- 2 В сплывающем меню выбираем Count.
- 3 Переходим в режим ТБЛ.
- 4 Видим, что «шапки» колонок названы по умолчанию Avg-оценка и Count-оценка.

**Вопрос:**

*Как переименовать «шапку» колонок Avg-оценка и Count-оценка?*

**Решение:**

Достаточно в запросе в верхней ячейке поставить знак двоеточия («:») перед названием используемой таблицы (в данном случае ТБЛ Успеваемость). А перед «:» написать новое имя «шапки» данной колонки.

Теперь ячейки должны выглядеть следующим образом:

*Средний балл:Оценка.*

*Кол-во оценок:Оценка.*

Сохраняем запрос.

**Вопрос:**

*Как вывести успеваемость каждой из групп?*

Задание решить самостоятельно.

**Самостоятельная работа 1.** Подсчитать и вывести, во-первых, количество студентов в группе и, во-вторых, количество студентов в университете.

**Самостоятельная работа 2.** Вывести нагрузку, во-первых, преподавателей и, во-вторых, запросом, нагрузку кафедры.

Для запроса понадобятся:

1 ТБЛ Преподаватели и ТБЛ Дисциплины.

2 Ячейки КодПреподавателя, ФамилияПреподавателя, ИмяПреподавателя, ОтчествоПреподавателя, КафедраПреподавателя, НазваниеДисциплины, Часы.

Сохранить оба запроса.

## Работа с перекрестными запросами. Создание сводной ведомости

- 1 Создаем новый запрос в режиме конструктора.
- 2 Добавляем с помощью кнопки *Отобразить таблицу* ТБЛ Студенты, ТБЛ Успеваемость.
- 3 Работать будем с ячейками *ФамилияСтудента*, *КодДисциплины*, *Оценка*.
- 4 Тип запроса выставляем *CrossTab* (перекрестный).
- 5 Подключаем групповые операции, нажатием на иконку  $\Sigma$  (Totals).
- 6 В ячейке *ФамилияСтудента* выставляем *crosstab*, *заголовки строк* (учитывая, что в качестве заголовков строк необходимо определить хотя бы одно поле).
- 7 В ячейке *КодДисциплины* выставляем *crosstab*, *заголовки столбцов* (учитывая, что в качестве заголовков столбцов необходимо определить только одно поле).
- 8 В ячейке *Оценка* выставляем *crosstab*, *значение* (учитывая, что в качестве значения необходимо определить только одно поле) и *groupby* (*Avg*), для вывода среднего балла студента.
- 9 Выводим запрос на экран.
- 10 Видим, что все дисциплины вывелись в цифровом виде (по коду каждой дисциплины).
- 11 Добавляем ТБЛ *Дисциплины* (т. к. ячейки *НазваниеДисциплины* в ТБЛ *Успеваемость* нет).
- 12 Удаляем в конструкторе запроса ячейку *КодДисциплины* и вместо него добавляем ячейку *Название Дисциплины*, выставляем *crosstab*, *заголовки столбцов*.
- 13 Выводим запрос на экран.

В сводных ведомостях одной фамилии студента, как правило, недостаточно. Необходимо вписать как имя студента, отчество студента, так и его группу. Для этого:

- 1 Добавляем ячейки *ИмяСтудента*, *ОтчествоСтудента*, *ГруппаСтудента*.
- 2 Выставляем в добавленных ячейках поле *crosstab*, *заголовки строк* (помним, что в качестве заголовков строк мы имеем право определить одно и более значений).

3 Выводим запрос на экран.

4 Видим, что фамилия, имя, отчества и группа каждого студента находится в отдельной ячейке.

**Вопрос:**

*Как объединить ячейки с фамилией, именем, отчеством и группой студента в одну общую?*

**Решение:**

1 Необходимо подключить построитель выражений с помощью кнопки *Построить*.

2 В левом поле выбираем *table* и в нем находим ТБЛ Студенты.

3 Добавляем (двойным нажатием по левой кнопке мыши) ячейки *ИмяСтудента*, *ОтчествоСтудента* и *ГруппаСтудента*. Данные ячейки отобразятся в среднем поле.

4 В верхнем поле заменяем все «*expr*» на *&* (знак объединения);

5 Нажимаем *ОК*.

6 Выводим запрос на экран.

7 Для того чтобы расставить пробелы, снова переходим в построитель выражений и в верхнем поле рядом со знаком *&* вставляем пробел в двойных кавычках (« *»*).

Строка должна иметь следующий вид: *[...]/![...] & « » & [...]/![...]* и т. д.

8 Выводим запрос на экран.

9 Двойным нажатием левой кнопки мыши по границам ячеек (в режиме ТБЛ) делаем автоподбор ширины колонок.

**Самостоятельно** переименовать название колонки с ФИО и группой студентов.

Сохранить запрос.

**Самостоятельная работа 3.** Вывести сводную ведомость преподавателей и их нагрузку. В поле *Значение* указать общее количество часов.

Сохранить запрос.



**Задание.** Создать запрос вывода ФИО студентов и группу каждого студента. Заменить полные имена и отчества студентов на инициалы.

1 Создаем новый запрос в режиме конструктора.

2 Добавляем с помощью кнопки *Отобразить таблицу* ТБЛ Студенты.

3 Добавляем ячейки *ФамилияСтудента*, *ИмяСтудента*, *ОтчествоСтудента*, *ГруппаСтудента*.

4 Устанавливаем курсор в строке *Поле* *ФамилияСтудента*.

5 Нажимаем кнопку *Построить*, тем самым запуская построитель выражений.

6 В верхнем поле после *ФамилияСтудента* ставим *&« »&* для добавления пробела и возможности добавления функции.

7 Выбираем в левом поле *Функции*, далее *Встроенные функции*.

8 В среднем поле выбираем *Текстовые*.

9 В правом поле выбираем *Left*.

10 Устанавливаем курсор в верхнем поле в то место выражения, куда данная функция должна быть вставлена, и нажимаем кнопку *Вставить* в окне построителя выражений.

11 В результате этих действий в выражение будет добавлен шаблон функции *Left*:

*Left* («stringexpr»; «n»).

12 Вместо «stringexpr» вписываем *ИмяСтудента*. Вместо «n» – количество символов, которые следует оставить слева. Т. к. в инициалах, как правило, используют лишь первую букву имени, ставим вместо «n» цифру *1*. Выражение должно иметь следующий вид:

[*ФамилияСтудента*] &« »&Left([*ИмяСтудента*];1).

13 После инициала имени следует поставить точку. Теперь выражение имеет вид:

[*ФамилияСтудента*] &« »&Left([*ИмяСтудента*];1) & «.».

**Самостоятельно** проделать аналогичные действия и добавить инициал отчества к фамилии и инициалу имени студента.

## Лабораторна работа 5. СТВОРЕННЯ ФОРМ

1 В окне открытой БД выбираем в списке *Объекты* кнопку *Таблицы*.

2 Выделяем *ТБЛСтуденты*. Нажимаем на иконку *Автоформа*.

3 Знакомство с содержанием автоформы: рабочие поля, кнопки.

4 Переходим в режим конструктора.

5 Знакомство с *Областью данных*.

6 Знакомство с *Панелью элементов*.

### **Самостоятельная работа:**

1 В режиме конструктора изменить внешний вид формы на свой вкус с помощью манипуляции над размерами и цветом шрифта, а также расположением рабочих ячеек.

2 В режиме формы изменить информацию одного из студентов.

3 В режиме формы добавить нового студента и полностью заполнить информацию о нем.

1 Сохраняем форму. Называем ФРМСтуденты.

2 В окне открытой БД выбираем в списке *Объекты* кнопку *Таблицы*.

3 Выделяем *ТБЛОценки*. Открываем выплывающий список в иконке *Автоформа*.

4 Выбираем *Форма*.

5 Выбираем *Автоформа: ленточная*. В нижнем поле должна быть выбрана *ТБЛОценки*. Нажимаем *ОК*.

6 Переходим в режим конструктора формы.

7 Знакомство с внешним видом ленточной автоформы.

8 Знакомство с областями: *Заголовок формы*, *Область данных*, *Примечание формы*.

9 В *Примечании* формы создаем поле *Средний бал*: выбираем в *Панели элементов* иконку *Поле* и перетаскиваем ее на область примечания формы. Вместо слова *Поле* записываем *Средний бал*. В ячейке *Свободный* записываем  $=avg([Оценка])$ .

10 Переходим в режим формы и проверяем, корректно ли отображается средний балл студентов.

11 Применяем сортировку и фильтры в ленточной форме.

**Самостоятельная работа:**

1 Добавить в примечание формы подсчет количества оценок.

2 Заменить расположение/размер/цвет/жирность шрифта на свой вкус. Сохранить форму.

3 Создать ленточную форму таблицы Преподаватели. Подсчитать количество преподавателей в ВУЗе. Сохранить форму.

**Вопрос:**

Как вывести информацию о преподавателях и подсчитать их итоговую нагрузку?

**Решение:**

1 В окне открытой БД выбираем в списке *Объекты* кнопку *Таблицы*.

2 Выделяем ТБЛОценки. Открываем выплывающий список в иконке *Автоформа*.

3 Выбираем *Форма*.

4 Выбираем *Мастер форм*. Нажимаем *ОК*.

5 Выбираем в поле *Таблицы и Запросы* ТБЛПреподаватели. В поле *Доступные поля* выбираем *КодПреподавателя*, *ФамилияПреподавателя*, *ИмяПреподавателя*, *ОтчествоПреподавателя*, *КафедраПреподавателя*.

6 В поле *Таблицы и Запросы* переходим на ТБЛДисциплины. В поле *Доступные поля* выбираем *НазваниеДисциплины*, *КоличествоЧасов*.

7 Нажимаем *Далее*.

8 Выбираем внешний вид формы: *Ленточный*.

9 Нажимаем *Далее*.

10 Самостоятельно выбираем понравившийся стиль формы;

11 Нажимаем *Далее*.

12 Выделяем *Открываем форму для просмотра и ввода данных*.

13 Нажимаем *Готово*.

14 Переходим в режим конструктора формы.

15 Выбираем в *Панели элементов* иконку *Поле* и перетаскиваем ее на область примечания формы. Вместо слова *Поле* записываем *Итоговая нагрузка*. В ячейке *Свободный* записываем  $=sum([КоличествоЧасов])$ .

16 Сохраняем форму.

## **Лабораторна робота 6. ЗНАЙОМСТВО ЗІ ЗВІТАМИ**

Все выполняется аналогично лабораторной работе 5.

## СПИСОК ЛИТЕРАТУРИ

1 Бекаревич, Ю.Б. Самоучитель Microsoft Access 2009 [Текст] / Ю.Б. Бекаревич, Н.В. Пушкина. – СПб.: БХВ – Петербург, 2009. – 720 с.

2 Дейт, К. Дж. Введение в системы баз данных, 6-е изд. [Текст] / К. Дж. Дейт. – К.; М.; СПб.: Издательский дом "Вильямс", 2008. – 848 с.

3 Калянов, Г.Н. CASE-технологии: Консалтинг в автоматизации бизнес-процессов [Текст] / Г.Н. Калянов. – 3-е изд. – М.: Горячая линия-Телеком, 2008. – 320 с.

4 Маклаков, С.В. BPWin, ERWin. CASE – средства разработки информационных систем [Текст] / С.В. Маклаков. – М.: Диалог-МИФИ, 2007 – 300 с.

5 Мандрыкин, А.В. Информационные технологии в экономике [Текст]: учеб. пособие / А.В. Мандрыкин, А.В. Непышневский. – Воронеж: ВГТУ, 2008. – 235 с.

6 Михеева, В.Д. Microsoft Access 2002. [Текст] / В.Д. Михеева, И.А. Харитонова.– СПб.: БХВ – Петербург, 2007. – 1040 с.

