**Как работает запрос SQL со связанным подзапросом**

SELECT \*

FROM EXAM\_MARKS

WHERE STUDENT\_ID =

(SELECT STUDENT\_ID

FROM STUDENT

WHERE SURNAME = 'Петров');

Как работает запрос SQL со связанным подзапросом?

• Выбирается строка из таблицы, имя которой указано во внешнем запросе.

• Выполняется подзапрос и полученное значение применяется для анализа этой строки в условии предложения WHERE внешнего запроса.

• По результату оценки этого условия принимается решение о включении или не включении строки в состав выходных данных

• Процедура повторяется для следующей строки таблицы

внешнего запроса.

Следует обратить внимание, что приведенный выше запрос корректен только в том случае, если в результате выполнения указанного в скобках подзапроса возвращается единственное значение. Если в результате выполнения подзапроса будет возвращено несколько значений, то этот подзапрос будет

ошибочным. В данном примере это произойдет, если в таблице STUDENT будет несколько записей со значениями поля

SURNAME = 'Петров'.

В некоторых случаях для гарантии получения единственного значения в результате выполнения подзапроса используется

DISTINCT.

Одним из видов функций, которые автоматически

всегда выдают в результате единственное значение для любого

количества строк, являются агрегирующие функции.

Оператор IN также широко применяется в подзапросах. Он

задает список значений, с которыми сравниваются другие значения для определения истинности задаваемого этим оператором предиката.

Данные обо всех оценках (таблица EXAM\_MARKS) студентов

из Воронежа можно выбрать с помощью следующего запроса:

SELECT \*

FROM EXAM\_MARKS

WHERE STUDENT\_ID IN

(SELECT STUDENT\_ID

FROM STUDENT

WHERE CITY = 'Воронеж');

Подзапросы можно применять внутри предложения HAVING.

Пусть требуется определить количество предметов обучения

с оценкой, превышающей среднее значение оценки студента

с идентификатором 301:

SELECT COUNT(DISTINCT SUBJ\_ID), MARK

FROM EXAM\_MARKS

GROUP BY MARK

HAVING MARK >

(SELECT AVG(MARK)

FROM EXAM\_MARKS

WHERE STUDENT\_ID = 301);

При использовании подзапросов во внутреннем запросе

можно ссылаться на таблицу, имя которой указано в предложении FROM внешнего запроса. В этом случае такой связанный

подзапрос выполняется по одному разу для каждой строки таблицы основного запроса.

Пример: выбрать сведения обо всех предметах обучения, по

которым проводился экзамен 20 января 1999 г.

SELECT \*

FROM SUBJECT SU

WHERE '20/01/1999' IN

(SELECT EXAM\_DATE

PROM EXAM\_MARKS EX

WHERE SU.SUBJ\_ID = EX.SUBJ\_ID);

В некоторых СУБД для выполнения этого запроса может

потребоваться преобразование значения даты в символьный

тип. В приведенном запросе su и ЕХ являются псевдонимами

(алиасами), то есть специально вводимыми именами, которые

могут быть использованы в данном запросе вместо настоящих

имен. В приведенном примере они используются вместо имен

таблиц SUBJECT и EXAM\_MARKS.

Эту же задачу можно решить с помощью операции соединения таблиц:

SELECT DISTINCT SU.SUBJ\_ID, SUBJ\_NAME, HOUR, SEMESTER

FROM SUBJECT FIRST, EXAM\_MARKS SECOND

WHERE FIRST.SUBJ\_ID = SECOND.SUBJ\_ID

AND SECOND.EXAM\_DATE = '20/01/1999';

В этом выражении алиасами таблиц являются имена FIRST

И SECOND.

Можно использовать подзапросы, связывающие таблицу со

своей собственной копией.

Например, надо найти идентификаторы, фамилии и стипендии студентов, получающих стипендию выше средней на курсе, на котором они учатся.

SELECT DISTINCT STUDENT\_ID, SURNAME, STIPEND

FROM STUDENT El

WHERE STIPEND >

(SELECT AVG(STIPEND)

FROM STUDENT E2

WHERE El.KURS = E2.KURS);

Тот же результат можно получить с помощью следующего

запроса:

SELECT DISTINCT STUDENT\_ID, SURNAME, STIPEND

FROM STUDENT El,

(SELECT KURS, AVG (STIPEND) AS AVG\_STIPEND

FROM STUDENT E2

GROUP BY E2.KURS) E3

WHERE El.STIPEND > AVG\_STIPEND AND El.KURS=E3.KURS;

Обратите внимание — второй запрос будет выполнен гораздо

быстрее. Дело в том, что в первом варианте запроса агрегирующая функция AVG выполняется над таблицей, указанной в подзапросе, для каждой строки внешнего запроса.

В другом варианте вторая таблица (алиас Е2) обрабатывается агрегирующей

функцией один раз, в результате чего формируется вспомогательная таблица (в запросе она имеет алиас ЕЗ), со строками которой

затем соединяются строки первой таблицы (алиас Е1). Следует

иметь в виду, что реальное время выполнения запроса в большой

степени зависит от оптимизатора запросов конкретной СУБД.

3. Связанные подзапросы в HAVING

Предложение GROUP BY позволяет группировать выводимые SELECT-запросом записи по значению некоторого поля. Использование предложения HAVING позволяет при выводе осуществлять фильтрацию таких групп.

Предикат предложения HAVING оценивается не для каждой

строки результата, а для каждой группы выходных записей,

сформированной предложением GROUP BY внешнего запроса.

Пусть, например, необходимо по данным из таблицы

EXAM\_MARKS определить сумму полученных студентами оценок

(значений поля MARK), сгруппировав значения оценок по датам

экзаменов и исключив те дни, когда число студентов, сдававших в течение дня экзамены, было меньше 10.

SELECT EXAM\_DATE, SUM(MARK)

FROM EXAM\_MARKS A

GROUP BY EXAM\_DATE

HAVING 10 <

(SELECT COUNT(MARK)

FROM EXAM\_MARKS В

WHERE A.EXAM\_DATE = В.EXAM\_DATE);

Подзапрос вычисляет количество строк с одной и той же

датой, совпадающей с датой, для которой сформирована очередная группа основного запроса.