

Розділ 10

Групування даних

Повторення

- ◆ Що означає «згрупувати рядки електронної таблиці за значеннями певного параметра»?
- ◆ Які операції можна виконувати над згрупованими рядками електронної таблиці?
- ◆ Опишіть алгоритм, за яким виконується функція DSUM (рос. БДСУММ) табличного процесора?
- ◆ Опишіть загальну структуру вибіркового запиту SQL.

На попередньому занятті ми вибирали в таблицях записи, що задовольняють певні критерії. Але в багатьох випадках вибрати записи — значить зробити тільки половину справи, адже часто з ними потрібно виконати якісь підсумкові операції: підрахувати їх кількість, підсумувати значення того чи іншого поля тощо. Підсумкову операцію, наприклад обчислення середнього віку учнів у кожному класі або визначення кількості класів, у яких викладає кожен учитель, часто виконують відразу над багатьма *групами записів*. У табличному процесорі для виконання таких завдань, нагадаємо, використовують проміжні підсумки та зведені таблиці, а в СКБД застосовують спеціальні різновиди запитів. Крім того, у СКБД Microsoft Access засобом відображення на аркушах друкованого формату підсумкових характеристик для груп записів є *звіти*. Ці засоби ми розглянемо на сьогоднішньому уроці.

Групові операції в запитах

Уважно розгляньте таблицю 10.1. Це таблиця Учні, відсортована за значеннями поля клас. Один із результатів сортування полягає в тому, що записи учнів, які навчаються в одному класі, розташовано поруч. Інакше кажучи, таблицю *згруповано* за полем клас: спочатку розташовано групу учнів 10А класу, потім — 10Б, 11А, і нарешті — 11Б класу.

код	прізвище	ім'я	дата народження	стать	клас
1	Шпак	Максим	11.12.1996	ч	10А
5	Григорук	Петро	05.05.1997	ч	10Б
6	Райчук	Олена	12.01.1998	ж	10Б
2	Хоменко	Ірина	09.09.1995	ж	11А
3	Недригайло	Петро	01.07.1995	ч	11Б
4	Буджак	Оксана	14.01.1996	ж	11Б

Табл. 10.1. Таблиця Учні, відсортована за полем клас

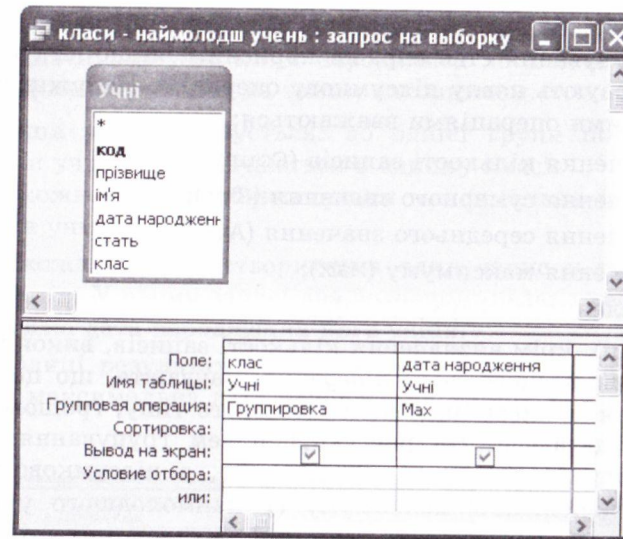
У цій таблиці легко обчислювати підсумкові показники для груп записів. Наприклад, у табл. 10.2 показано результат обчислення для кожного класу кількості учнів і дати народження наймолодшого з них.

Групування шляхом сортування ми виконували в табличному процесорі Microsoft Excel. У СКБД Microsoft Access для групування записів сортувати таблицю не потрібно. Групування і обчислення підсумкових операцій здійснюють за допомогою конструктора запитів. У вікні конструктора потрібно клацнути правою кнопкою миші поле, за значеннями якого будуть виділятися групи, і вибрати у контекстному меню команду **Групповые операции** (Групові операції). Після цього в бланк запиту буде додано рядок **Групповая операция** (рис. 10.1, а).

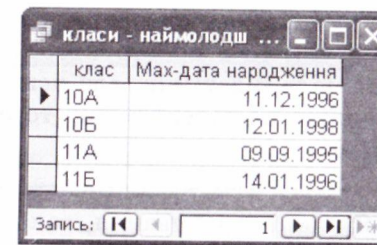
код	прізвище	ім'я	дата народження	стать	клас
1	Шпак	Максим	11.12.1996	ч	10А
Дата народження наймолодшого учня			11.12.1996		
Кількість учнів	1				
5	Григорук	Петро	05.05.1997	ч	10Б
6	Райчук	Олена	12.01.1998	ж	10Б
Дата народження наймолодшого учня			12.01.1998		
Кількість учнів	2				
2	Хоменко	Ірина	09.09.1995	ж	11А
Дата народження наймолодшого учня			09.09.1995		
Кількість учнів	1				
3	Недригайло	Петро	01.07.1995	ч	11Б
4	Буджак	Оксана	14.01.1996	ж	11Б
Дата народження наймолодшого учня			14.01.1996		
Кількість учнів	2				

Табл. 10.2. Обчислення підсумкових показників для груп записів у табличному процесорі

З рис. 10.1, б видно, що результати групування рядків таблиці в табличному процесорі та в СКБД дещо відмінні. У СКБД результатом запиту з групуванням є таблиця, де кожна група записів початкової таблиці «згортається» в один запис (наприклад, на рис. 10.1, б замість двох записів «10Б» ми бачимо один). Власне об'єднання кількох записів в один в реляційних базах даних і називається групуванням.



а



б

Рис. 10.1. Групування таблиці: а — бланк запити; б — результат виконання запити

Група — це набір записів з однаковим значенням певного поля. Під групуванням розуміють процес об'єднання групи записів в один; поле, значення якого однакові, називається **полем групування**.

Однак із самого факту згортання кількох записів в один мало зиску. Групування стає справді корисним, якщо до груп записів застосовують певну підсумкову операцію. Найуживанішими підсумковими операціями вважаються:

- ◆ визначення кількості записів (Count);
- ◆ обчислення сумарного значення (Sum);
- ◆ обчислення середнього значення (Avg);
- ◆ обчислення максимуму (Max);
- ◆ обчислення мінімуму (Min).

Усі операції, крім визначення кількості записів, виконують над значеннями певного *поля підсумків*. Зауважте, що це не поле групування, а якесь інше поле числового типу, грошового типу або типу дата/час. На рис. 10.1 полем групування є клас, полем підсумків — дата народження, а підсумковою операцією — обчислення максимуму (у наймолодшого учня дата народження максимальна). Тип підсумкової операції вибирають зі списку, що розкривається кнопкою ▾, розташованою в клітинках рядка **Групповая операция** справа.

Зазначимо, що коли підсумкова операція полягає у визначенні кількості записів у групі, то в конструкторі запитів ми також маємо вибрати певне поле підсумків, але неважливо, яке саме. Справді, кількість рядків не залежить від того, за яким стовпцем ми її підраховуємо.

Для допитливих. Як полів групування, так і полів підсумків у запиті може бути кілька. Якщо використовують кілька полів групування, то до однієї групи відносять записи, які в усіх цих полях мають однакові значення.

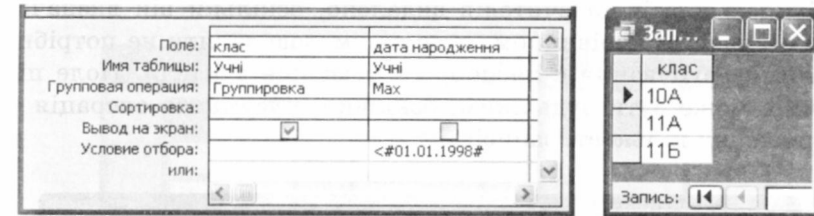
Завдання 10.1

У базі даних **школа** створіть і виконайте запит *визначити дату народження наймолодшого учня в кожному класі*.

Поле підсумків можна використовувати в умовах відбору так само, як і будь-яке інше поле. Самі умови записують у рядку **Условие отбора** конструктора запитів. Як приклад розглянемо

подані на рис. 10.2 бланк і результат виконання запити «Визначити класи, всі учні яких народилися до 1 січня 1998 року». СКБД виконує запит за таким алгоритмом.

1. Таблиця учнів групується: до однієї групи потрапляють записи учнів, які навчаються в одному класі.
2. Для кожної групи обчислюється максимальна дата народження учня.
3. Для кожної групи створюється один запис у таблиці результатів. У цьому записі два значення: назва класу та максимальна дата народження його учнів.
4. У таблиці результатів залишаються тільки ті записи, для яких максимальна дата народження менша за 1 січня 1998 року.



а

б

Рис. 10.2. Запит «Визначити класи, всі учні яких народилися до 1 січня 1998 року»: а — бланк запити; б — результат виконання

Запити з групуванням можна створювати не за однією, а за кількома таблицями, з'єднуючи їх так само, як і в запитах без групування. Приклади кількотабличних запитів із групуванням буде розглянуто далі.

Для допитливих. У запитах із групуванням можна вказувати лише поля групування та підсумків, адже тільки вони мають одне значення для кожної групи і тому можуть бути подані у вигляді одного запису. «Звичайного» поля не може бути, оскільки в кожній групі його значень багато і незрозуміло, яке з них відобразити в таблиці результатів.

Завдання 10.2

Створіть і виконайте запит *визначити класи, всі учні яких народилися до 1 січня 1998 року.*

Приклади запитів

Приклад 1

Для кожного вчителя визначити, скількох учнів він навчає.

Вікно конструктора запиту зображено на рис. 10.3. Прізвища та паспорти вчителів беремо з таблиці *Учителі*, яку з'єднано з таблицею *Викладання*, її — з таблицею *Класів*, а вже таблицю *Класів* — з таблицею *Учні*. Зауважте, що прямий зв'язок між таблицею *Класів* та вчителів видалено, оскільки він визначає, хто є класним керівником, а це за умовою запиту не потрібно. Полями групування є паспорт та прізвище вчителя. Поле підсумків може бути будь-яким, оскільки підсумкова операція — обчислення кількості записів.

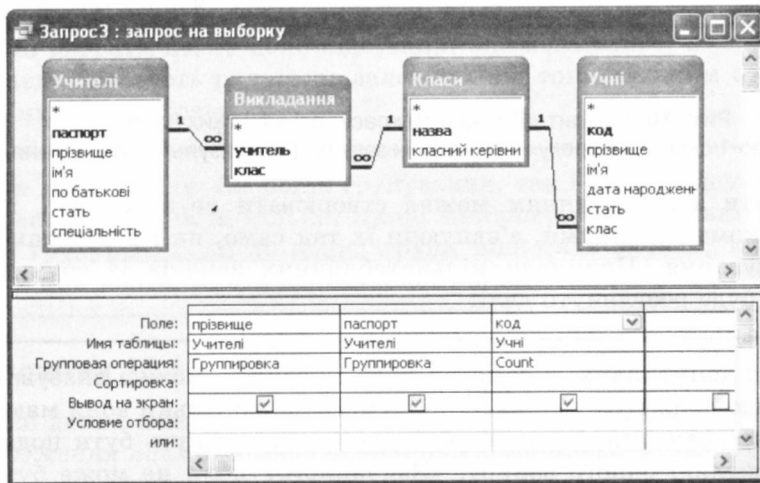


Рис. 10.3. Вікно конструктора запиту «Для кожного вчителя визначити, скількох учнів він навчає»

Приклад 2

Припустимо, що до таблиці *учнів* додано поле *успішність*, у якому зберігається середній бал в таблиці за минулий навчальний рік. Запит є таким: *визначити прізвища класних керівників тих класів, середня успішність у яких нижча за 8 балів.*

Вікно конструктора запиту зображено на рис. 10.4. У запиті два поля групування: прізвище вчителя та назва класу. Групувати записи за назвою класу потрібно тому, що саме для всіх учнів кожного класу обчислюється середня успішність, а за прізвищем вчителя — тому, що значення цього поля відображається. Пояснимо друге твердження. У запиті з групуванням, нагадаємо, будь-яке поле має бути полем групування або підсумків. Прізвище вчителя — це не підсумкове значення, а отже, відповідне поле має бути полем групування, а не підсумків.

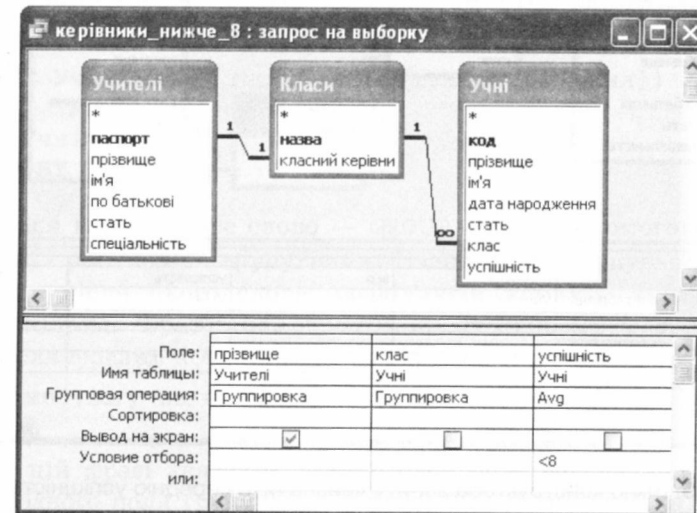


Рис. 10.4. Вікно конструктора запиту «Визначити прізвища класних керівників тих класів, середня успішність у яких нижча за 8 балів»

Зауважте також, що в цьому запиті на результат підсумкової операції накладено умову відбору (середня успішність нижча 8 балів). Зрозуміло, що ця умова перевірятиметься для вже згру-

пованої таблиці, тобто відповідати чи не відповідати їй будуть цілі класи.

Приклад 3

Визначити середню успішність учнів, яких навчає Сошко Катерина.

Вікно конструктора запиту зображено на рис. 10.5. У цьому запиті умову відбору (учитель — Сошко Катерина) накладено на поле групування. Можна вважати, що запит виконується так: після з'єднання таблиць відбираються записи, що стосуються Сошко Катерини, а потім обчислюється середнє значення поля успішність в отриманій таблиці.

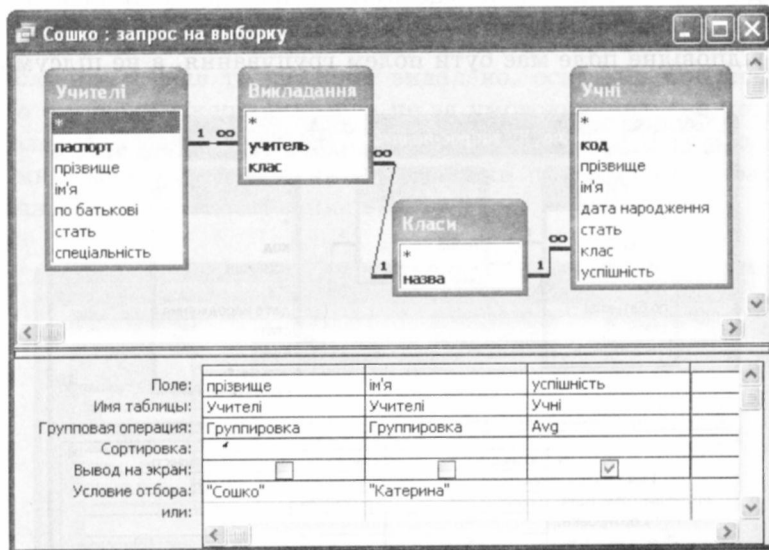


Рис. 10.5. Вікно конструктора запиту «визначити середню успішність учнів, яких навчає Сошко Катерина»

Завдання 10.3

Реалізуйте запити, розглянуті в прикладах 1–3. Для створення запитів з прикладів 2 і 3 додайте до таблиці Учні поле успішність.

Завдання 10.4

Реалізуйте такі запити в базі даних школа (до таблиці учнів додано поле успішність).

- Визначити найвищу успішність учнів 11А класу.
- Для кожного класу визначити, скільки математиків у ньому викладає.
- Визначити, у скількох учнів, молодших за 16 років, викладає Петрова Ніна Володимирівна. (Подумайте, яке обмеження потрібно накласти на дату народження учня.)

Групування в мові SQL*

Відобразіть запит, створений у завданні 10.1 (визначити дату народження наймолодшого учня в кожному класі), у режимі SQL. Його текст має бути таким.

```
SELECT Учні.клас, Max(Учні.[дата народження]) AS
    [Max-дата народження]
FROM Учні
GROUP BY Учні.клас;
```

Нове для вас ключове слово — GROUP BY, за допомогою якого, власне, і реалізують групування. Назви полів групування перелічують після цього слова, розділяючи комами, якщо таких полів кілька. Фразу GROUP BY записують після фрази WHERE або після фрази FROM, якщо фраза WHERE відсутня.

Звернімо увагу на особливості фрази SELECT запиту з групуванням.

- У цій фразі вказують хоча б одне поле групування. (Якщо жодного поля групування у фразі SELECT не вказати, у результаті запиту ми не побачимо груп і запит стане безглуздим.)
- Крім поля групування у фразі SELECT можна зазначити підсумкові функції, перелічені вище. Аргумент такої функції — назва поля підсумків, яку записують у дужках. Наприклад, у нашому запиті функція називається Max, а її аргумент — Учні.[дата народження].

3. Полю, де відображаються підсумки, можна надати назву, записавши її після ключового слова AS. Зазначимо, що за допомогою цього слова можна перейменовувати будь-які поля в таблиці результатів запиту.
4. Крім полів групування та підсумкових функцій у фразі SELECT запиту з групуванням неможна вказувати жодних інших полів. Щоб зрозуміти, чому так, припустимо, що в розглянутому щойно запиті ми відображуємо ще прізвище учня:

```
SELECT Учні.клас, Учні.прізвище,...
...
GROUP BY Учні.клас
```

Оскільки для кожної групи (у даному випадку — класу) в таблиці результатів відводиться один рядок (див. рис. 10.1, б), а в класі може бути кілька учнів, то їх прізвища просто не вмістяться в одній клітинці (нагадаємо, що кожна клітинка реляційної БД містить тільки одне значення). Саме тому така конструкція неприпустима.

Завдання 10.5°

Реалізуйте мовою SQL запит *визначити, скільки учнів навчається в кожному класі*. Для обчислення кількості використайте функцію Count. Її аргументом може бути * (символ, що позначає будь-яке поле), тому що, як зазначалося вище, неважливо, за яким полем рахувати кількість записів.

Тепер розглянемо SQL-текст запиту, створеного у завданні 10.2: визначити класи, всі учні яких народилися до 1 січня 1998 року.

```
SELECT Учні.клас
FROM Учні
GROUP BY Учні.клас
HAVING Max(Учні.[дата народження])<#1/1/1998#;
```

У ньому є нове для вас ключове слово HAVING. Після цього слова записують умову відбору груп записів, тобто фраза HAVING відіграє для груп записів ту саму роль, що і фраза WHERE для окремих записів. У цілому наведений запит СКБД виконує так:

1. Таблиця Учні групується за полем клас: до однієї групи відносяться записи учнів одного класу.
2. Для кожної групи перевіряється умова Max(Учні.[дата народження])<#1/1/1998#, тобто чи народився наймолодший учень до 1 січня 1998 року.
3. Значення поля клас тих груп, для яких умова виконується, відображується в таблиці результатів.

Завдання 10.6°

Реалізуйте мовою SQL запит *визначити прізвища вчителів, які викладають більше, ніж в одному класі*.

Звіти

Підіємо певні підсумки. У реляційній базі даних можна створювати запити двох різновидів: із групуванням та без нього. У запитах із групуванням усі записи в кожній групі об'єднуються в один, а результат запиту без групування — це таблиця, у якій відображено значення всіх полів усіх записів (рис. 10.6, а). Однак у деяких випадках для користувача незручна ні перша, ні друга форма подання інформації. Скажімо, у запиті без групування «Для кожного вчителя визначити прізвища та імена учнів, яких він навчає», результат виконання якого зображено на рис. 10.6, а, доречно було б виконати «м'яке» групування, відобразивши по одному разу прізвище, ім'я та по батькові кожного вчителя, а під ним — список усіх його учнів (рис. 10.6, б). У Microsoft Access подати дані в такий спосіб дозволяють *звіти* (рос. *Отчеты*). Звіти — це екранні форми, що призначені насамперед для подальшого друку на папері, і тому найчастіше вони подаються у вигляді аркушів паперу формату А4.

Учителі.прізвище	Учителі.ім'я	Учні.прізвище	Учні.ім'я
Сошко	Катерина	Шлак	Максим
Петрова	Ніна	Шлак	Максим
Сошко	Катерина	Хоменко	Ірина
Корбут	Василь	Хоменко	Ірина
Петрова	Ніна	Хоменко	Ірина
Михайлюк	Дмитро	Недригайло	Петро
Михайлюк	Дмитро	Буджак	Оксана
Корбут	Василь	Недригайло	Петро
Корбут	Василь	Буджак	Оксана
Петрова	Ніна	Недригайло	Петро
Петрова	Ніна	Буджак	Оксана

а

Учителі та учні

Учителі.прізвище	Учителі.ім'я	Учні.прізвище	Учні.ім'я
Корбут	Василь	Буджак	Оксана
		Недригайло	Петро
		Хоменко	Ірина
Михайлюк	Дмитро	Буджак	Оксана
		Недригайло	Петро
Петрова	Ніна	Буджак	Оксана
		Недригайло	Петро
		Хоменко	Ірина
		Шлак	Максим
Сошко	Катерина	Хоменко	Ірина
		Шлак	Максим

б

Рис. 10.6. Подання результатів запиту: а — в табличній формі; б — у формі звіту

В одному звіті може відображатися інформація з однієї або кількох таблиць або запитів. Створювати звіти, як і інші об'єкти БД, можна в режимі майстра або за допомогою конструктора. Зауважимо, що коли у звіті відображаються дані з кількох таблиць, краще спочатку створити запит, де ці таблиці з'єднуються, а вже потім, на основі цього запиту, — звіт.

Подібно до форм, звіти найзручніше створювати в режимі майстра. Розглянемо детально всі кроки майстра на прикладі звіту за таким запитом.

Для кожного вчителя визначити класи, у яких він викладає, а також прізвища, імена та середню успішність учнів цих класів.

Попередній крок

Створюємо запит, умову якого ми щойно сформулювали. Вікно конструктора запиту показано на рис. 10.7. Зауважте, що прямий зв'язок між таблицею класів та вчителів видалено, оскільки він визначає, хто є класним керівником, а за умовою запиту нам така інформація не потрібна. Зберігаємо запит під назвою Учителі_класи_учні.

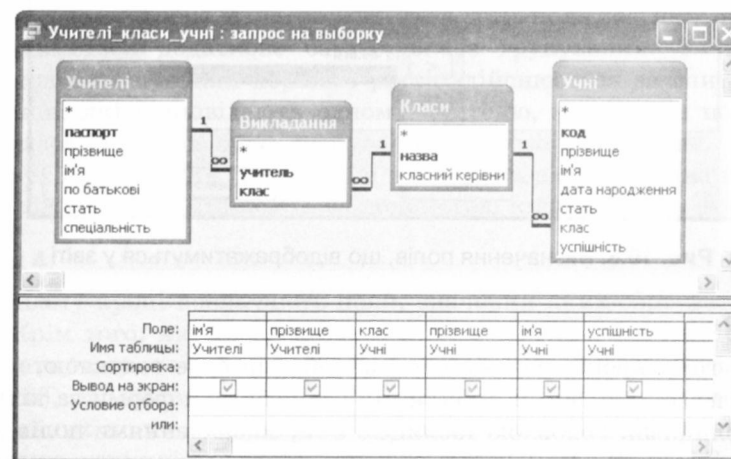


Рис. 10.7. Вікно конструктора запиту «Для кожного вчителя визначити класи, у яких він викладає, а також прізвища, імена та середню успішність учнів цих класів»

Крок 1

Запускаємо майстер звітів, клацнувши кнопку **Создание отчета** с помощью мастера (Створення звіту за допомогою майстра) на вкладці **Отчеты** головного вікна бази даних. Перший крок майстра звітів такий самий, як і майстрів форм та запитів. На ньому потрібно вибрати поля, дані з яких відобразяться у звіті. У нашому випадку зі списку **Таблицы и запросы** слід вибрати запит **Учителі_класи_учні**, а потім за допомогою кнопки **»** — усі його поля (рис. 10.8).

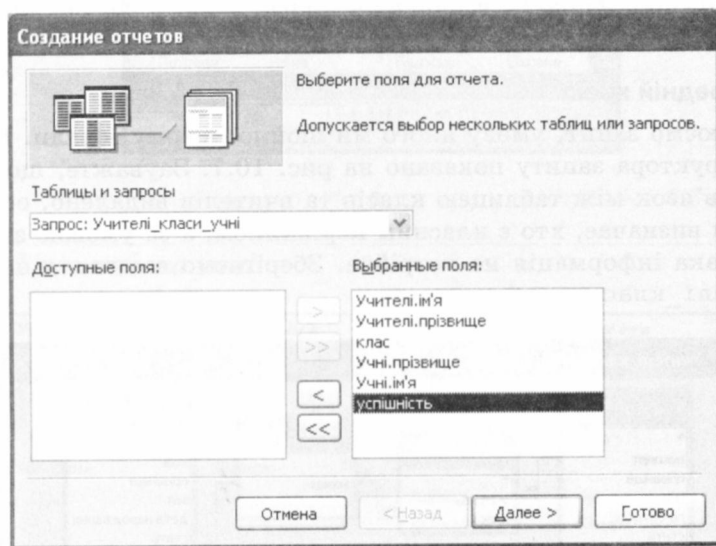


Рис. 10.8. Визначення полів, що відобразяться у звіті

Крок 2

На другому кроці визначаємо, яка таблиця буде головною, а яка — підлеглою, як ми це робили під час побудови форми за кількома таблицями. Головною таблицею є та, за значеннями полів якої проводиться групування, а підлеглою — та, значення полів якої відобразяться в групах. У нашому звіті головною буде таблиця вчителів, а підлеглою — таблиця учнів, оскільки для кожного вчителя ми виділятимемо групу учнів (рис. 10.9).

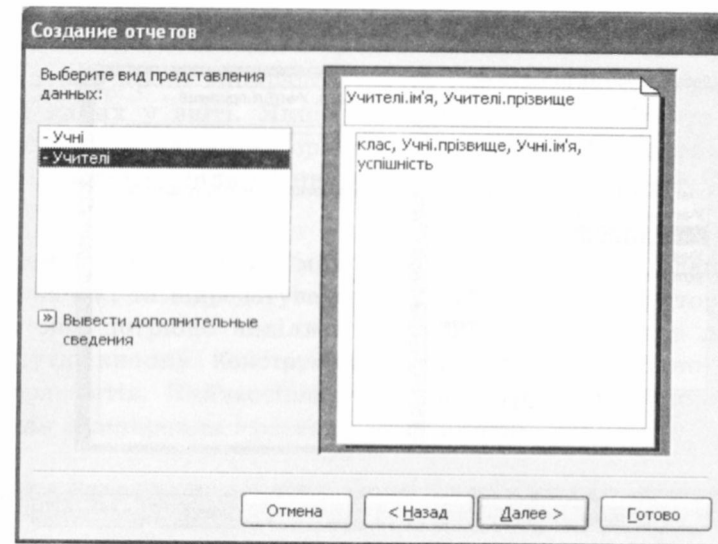


Рис. 10.9. Визначення головної та підлеглої таблиць

Крок 3

На третьому кроці майстра можна задати поля, за якими виконуватиметься додаткове багаторівневе групування. У нашому прикладі групування першого рівня здійснюється за вчителями, а учнів, які відповідають одному вчителю, можна ще поділити за класами — це буде групування другого рівня (рис. 10.10). Додаткові поля для групування вибирають за допомогою кнопки **>**, а рівень групи змінюють за допомогою кнопок **+** та **-**.

Крок 4

На цьому кроці визначають поля, значення яких сортуватимуться. Крім того, якщо серед полів звіту є числові, грошові або поля дати/часу, то у вікні майстра з'являється кнопка **Итоги** (Підсумки), за допомогою якої можна задати підсумкову операцію для таких полів. Наприклад, у нашому запиті можна обчислити середню успішність учнів кожного класу (рис. 10.11). Залежно від того, який перемикач встановлено у групі **Показать**, відобразатимуться або тільки результати підсумкових операцій (**только итоги**), або і підсумкові значення, і самі дані (**данные и итоги**).

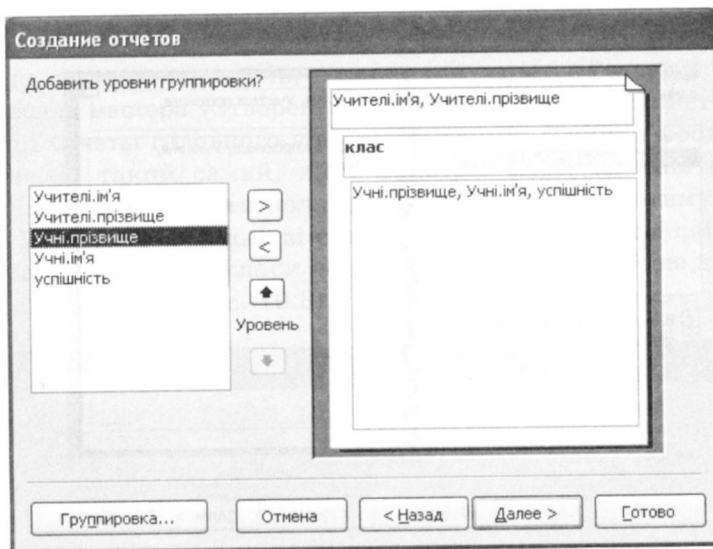


Рис. 10.10. Додаткове групування у звітах

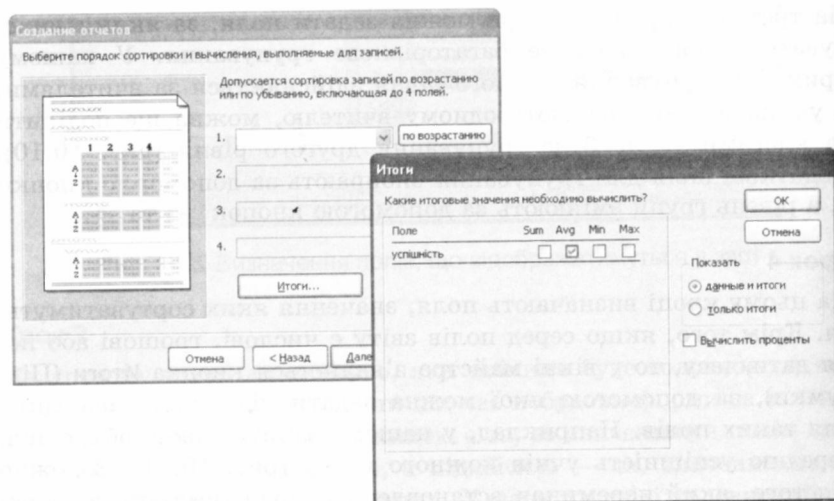


Рис. 10.11. Виконання підсумкових операцій у звітах

Кроки 5–7

На п'ятому кроці вибирають спосіб розташування та вирівнювання даних у звіті. Якщо має відобразитися багато полів, краще обрати альбомну орієнтацію аркуша. На шостому кроці слід задати стиль оформлення звіту, а на сьомому — його ім'я.

Готовий звіт (рис. 10.12) можна переглянути, двічі клацнувши його значок, та відредагувати за допомогою конструктора. Для цього звіт потрібно виділити в головному вікні бази даних і клацнути кнопку **Конструктор**, щоб перейти у вікно конструктора звітів. Найчастіше режим конструктора використовують для розширення і редагування написів.



Ім'я учителя	Прізвище учителя	Клас	Прізвище учня	Ім'я учня	Успішність
Василь	Корбут	11А			
			Хоменко	Ірина	10,6
Середня успішність класу					10,60
		11Б			
			Буджак	Оксана	11
			Недригайло	Петро	9
Середня успішність класу					10
Середня успішність в усіх учнів учителя					10,2

Рис. 10.12. Визначення способу розташування й вирівнювання даних у звіті

Завдання 10.7

Сконструйте звіт за запитом для кожного вчителя визначити класи, у яких він викладає, та прізвища й імена учнів, яких

він навчає, виконавши кроки, описані вище. Відобразіть також середню успішність учнів кожного класу.

Для допитливих. Якщо ви побудували звіт на основі якогось запиту, а потім цей запит змінили, на звіті ці зміни ніяк не відіб'ються. Щоб дійсно змінити набір даних, які відображаються у звіті, потрібно клацнути правою кнопкою миші індикатор  в лівому верхньому куті вікна конструктора звіту, вибрати з контекстного меню команду **Свойства** і на вкладці **Данные** вікна властивостей звіту клацнути кнопку , розташовану справа від поля **Источник записей**. У результаті ви перейдете знайоме вікно конструктора запиту. Але це буде той запит, який справді пов'язаний зі звітом (в області запитів головного вікна бази даних його немає).

Завдання 10.8

Створіть звіт, у якому для кожного класу відобразатимуться прізвища, імена та спеціальності вчителів, що в ньому викладають, а також кількість таких учителів.

Висновки

- ◆ Група — це набір записів з однаковим значенням певного поля. Під групуванням розуміють процес об'єднання групи записів в один; поле, значення якого однакові, називається полем групування.
- ◆ Підсумкова операція полягає в обчисленні в кожній групі записів певної підсумкової характеристики, наприклад, максимуму, суми, середнього. Підсумкову операцію виконують над значеннями якогось поля, що називається полем підсумків.
- ◆ Поле підсумків повинно мати числовий тип, грошовий тип або тип дати/часу і бути не тим полем, за яким здійснювалося групування.
- ◆ Операцію визначення кількості записів можна виконувати над значеннями будь-якого поля; її результат від вибору поля не залежить.

- ◆ Загальний синтаксис SQL-запиту з групуванням такий:

SELECT список полів, значення яких потрібно отримати

FROM список таблиць, з яких вибираються дані

WHERE умова, яку мають задовольняти записи, що вибираються

GROUP BY список полів групування

HAVING умова, яку мають задовольняти групи записів

- ◆ Звіти — це екранні форми, що подаються у вигляді аркушів паперу і призначені насамперед для подальшого друку. За допомогою звітів можна візуально групувати записи, не об'єднуючи їх.

Завдання для самостійного виконання

1. Реалізуйте такі запити.
 - а) Визначити назви класів, у яких навчається більше 2 учнів.
 - б) Визначити прізвища вчителів, які викладають більш, ніж в одному класі.
 - в) Визначити дату народження наймолодшого учня.
 - г) Визначити кількість учителів, що є класними керівниками.
 - д) Для кожного вчителя визначити максимальну успішність його учнів (використайте поле *успішність*).
 - е*) Визначити прізвище наймолодшого учня.
 - є*) Визначити спеціальності, за якими працюють тільки вчителі-жінки.
2. Створіть звіти за описаними далі запитами. Передбачте групування, де це доречно.
 - а) Для кожного класу визначити прізвища та імена учнів, що в ньому навчаються, а також їх найвищу та найнижчу успішність.
 - б) Для кожної спеціальності визначити вчителів, що її мають, кількість таких учителів та класи, у яких вони викладають.

Питання для роздумів

1. Сформулюйте запити, реалізацію яких показано на рис. 10.13, а-в.
- 2*. Запишіть SQL-текст запитів, вікна конструктора яких зображено на рис. 10.13, а-в.

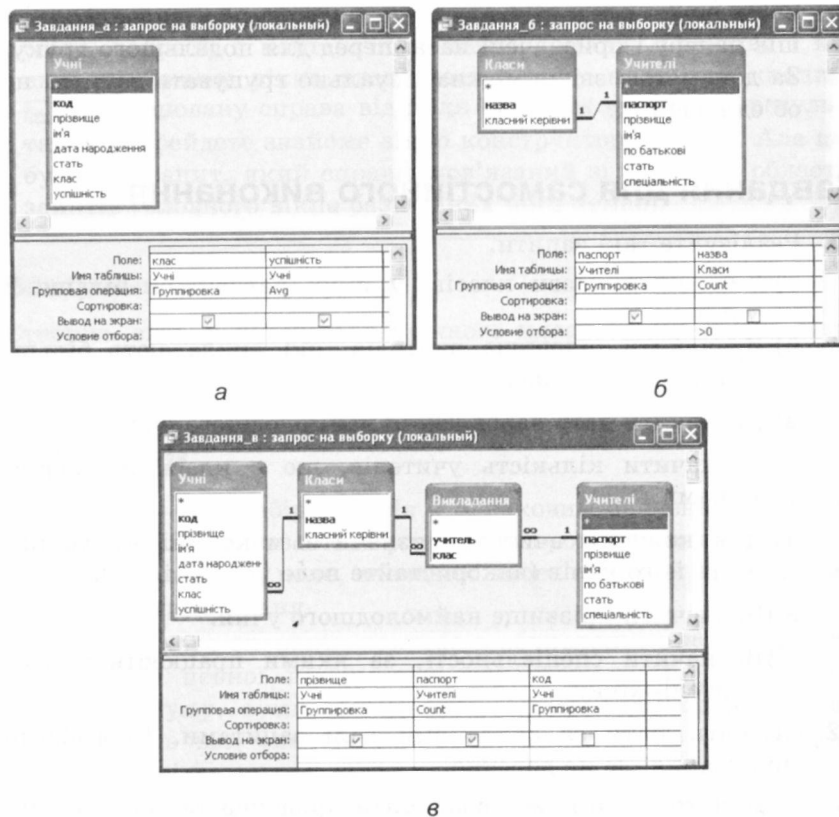


Рис. 10.13. Вікна конструктора запитів

- 3*. Прочитайте уважно коментар до прикладу 3, розглянутого в цьому розділі. Зауважте, що фактично групування даних в цьому запиті не відбувається, хоча й виконується підсум-

кова операція. Спробуйте реалізувати запит з прикладу 3 мовою SQL, використавши тільки фрази SELECT, FROM і WHERE. У фразі SELECT слід застосувати агрегатну функцію.

- 4*. Чому у запиті, розглянутому в прикладі 1, полями групування є паспорт і прізвище, а не тільки прізвище? Наведіть приклад даних, коли групування лише за полем прізвище давало б некоректний результат.
- 5*. Припустимо, що до бази даних **школа** додано таблицю Предмет, яку з'єднано з таблицями вчителів і класів через таблицю Викладання. Чому для такої бази реалізація запиту «для кожного вчителя визначити, скількох учнів він навчає», зображена на рис. 10.3, буде некоректною?

Завдання для досліджень

1. З'ясуйте, для чого призначений перехресний запит і як його конструювати. Додайте до бази даних **школа** таблицю Предмет з полями назва та кількість уроків на тиждень і з'єднайте її зв'язком з таблицею Викладання (визначте самостійно множинність зв'язку та створіть зовнішній ключ, потрібний для його моделювання). Побудуйте перехресний запит *визначити, скільки предметів викладає кожен учитель кожному учневі*.
2. До звіту, створеного в завданні для самостійного виконання 2б з цього розділу, додайте поле, де відобразатиметься середня успішність учнів, що навчаються у вчителів кожної спеціальності.