

Лабораторная работа № 14.

ТЕМА: «Выборки данных из нескольких таблиц»

ЦЕЛЬ: Научиться делать запросы из нескольких таблиц сразу.

Порядок выполнения работы:

1. Сделать объединённый запрос используя union;
2. Сделать объединённый запрос используя операции соединения join.

Теоретические сведения

Union

Для объединения запросов используется служебное слово **UNION**:

1. <запрос 1>
2. UNION [ALL]
3. <запрос 2>

Предложение **UNION** приводит к появлению в результирующем наборе всех строк каждого из запросов. При этом, если определен параметр **ALL**, то сохраняются все дубликаты выходных строк, в противном случае в результирующем наборе присутствуют только уникальные строки. Заметим, что можно связывать вместе любое число запросов. Кроме того, с помощью скобок можно задавать порядок объединения.

Операция объединения может быть выполнена только при выполнении следующих условий:

- количество выходных столбцов каждого из запросов должно быть одинаковым;
- выходные столбцы каждого из запросов должны быть совместимы между собой (в порядке их следования) по типам данных;
- в результирующем наборе используются имена столбцов, заданные в первом запросе;
- предложение **ORDER BY** применяется к результату соединения, поэтому оно может быть указано только в конце всего составного запроса.

Явные операции объединения

В предложении **FROM** может быть указана **явная операция соединения** двух и более таблиц. Среди ряда операций соединения, описанных в стандарте языка SQL, многими серверами баз данных поддерживается только операция **соединения по предикату**. Синтаксис соединения по предикату имеет вид:

1. FROM <таблица 1>
2. [INNER]
3. { {LEFT | RIGHT | FULL } [OUTER] } JOIN <таблица 2>
4. [ON <предикат>]

Соединение может быть либо внутренним (**INNER**), либо одним из внешних (**OUTER**). Служебные слова **INNER** и **OUTER** можно опускать, поскольку внешнее соединение однозначно определяется его типом — **LEFT** (левое), **RIGHT** (правое) или **FULL** (полное), а просто **JOIN** будет означать внутреннее соединение.

Предикат определяет условие соединения строк из разных таблиц. При этом **INNER JOIN** означает, что в результирующий набор попадут только те соединения строк двух таблиц, для которых значение предиката равно **TRUE**. Как правило, предикат определяет эквисоединение по внешнему и первичному ключам соединяемых таблиц, хотя это не обязательно.

Краткая информация о базе данных "Компьютерная фирма":

Схема БД состоит из четырех таблиц:

Product(maker, model, type)

PC(code, model, speed, ram, hd, cd, price)

Laptop(code, model, speed, ram, hd, price, screen)

Printer(code, model, color, type, price)

Таблица Product представляет производителя (maker), номер модели (model) и тип ('PC' - ПК, 'Laptop' - ПК-блокнот или 'Printer' - принтер). Предполагается, что номера моделей в таблице Product уникальны для всех производителей и типов продуктов. В таблице PC для каждого ПК, однозначно определяемого уникальным кодом – code, указаны модель – model (внешний ключ к таблице Product), скорость - speed (процессора в мегагерцах), объем памяти - ram (в мегабайтах), размер диска - hd (в гигабайтах), скорость считывающего устройства - cd (например, '4x') и цена - price. Таблица Laptop аналогична таблице PC за исключением того, что вместо скорости CD содержит размер экрана -screen (в дюймах). В таблице Printer для каждой модели принтера указывается, является ли он цветным - color ('y', если цветной), тип принтера - type (лазерный – 'Laser', струйный – 'Jet' или матричный – 'Matrix') и цена - price.

Задание:

- 1) Найдите производителей самых дешевых цветных принтеров. Вывести: maker, price
- 2) Перечислите номера моделей любых типов, имеющих самую высокую цену по всей имеющейся в базе данных продукции.

Ход работы:

1. Прочитать теорию.
2. Ознакомиться с базой данных "Компьютерная фирма".
3. Выполнить задание.

Контрольные вопросы:

1. Для чего используется оператор union?
2. Для чего используется оператор join?
3. Формат оператора соединения и оператора объединения?