

Практическое задание 7

Язык SQL

Использование подзапросов, возвращающих единичное значение

Дана БД, моделирующая работу торговой фирмы со схемой БД(Товар, Клиент, Сделка)

Товар(ID_тов, Назв, Цена, Тип, сорт, гор_Тов)

Клиент(ID_кл, Фирма, Фам, Имя, Отч, год_Кл, Тел)

Сделка(ID_сд, ID_Тов, ID_Кл, Кол-во, Дата)

Склад(ID_тов, Кол-во)

Пример 9.1. Определить даты сделок, превысивших по количеству товара среднее значение и указать для этих сделок превышение над средним уровнем.

Пример 9.2. Определить клиентов, в сделках которых количество товара отличается от максимального не более чем на 10%.

Применение подзапроса в предложении **HAVING**.

Пример 9.3. Определить даты, когда среднее количество проданного за день товара оказалось больше среднего показателя по всем сделкам вообще.

Использование подзапросов, возвращающих множество значений

Во многих случаях значение, подлежащее сравнению в предложениях **WHERE** или **HAVING**, представляет собой не одно, а несколько значений. Вложенные подзапросы генерируют непоименованное промежуточное отношение, временную таблицу. Оно может использоваться только в том месте, где появляется в подзапросе. К такому отношению невозможно обратиться по имени из какого-либо другого места запроса. Применяемые к подзапросу операции основаны на тех операциях, которые, в свою очередь, применяются к множеству, а именно:

- { **WHERE** | **HAVING** } выражение [**NOT**] **IN** (подзапрос);
- { **WHERE** | **HAVING** } выражение оператор_сравнения { **ALL** | **SOME** | **ANY** }(подзапрос);
- { **WHERE** | **HAVING** } [**NOT**] **EXISTS** (подзапрос);

Пример 9.4. Определить список отсутствующих на складе товаров.

Пример 9.5. Определить товары, которые покупают клиенты из Москвы.

Введение в запрос фразы "только" требует использования операции **NOT IN**.

Пример 9.6. Определить товары, покупку которых осуществляют **только** клиенты из Москвы, и никто другой.

Пример 9.7. Какие товары ни разу не купили московские клиенты?

Пример 9.8. Определить фирмы, покупающие товары местного производства.

Использование ключевых слов **ANY** и **ALL**

Ключевые слова **ANY** и **ALL** могут использоваться с подзапросами, возвращающими один столбец чисел.

Если подзапросу будет предшествовать ключевое слово **ALL**, условие сравнения считается выполненным, только когда оно выполняется для всех значений в результирующем столбце подзапроса.

Если записи подзапроса предшествует ключевое слово **ANY**, то условие сравнения считается выполненным, когда оно выполняется хотя бы для одного из значений в результирующем столбце подзапроса.

Если в результате выполнения подзапроса получено пустое значение, то для ключевого слова **ALL** условие сравнения будет считаться выполненным, а для ключевого слова **ANY** – невыполненным. Ключевое слово **SOME** является синонимом слова **ANY**.

Пример 9.9. Найти фирму, которая приобрела товаров на самую большую сумму.

Использование операций EXISTS и NOT EXISTS

Ключевые слова *EXISTS* и *NOT EXISTS* предназначены для использования только совместно с *подзапросами*. Результат их обработки представляет собой логическое значение *TRUE* или *FALSE*. Для ключевого слова *EXISTS* результат равен *TRUE* в том и только в том случае, если в возвращаемой *подзапросом* результирующей таблице присутствует хотя бы одна строка. Если результирующая таблица *подзапроса* пуста, результатом обработки операции *EXISTS* будет значение *FALSE*. Для ключевого слова *NOT EXISTS* используются правила обработки, обратные по отношению к ключевому слову *EXISTS*. Поскольку по ключевым словам *EXISTS* и *NOT EXISTS* проверяется лишь наличие строк в результирующей таблице *подзапроса*, то эта таблица может содержать произвольное количество столбцов.

Пример 9.10. Определить список отсутствующих на складе товаров

Запрос добавления

Оператор *INSERT* применяется для *добавления записей* в таблицу. Формат оператора:

```
<оператор_вставки> ::= INSERT INTO <имя_таблицы>
    [(имя_столбца [, ...n])
    {VALUES (значение[, ...n])} |
    <SELECT_оператор>}
```

Пример 9.11. Добавить в итоговую таблицу сведения об общей сумме ежемесячных продаж каждого наименования товара.

Запрос удаления

Оператор *DELETE* предназначен для *удаления группы записей* из таблицы.

```
<оператор_удаления> ::= DELETE
    FROM <имя_таблицы> [WHERE <условие_отбора>]
```

Пример 9.12. Удалить все прошлогодние сделки.

В примере условие отбора формируется с учетом года (функция *Year*) от текущей даты (функция *GETDATE()*).

Запрос обновления

Оператор *UPDATE* применяется для *изменения значений* в группе записей или в одной записи указанной таблицы.

```
<оператор_изменения> ::=
    UPDATE имя_таблицы SET имя_столбца=
        <выражение> [, ...n]
    [WHERE <условие_отбора>]
```

Параметр *имя_таблицы* – это либо имя таблицы базы данных, либо имя обновляемого представления. В предложении *SET* указываются имена одного и более столбцов, данные в которых необходимо изменить. Предложение *WHERE* является необязательным. Если оно опущено, значения указанных столбцов будут изменены во всех строках таблицы. Если предложение *WHERE* присутствует, то обновлены будут только те строки, которые удовлетворяют условию отбора. Выражение представляет собой новое значение соответствующего столбца и должно быть совместимо с ним по типу данных.

Пример 9.13. Для товаров первого сорта установить цену в значение 140 и остаток – в значение 20 единиц.

Пример 9.14. Увеличить цену товаров первого сорта на 25%.

Пример 9.15. В сделке с максимальным количеством товара увеличить число товаров на 10%.

Домашнее задание

1. Дорешать задачи из занятия 9.

2. Придумать 4 запроса с использованием подзапросов (в электр. виде)