

Практическая работа №7

Формирование запросов с вычислениями. (4)

Цель занятия: Научиться создавать SQL запросы с вычисляемыми полями.

Теоретическая часть

Построение вычисляемых полей

В общем случае для создания вычисляемого (производного) поля в списке SELECT следует указать некоторое выражение языка SQL. В этих выражениях применяются арифметические операции сложения, вычитания, умножения и деления, а также встроенные функции языка SQL. Можно указать имя любого столбца (поля) таблицы или запроса, но использовать имя столбца только той таблицы или запроса, которые указаны в списке предложения FROM соответствующей инструкции. При построении сложных выражений могут понадобиться скобки.

Стандарты SQL позволяют явным образом задавать имена столбцов результирующей таблицы, для чего применяется фраза AS.

Пример 1. Рассчитать общую стоимость для каждой сделки. Этот запрос использует расчет результирующих столбцов на основе арифметических выражений.

```
SELECT Название, Цена, Количество, Цена*Количество AS Стоимость  
FROM Товар
```

Пример 2. Получить список фирм с указанием фамилии и инициалов клиентов.

```
SELECT Фирма, Фамилия+" "+ Left(Имя,1)+"."+Left(Отчество,1)+"." AS ФИО  
FROM Клиент
```

В запросе использована встроенная функция Left, позволяющая вырезать в текстовой переменной один символ слева в данном случае.

Пример 3. Получить список товаров с указанием года и месяца продажи.

```
SELECT Название, Year(Дата) AS Год, Month(Дата) AS Месяц  
FROM Товар
```

Использование итоговых (агрегатных) функций

С помощью итоговых (агрегатных) функций в рамках SQL-запроса можно получить ряд обобщающих статистических сведений о множестве отобранных значений выходного набора.

Пользователю доступны следующие основные итоговые функции:

- **Count(Выражение)** - определяет количество записей в выходном наборе SQL-запроса;
- **Min/Max(Выражение)** - определяют наименьшее и наибольшее из множества значений в некотором поле запроса;
- **Avg(Выражение)** - эта функция позволяет рассчитать среднее значение множества значений, хранящихся в определенном поле отобранных запросом записей. Оно является арифметическим средним значением, т.е. суммой значений, деленной на их количество.
- **Sum(Выражение)** - вычисляет сумму множества значений, содержащихся в определенном поле отобранных запросом записей.

Чаще всего в качестве выражения выступают имена столбцов. Выражение может вычисляться и по значениям нескольких таблиц.

Все эти функции оперируют со значениями в единственном столбце таблицы или с арифметическим выражением и возвращают единственное значение. Функции COUNT, MIN и MAX применимы как к числовым, так и к нечисловым полям, тогда как функции SUM и AVG могут использоваться только в случае числовых полей, за исключением COUNT(*). При вычислении результатов любых функций сначала исключаются все пустые значения, после чего требуемая операция применяется только к оставшимся конкретным значениям столбца.

Очень важно отметить, что итоговые функции могут использоваться только в списке SELECT и в составе предложения HAVING. Во всех других случаях это недопустимо. Если список полей в **SELECT** содержит итоговые функции, а в тексте запроса отсутствует фраза **GROUP BY**,

обеспечивающая объединение данных в группы, то ни один из элементов списка **SELECT** не может включать каких-либо ссылок на поля, за исключением ситуации, когда поля выступают в качестве аргументов итоговых функций.

Пример 4. Определить первое по алфавиту название товара.

SELECT Min(Название) AS Min_Название FROM Товар

Пример 5. Определить первое и последнее по алфавиту название товара.

SELECT Min(Название) AS Min_Название, Max(Название) AS Max_ FROM Товар

Пример 6. Определить количество сделок.

SELECT Count(*) AS Количество_сделок FROM Сделка

Пример 7. Определить суммарное количество проданного товара.

SELECT Sum(Количество) AS Количество_товара FROM Сделка

Пример 8. Определить среднюю цену проданного товара.

SELECT Avg(Цена) AS Средн_Цена FROM Товар

Пример 9. Подсчитать общую стоимость проданных товаров.

SELECT Sum(Цена*Количество) AS Стоимость FROM Товар

Параметр GROUP BY

Часто в запросах требуется формировать промежуточные итоги, по группам. Для этой цели в операторе **SELECT** используется параметр **GROUP BY**. Запрос, в котором присутствует **GROUP BY**, называется группирующим запросом, поскольку в нем группируются данные, полученные в результате выполнения операции **SELECT**, после чего для каждой отдельной группы создается единственная суммарная строка.

Все имена полей, приведенные в списке **SELECT**, должны присутствовать и во фразе **GROUP BY** - за исключением случаев, когда имя столбца используется в итоговой функции.

Если совместно с **GROUP BY** используется предложение **WHERE**, то оно обрабатывается первым, а группированию подвергаются только те строки, которые удовлетворяют условию поиска.

Пример 10. Вычислить средний объем покупок, совершенных каждым покупателем.

**SELECT Фамилия, Avg(Количество) AS Среднее_количество
FROM Клиент
GROUP BY Фамилия**

Пример 11. Определить, на какую сумму был продан товар каждого наименования.

**SELECT Название, Sum(Цена*Количество) AS Стоимость
FROM Сделка
GROUP BY Название**

Пример 12. Подсчитать общее количество купленного для каждой фирмы товара и его стоимость.

**SELECT Фирма, Sum(Количество) AS Общее_Количество,
Sum(Цена*Количество) AS Стоимость
FROM Сделка
GROUP BY Фирма**

Параметр HAVING

При помощи **HAVING** отражаются все предварительно сгруппированные посредством **GROUP BY** блоки данных, удовлетворяющие заданным в **HAVING** условиям. Это дополнительная возможность "профильтровать" выходной набор.

Условия в **HAVING** отличаются от условий в **WHERE**:

- **HAVING** исключает из результирующего набора данных группы с результатами агрегированных значений;
- **WHERE** исключает из расчета агрегатных значений по группировке записи, не удовлетворяющие условию;
- в условии поиска **WHERE** нельзя задавать агрегатные функции.

Пример 13. Определить фирмы, у которых общее количество сделок превысило три.

```
SELECT Фирма, Count(Количество) AS Количество_сделок
FROM Сделка
GROUP BY Фирма
HAVING Count(Количество)>3
```

Пример 14. Вывести список товаров, проданных на сумму более 10000 руб.

```
SELECT Название, Sum(Цена*Количество) AS Стоимость
FROM Товар
GROUP BY Название
HAVING Sum(Цена*Количество)>10000
```

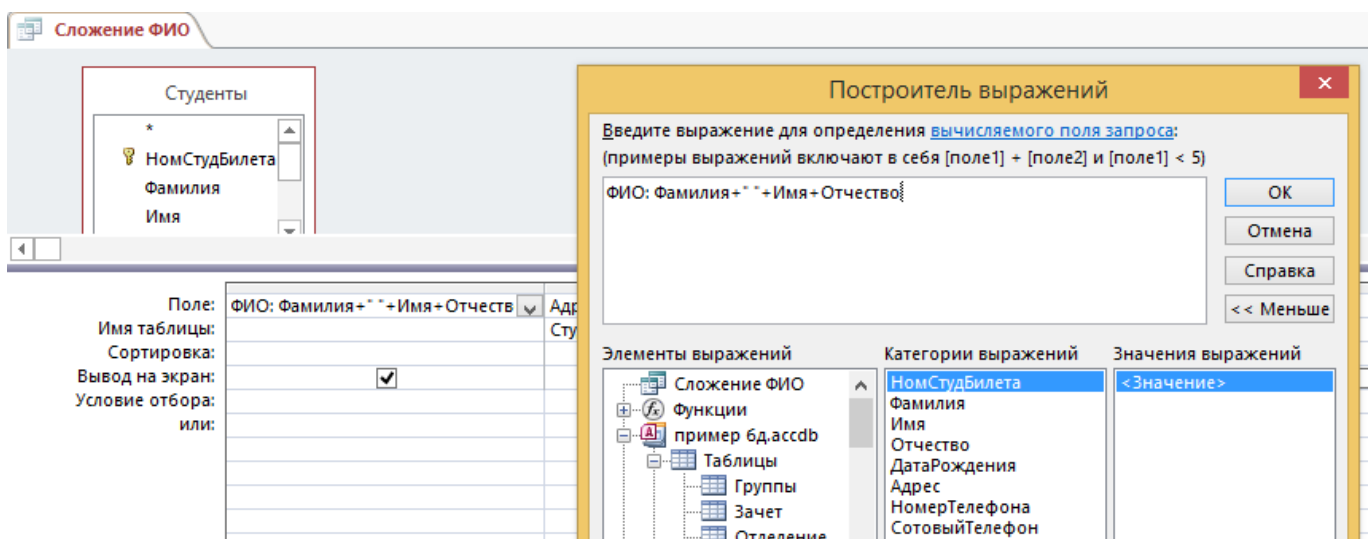
Пример 15. Вывести список товаров, проданных на сумму более 10000 без указания суммы.

```
SELECT Название
FROM Товар
GROUP BY Название
HAVING Sum(Цена*Количество)>10000
```

Создание запроса с вычисляемыми полями с помощью конструктора

Создания запроса с вычисляемыми полями осуществляется также как и обычного запроса и выполняется следующим образом:

1. На ленте **Создание**, в разделе **Запросы** нажмите кнопку **Конструктор запросов**.
2. Добавьте таблицы, которые будут использоваться в запросе.
3. Двойным щелчком мыши выберите из таблицы поля, которые будут отображаться при выполнении запроса.
4. В строке **Условия отбора** укажите условие для нужного поля. Отображаются только записи, которые соответствуют всем условиям в строке **Условия отбора**. Чтобы указать альтернативные условия для отдельного поля, введите первое условие в строке **Условия отбора**, второе условие в строке **или** и т. д.
5. Для создания вычисляемого поля щелкните в пустом столбце в строке «Поле», сформируйте вычисляемое поле по следующему шаблону: **Имя поля: Выражение**.
6. Сохраните полученный запрос и дайте ему имя.

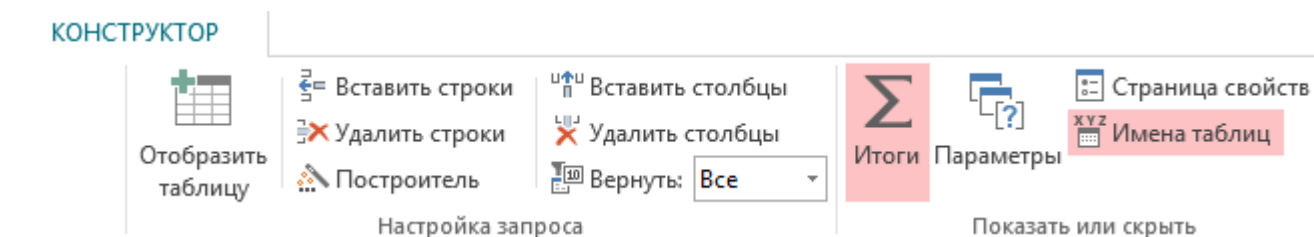


Для удобства создания вычисляемого поля можно использовать **Построитель выражений**, в котором можно либо вручную написать выражение, либо построить выражение, выбирая элементы из списка.

Создание итогового запроса с помощью конструктора

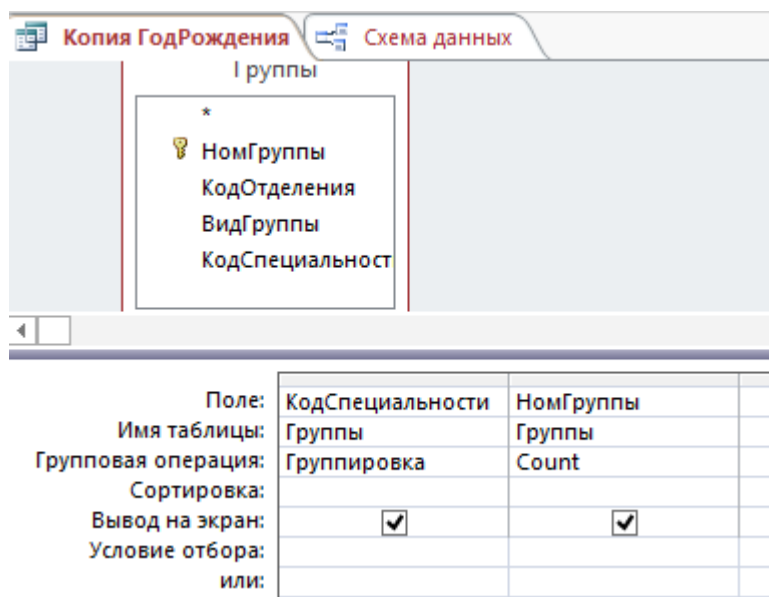
Создания запроса с итоговыми полями осуществляется также как и обычного запроса и выполняется следующим образом:

1. На ленте **Создание**, в разделе **Запросы** нажмите кнопку **Конструктор запросов**.
2. Добавьте таблицы, которые будут использоваться в запросе.
3. Двойным щелчком мыши выберите из таблицы поля, которые будут отображаться при выполнении запроса.
4. На вкладке **Конструктор** в группе **Показать или скрыть** нажмите кнопку **Итоги**.



Под последней строкой данных в таблице появится новая строка **Групповая операция**, а в строке для каждого поля запроса будет указано **Группировка**.

5. В строке **Групповая операция** щелкните поле, по которому вы хотите выполнить подсчет, и выберите в списке функцию, например **Count**.
6. Сохраните полученный запрос и дайте ему имя.



Практическое задание

1. Создайте копию файла БД предыдущей работы и дайте ему имя Работа№7.
2. Удалите элементы запросов из БД.
3. Подсчитайте количество предметов на каждой специальности. Сохраните запрос и полученный список в текстовый файл. Имя запроса – КолПредметовНаСпециал.
4. Сформируйте список: **ФИО студента** с инициалами и **Возраст**. Отсортируйте в обратном алфавитном порядке. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ВозрастСтудентов.

5. Для каждого предмета определить среднюю зачетную оценку. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ЗачетыСреднОценкаПредметов.
6. Сформируйте список студентов родившихся в текущем месяце, которым исполнится 18 лет. Сохраните запрос и полученный список в текстовый файл. Имя запроса – СовершеннолетиеСтудента.
7. Подсчитайте количество студентов в каждой группе. Сохраните запрос и полученный список в текстовый файл. Имя запроса – КолСтудентовВГруппе.
8. Для каждого студента определить среднюю оценку по каждому предмету на зачете. Сохраните запрос и полученный список в текстовый файл. Имя запроса – СредОценкиСтудента.
9. Сформируйте список совершеннолетних студентов. Сохраните запрос и полученный список в текстовый файл. Имя запроса – СовершеннолетниеСтуденты.
10. Подсчитайте количество часов, которые ведет каждый преподаватель. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ЧасыПреподавателей.
11. Сформируйте список студентов заданного года рождения. Сохраните запрос и полученный список в текстовый файл. Имя запроса – СтудентыГодРождения.
12. Оформите отчет.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какую фразу нельзя использовать без фразы GROUP BY?
2. Определите случаи, когда необходимо использовать группирование данных?
3. Какое поле должно обязательно присутствовать во фразе Select при группировании данных?