

Урок 4

План заняття:

- 1. Функції агрегування. Необхідність використання
- 2. Групування даних. Оператори GROUP BY та HAVING
- 3. Домашне завдання

1. Функції агрегування. Необхідність використання

В SQL, крім звичайних функцій, існує ряд спеціальних функцій, які називають функціями агрегування або групування. Кожна з цих функцій працює з сукупністю значень поля деякої таблиці, а результатом функцій є єдине значення. Саме від принципу їх роботи і походить назва цих функцій. Прикладів необхідності їх використання можна привести безліч. Наприклад, необхідно вивести суму продажу товарів на кожну дату, загальну кількість проданих будинків, розміщених в окремих районах або ж необхідно визначити мінімальну і максимальну ціни на товари окремої категорії.

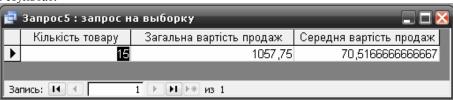
Функції агрегації (агрегування) переважно повертають числові результати і зазвичай використовуються для розрахунку певної статистики. В кожній СУБД підтримується свій набір таких функцій, але ряд з них затверджені стандартом SQL та спільні для всіх. Отже, до функцій агрегування належать:

- **COUNT** кількість значень (записів) в полі;
- **SUM** сума значень поля;
- **AVG** середнє арифметичне значень поля;
- MAX максимальне значення з сукупності;
- MIN мінімальне значення в сукупності;
- *StDev середньоквадратичного відхилення значень поля;
- *Var дисперсія роз приділених значень, які містяться у вказаному полі;
- *First перший запис з результуючого набору по вказаному полю;
- *Last останній запис з результуючого набору по вказаному полю.
 - * підтримуються лише Microsoft Access.

Варто також відмітити, що для функцій SUM і AVG поле ОБОВ'ЯЗКОВО повинно містити числові значення, в інших випадках поле можу містити дані довільного типу. Попрацюємо та розберемо кілька прикладів:

. Створимо запит на отримання інформації про загальну кількість товарів, їх загальну та середню вартість.

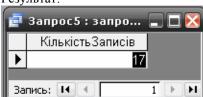
Результат:



2. Ніби нічого складного, але розглянемо маленький нюанс. Напишемо запит, що дозволить підрахувати лише загальну кількість записів в таблиці, що містить інформацію про товари:

SELECT COUNT(*) as КількістьЗаписів FROM Product;

Результат:



Функція COUNT з зірочкою в якості параметра включає при підрахунку як NULL-значення, так і значення, що повторюються. Припустимо, виникла ситуація, коли необхідно вивести список товарів з унікальними назвами (цілком

КОМПЬЮТЕРНАЯ АКАДЕМИЯ «ШАГ»



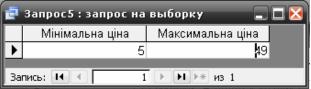
імовірна ситуація, коли товар з назвою «Банани» поставляється кількома постачальниками і виробляється кількома виробниками). Для здійснення таких підрахунків згідно стандарту SQL в тілі функцій агрегування, зокрема і COUNT можна використовувати ключове слово **DISTINCT** (різний) з метою підрахунку лише унікальних ненулевих значень, а також ключове слово **ALL** (всі) для врахування при підрахунку всіх значень поля, крім NULL. Доречі, ALL неявно підставляється по замовчуванню. В такому разі наш запит перепишеться:

```
SELECT COUNT(DISTINCT Name) as [Кількість унікальних назв товарів] FROM Product;
```

Але, СУБД MS Access не в повній мірі підтримує стандарт ANSI SQL, і тому даний запит викличе помилку. Доречі, явне використання ключового слова ALL в будь-якій функції агрегування також в MS Access викличе помилку.

3. Вивести на екран мінімальну та максимальну ціни на товари категорії «Кондитерські»:





2. Групування даних. Оператори GROUP BY та HAVING

Робота з функціями агрегування відрізняється від звичайної вибірки, оскільки їх результатом ϵ одне ϵ дине значення. В зв'язку з цим при створенні запиту на вибірку неможна одночасно при операторі SELECT використовувати звичайні поля та функції агрегування.

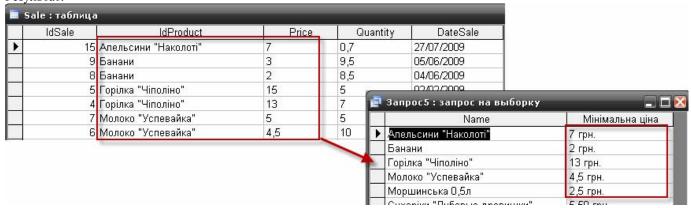
Для реалізації такої можливості використовується поняття групування даних та оператор, що його здійснює - **GROUP BY**. Вказана конструкція дозволяє визначати групи, які володіють спільними характеристиками, а потім генерувати статистику для кожної групи (з використанням нобхідної функції агрегування). В результаті СУБД буде повертати стільки рядків, скільки груп ви визначите.

Щоб краще розібратись з роботою даного оператора розглянемо кілька прикладів:

1. Вивести назви товарів та найменші ціни їх продажу:

```
SELECT Product.Name, MIN(Sale.Price) &' грн.' as [Мінімальна ціна]
FROM Product, Sale
WHERE Product.IdProduct=Sale.IdProduct
GROUP BY Product.Name;
```

Результат:



Поле «Name» з таблиці «Product» в списку SELECT являється полем групування. СУБД групує записи по значенню в даному полі, а потім шукає мінімальне значення ціни для кожної окремої групи, а потім виводиться результат.

При роботі з оператором GROUP BY слід пам'ятати про те, що всі поля групування, які ви задаєте в списку SELECT, повинні бути включені і в список GROUP BY. Якщо полів кілька, тоїх назви розділяються комами.

КОМПЬЮТЕРНАЯ АКАДЕМИЯ «ШАГ»



 Вивести на екран категорії продуктів, виробників товарів цих категорій та значення середньої ціни в розрізі цих груп (категорій та виробників). Відсортувати вибірку по назві виробника.

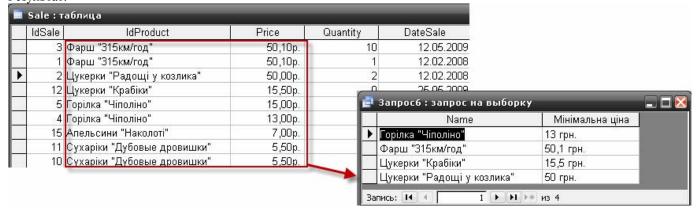
Результат:

📴 Запрос5 : запрос на выборку			
	Виробник	Категорія	Середня ціна
I	Banana Republica	Фрукти	9 грн.
	АТ "Русская водка"	Лікеро-горілчані	66,5 грн.
	ВАТ "Конфеті"	Кондитерські	49 грн.
	ВАТ "Росинка"	Бакалія	3 грн.
Г	ВАТ "Сальце України"	Ковбасні	54 грн.
	ВАТ "Сальце України"	Мясні	50 грн.
-	3AT "ППС"	Macui	42 rnu

3. А тепер виведемо інформацію про товари, мінімальна ціна продажу яких була більше 10 грн.:

```
SELECT Product.Name, MIN(Sale.Price)&' грн.' as [Мінімальна ціна]
FROM Product, Sale
WHERE Product.IdProduct=Sale.IdProduct
GROUP BY Product.Name
HAVING MIN(Sale.Price) > 10;
```

Результат:



Як Ви помітили, в попередньому запиті використовувався новий оператор **HAVING**. Він дозволяє визначити критерій, згідно якого певні групи не включаються в результуючу вибірку, аналогічно тому, як це робить оператор WHERE для окремих записів. Фактично, оператор HAVING дозволяє накласти умову на групи, тому він розміщується після оператора GROUP BY і вказується лише тоді, коли існує останній.

По суті, умову можна визначати і в операторі WHERE, вказуючи на критерій відбору кожного окремого запису, який входить до складу груп, але це робити недоречно. По-перше, падає швидкодія; по-друге, вказуючи умову в операторі WHERE, ви одразу виключаєте відібрані рядки з розгляду будь-якої операції групування. Оператор HAVING, навпаки, починає діяти лише після того, як групи сформовані, визначаючи, які саме групи будуть відображені в результаті.

Як правило, оператор HAVING містить функції агрегування, які вказані в списку SELECT, але це необов'язково.

4. Вивести кількість товарів кожної категорії, середня ціна поставки яких більше 20 грн.

```
SELECT Category.Name as [Категорія], COUNT(Product.IdProduct) as [Кількість]
FROM Product, Delivery, Category
WHERE Product.IdCategory=Category.IdCategory AND Product.IdProduct=Delivery.IdProduct
GROUP BY Category.Name
HAVING AVG(Delivery.Price) > 20;
```

КОМПЬЮТЕРНАЯ АКАДЕМИЯ «ШАГ»



5. Вивести категорії, товари, що до них належать та загальну суму їх продажу. Умова: товари лише категорій «Кондитерські» та «Фрукти».

3. Домашне завдання

- 1. Вивести загальну кількість товарів, максимальну та середню ціну їх продажу.
- 2. Вивести на екран кількість нових товарів в магазині, тобто товарів, які були поставлені на протязі останнього тижня (7 днів).
- 3. Підрахувати загальну кількість товарів двох виробників, наприклад ПП Іванов і ПП Воробей.
- 4. Вивести інформацію про найменшу вартість поставки для кожного постачальника товарів за останній місяць (для отримання інформації про останній місяць скористайтесь стандартними функціями) в євро. Відсортувати вибірку по назві постачальника в зростаючому порядку.
- 5. Підрахувати та вивести інформацію про загальну кількість продажу на кожен день в межах 01/01/2009 по поточний дань і вивести їх в спадаючому порядку кількості продаж.
- Вивести назви товарів виробника ПП Іванов, що поставлялись більш, ніж двума постачальниками.
- 7. Вивести інформацію про виробників, країни їх розташування, кількість товарів їх виробництва, які наявні в магазині. Умова: загальна вартість продажу товарів кожного виробника повинна бути в межах від 100 до 200 грн. Відсортувати вибірку по країні розташування в зростаючому порядку та кількості товарів виробників в спадаючому.
- 8. Вирахувати та вивести кількість товарів, кожної категорії, які необхідно списати. До списання підлягають товари, які наявні в магазині і не продавались з дати їх поставки на протязі 3 місяців.
- 9. Показати на экран кількість наявного товару по кожній категорії, при цьому враховувати лише товари, вартість поставки яких перевищувала 300 грн. Виведена інформація повинна стосуватись лише трьох постачальників: ТзОВ «Титанік», ПП Ваня, ВАТ «Швидкий вітер».
- 10. Показати категорію, товарів якої в магазині знаходиться найменьше.

Copyright © 2009 Iryn Petrova.