

## Лабораторная работа № 7. Предикат NULL. Подзапросы. Предикат EXISTS

Цель работы: научиться применять в запросах предикаты NULL, EXISTS, ANY и ALL.

**Предикат IS NULL.** Для выяснения смысла значения NULL рассмотрим пример. Пусть в городе N ведётся база данных, в которой хранятся данные обо всех жителях, включая детей. Очевидно, что в графу «профессия» записи о ребёнке поместить нечего, так как у ребёнка ещё нет профессии. Графа профессия может оказаться пустой и в том случае, когда в момент занесения данных профессия жителя не была известна. Предполагается, что графа будет заполнена позже. Для неизвестного значения в SQL применяется специальное обозначение NULL. Значение NULL имеют по умолчанию все поля, в которые ничего не заносилось.

NULL применяется в полях всех типов и само не имеет типа. Значение NULL можно использовать только в специальном предикате IS NULL, имеющем следующий синтаксис:

<выражение> IS [NOT] NULL

Предикат IS NULL принимает значение «истина» только, если выражение равно NULL.

Для работы с NULL-значениями полей создайте в базе данных таблицу, состоящую из двух любых текстовых полей, например, «ФИО» и «Адрес».

Создайте и выполните следующие запросы к таблице:

- a) выбрать все записи с NULL;
- b) выбрать все записи с “”;
- c) выбрать все записи, в которых есть адреса;
- d) выбрать все записи, в которых нет адресов;

**Подзапросы.** С помощью SQL можно вкладывать один запросы внутри другого. Внутренний запрос называют подзапросом. Обычно, внутренний запрос генерирует значение, которое проверяется в предикате внешнего запроса, определяющего верно оно или нет. Например, в следующем запросе

выбираются из таблицы «Товары» те товары, цена которых меньше средней цены всех товаров таблицы:

```
SELECT *  
FROM Товары  
WHERE Цена < (SELECT AVG (Цена) FROM Товары) ;
```

Самостоятельно постройте подзапрос к одной из таблиц своей базы данных.

### **Предикат EXISTS имеет синтаксис**

EXISTS подзапрос

и принимает значение ИСТИНА (TRUE), если подзапрос содержит хотя бы одну строку.

В следующем запросе выбираются фамилии всех сотрудников, оформлявших заказы для клиента ANTON, при условии, что хотя бы один заказ для клиента ANTON был размещён в мае любого года.

```
SELECT DISTINCT b.Фамилия  
FROM Заказы a, Сотрудники b  
WHERE EXISTS (SELECT * FROM Заказы WHERE КодКлиента='ANTON'  
AND DatePart('m', ДатаРазмещения)=5)  
AND a.КодСотрудника  
= b.КодСотрудника AND a.КодКлиента='ANTON' ;
```

Самостоятельно, используя таблицы своей базы данных, создайте и выполните запрос с предикатом EXIST.

### **Объединение результатов нескольких запросов – UNION. Создание таблицы из существующих таблиц – SELECT ... INTO**

Цель работы: научиться объединять в одной выводимой таблице строки, полученные разными запросами и создавать новую таблицу базы данных из существующих таблиц.

**Предложение UNION** применяется для объединения результатов нескольких запросов в одной выводимой таблице. Количество столбцов во всех запросах должно быть одинаковым и типы соответствующих столбцов

должны быть сравнимыми. В следующем примере выводятся адреса и города клиентов и заказов. Параметр ALL разрешает выводить дубликаты строк.

```
SELECT ALL Адрес, Город, 'Заказы ' AS Источник
FROM Клиенты
UNION
SELECT ALL АдресПолучателя AS Адрес, ГородПолучателя AS Го
род, 'Клиенты ' AS Источник
FROM Заказы;
```

Самостоятельно выберите из двух любых таблиц как минимум по два столбца и поместите выдачу результирующего запроса в новую таблицу:

**Создание таблицы из существующих таблиц с помощью SELECT ... INTO.** Во многих СУБД конструкция SELECT ... INTO <имя таблицы> используется для создания новой таблицы и вывода в неё результатов запроса. Например, таблица «Страны» с названиями всех стран, в которые направляются заказы, создаётся в результате выполнения следующего запроса:

```
SELECT DISTINCT СтранаПолучателя
INTO Страны
FROM Заказы;
```

Самостоятельно с помощью SELECT ... INTO создайте новую таблицу, содержащую данные из любой таблицы обо всех записях определенного значения. Например, создайте таблицу «Клиенты2», содержащую данные из таблицы «Клиенты» обо всех клиентах, живущих в Томске.