**Лабораторная работа №4**

**Создание запросов с подзапросами.**

*Цель**работы*:

приобретение практических навыков по созданию  запросов с подзапросами на языке SQL.

***Краткие теоретические сведения***

Подзапросом называется запрос, который помещается в другой запрос в качестве источника данных.

Результатом выполнения запроса с инструкцией SELECT в общем случае является таблица. В случае, когда в инструкции SELECT список столбцов содержит один столбец, а условие отбора строк таково, что отбирается только одна строка, результирующая таблица содержит единственное данное. Такую инструкцию SELECT можно использовать в логическом выражении для отбора строк. В этом случае значения столбца в логическом выражении для отбора строк будут сравниваться с результатом, возвращаемым инструкцией SELECT. Первая инструкция SELECT запроса называется внешним запросом (outer query), а внутренняя инструкция SELECT, используемая в логическом выражении, называется вложенным запросом (inner query) или подзапросом. Первым выполняется вложенный запрос, а его результат передается внешнему запросу.

Подзапросы позволяют получать данные из множества таблиц.

Подзапрос может применяться со следующими операторами:

* операторами сравнения;
* оператором IN;
* операторами ANY и ALL;

**Подзапросы и операторы сравнения**

Использование оператора равенства (=) с подзапросом показано в примере 4.1.

Пример 4.1. Выборка имен и фамилий сотрудников склада (название отдела «Склад»)

**USE Shop;**

**SELECT LName as Фамилия, FName as Имя**

**FROM Employees**

**WHERE deptid =**

**(SELECT deptid FROM Depts WHERE Dept = 'Склад');**

Результат выполнения этого запроса:

|  |  |
| --- | --- |
| **Фамилия** | **Имя** |
| Федосеенко | Валерий |
| Мазуров | Владимир |

В примере 4.1 сначала выполняется подзапрос (вложенный запрос), который возвращает идентификатор отдела с названием «Склад». Это значение подставляется в логическое выражение для отбора строк.

С подзапросами можно также использовать любые другие операторы сравнения, при условии, что вложенный запрос возвращает в результате одну строку и один столбец.

**Подзапросы и оператор IN**

Вложенный запрос, возвращающий множество значений также может быть использован в подзапросе с использованием оператора IN.

Оператор IN возвращает значение true, если в поле, имя которого записано перед IN записано значение, входящее в множество значений, записанных в скобках после слова IN. Это множество значений может быть сформировано с помощью подзапроса, возвращающего несколько значений.

Использование оператора IN в подзапросе показано в примере 4.2.

Пример 4.2. Получение списка наименований товаров в покупке с идентификатором, равным 1

**USE Shop;**

**SELECT good as Товар**

**FROM Goods**

**WHERE goodid IN**

**(SELECT goodid FROM Sales WHERE saleid=1);**

Результат выполнения этого запроса:

|  |
| --- |
| **Товар** |
| мука |
| свинина |
| треска |

Вложенный запрос может содержать свои вложенные запросы. Подзапросы такого типа называются подзапросами с многоуровневым вложением. Максимальная глубина вложения (т. е. количество вложенных запросов) зависит от объема памяти, которым компонент Database Engine располагает для каждой инструкции SELECT. В случае подзапросов с многоуровневым вложением система сначала выполняет самый глубокий вложенный запрос и возвращает полученный результат запросу следующего уровня, который в свою очередь возвращает свой результат запросу следующего уровня над ним и т. д. Конечный результат выдается запросом самого высшего уровня.

**Подзапросы и операторы ANY и ALL**

Операторы ALL и ANY используются в тех случаях, когда необходимо проверить условие на соответствие, с каждым значением, которое вернул подзапрос. Они работают только с подзапросами.

Операторы ANY и ALL всегда используются в комбинации с одним из операторов сравнения. Оба оператора имеют одинаковый синтаксис:

**Имя\_столбца оператор\_сравнения [ANY|ALL] (подзапрос)**

При использовании оператора ANY логическое выражение

**Имя столбца оператор\_сравнения ANY(подзапрос)**

Функция ANY будет иметь значение Истина (true) только для тех строк таблицы, в которых значение, записанное в поле Имя\_столбца, при сравнении с помощью оператора\_сравнения хотя бы с одним из возвращаемых подзапросом значений дает значение Истина.

Например

**use Shop**

**SELECT empId, LName as Фамилия, FName as Имя**

**FROM Employees**

**WHERE empId= ANY(SELECT empId FROM Ledger)**

В этом запросе подзапрос SELECT empId FROM Ledger возвращает все значения поля empId из таблицы Ledger. Усли в таблице Employees есть строки с совпадающими значениями поля empId, то внешний запрос выведет поля empId, LName и FName этих строк.

Результат выполнения этого запроса:

|  |  |  |
| --- | --- | --- |
| **empId** | **Фамилия** | **Имя** |
| 1001 | Сизов | Александр |
| 1004 | Петунин | Николай |

Логическое выражение в запросе с использованием оператора ALL будет истинным, если оно будет истинным для каждого значения, выбранного подзапросом, стоящим после слова ALL.

Пример.

**use Shop**

**SELECT empId, LName as Фамилия, FName as Имя**

**FROM Employees**

**WHERE empId > ALL(SELECT empId FROM Ledger)**

В результирующий набор приведенного запроса попадут все сотрудники из таблицы Employees, у которых значение поля empId больше, чем любое значение empId, возвращаемое подзапросом из таблицы Ledger.

Результат выполнения этого запроса:

|  |  |  |
| --- | --- | --- |
| **empId** | **Фамилия** | **Имя** |
| 1005 | Бамбизо | Виталий |
| 1006 | Федосеенко | Валерий |
| 1007 | Мазуров | Владимир |
| 1008 | Жданович | Юрий |
| 1009 | Бондаренко | Сергей |

**Подзапросы и функция EXISTS**

Функция EXISTS принимает подзапрос (вложенный запрос) в качестве аргумента и возвращает значение FALSE, если вложенный запрос не возвращает строк и значение TRUE в противном случае.

Пример 4.3. Выборка дат продажи товаров с идентификатором goodId=1

**USE Shop;**

**SELECT slaeId, saledate**

**FROM Ledger L**

**WHERE EXISTS (SELECT \* FROM sales S**

**WHERE L.saleId = S.saleId**

**AND goodId =1);**

Результат выполнения этого запроса:

|  |  |
| --- | --- |
| **saleId** | **saledate** |
| 1 | 2020-09-02 |
| 2 | 2020-09-03 |

Запрос 4.3 выполняется следующим образом. Сначала внешний запрос выбирает первую строку таблицы Ledger (saleId=1). Запрос, являющийся аргументом функции EXISTS определяет, есть ли в таблице Sales строки со значением поля saleId =1 и значением goodId=1. Такая строка существует, поэтому функция EXISTS возвращает значение TRUE и строка со значением saleId=1 из таблицы Ledger попадает в результирующий набор. Эти операции производятся над всеми строками таблицы Ledger. В итоге в результирующий набор выводятся только идентификаторы и даты тех покупок, в которых присутствует товар с идентификатором 1 (goodId=1).

**Задание на лабораторную работу.**

Используя подзапросы получить следующую информацию из базы данных.

**Вариант 1**

 1. Список читателей, у которых на руках находятся книги.

2. Список читателей, у которых находятся книги, срок возврата которых прошел.

3. Список издательств, книги которых поступили в прошлом году.

4. Фамилию читателя с максимальным сроком задержки возврата книги.

5. Фамилии читателей, которые должны вернуть книги в ближайшие 3 дня.

**Вариант 2**

1. Список клиентов, имеющих карточки, срок действия которых не истек.

2. Список клиентов, имеющих карточки «Visa Master».

3. Список клиентов, вносивших деньги на свои счета за последние 3 дня.

4. Фамилию клиента, внесшего максимальную сумму в прошлом месяце.

5. Список клиентов, имеющих более одной карточки.

**Вариант 3**

1. Список сотрудников, занимающих должность «Программист».
2. Список сотрудников, уволенных в прошлом месяце.
3. Список отделов, в которых есть программисты.
4. Название отдела, в котором максимальное количество сотрудников.
5. Список отделов, в которые были приняты новые сотрудники за последние 3 дня.

**Вариант 4**

1. Список старост факультета ФИТР.
2. Номер группы, в которой числится максимальное количество студентов.
3. Список студентов, у которых отсутствует номер или дата приказа о зачислении.
4. Список групп, в которых числятся студенты Иванов или Петров или Сидоров.
5. Список студентов, зачисленных два года назад.

**Вариант 5**

1. Список водителей, выезжавших в рейс на прошлой неделе.
2. Список автомобилей марки «МАЗ».
3. Список автомобилей, выезжавших в рейс на автомобилях МАЗ, ВАЗ или ГАЗ.
4. Фамилию водителей, выезжавшего в рейс максимальной протяженности.
5. Список водителей, не выезжавших в рейс за последние 3 дня.

**Вариант 6**

1. Список номеров стоимостью более 50 руб.
2. Список номеров, освобождающихся на следующей неделе.
3. Список клиентов, имеющих бронь на номера.
4. Список самых дешевых номеров.
5. Список клиентов, проживающих в номерах «Люкс» на втором этаже.

**Вариант 7**

1. Список врачей – терапевтов.
2. Список пациентов, записанных на прием к терапевту на сегодня.
3. Фамилию врача, к которому записано максимальное количество пациентов на сегодня.
4. Список врачей, которых посетил пациент Иванов на прошлой неделе.
5. График приема врача Петрова на сегодня.

**Вариант 8**

1. Список товаров, входящих в «Группу1».
2. Список адресов и фамилий клиентов, которым товар должен быть доставлен сегодня.
3. Фамилию и адрес клиента, сделавшего самый дорогой заказ.
4. Список номеров заказов, в которых присутствуют товары «Товар1» или «Товар2» или «Товар3».
5. Список адресов, по которым доставлялись заказы за последние 3 дня.

**Вариант 9**

1. Список аэропортов, в которые осуществляются полеты из аэропорта Минск.
2. Список пассажиров на рейс Минск-Париж на сегодня.
3. Список рейсов, на которые приобретал билет пассажир Иванов.
4. Список номеров рейсов и цену билета в аэропорты Берлин, Париж, Амстердам из Минска.
5. Номер рейса, на который продано максимальное количество билетов на прошлой неделе.

**Вариант 10**

1. Список учеников, не имеющих оценок по математике.
2. Список предметов, которые преподает преподаватель Иванов.
3. Список учеников, получивших оценки ниже 6 на прошлой неделе.
4. Фамилию ученика, имеющего максимальное количество оценок.
5. Фамилии и оценки учеников по предметам Математика, Физика, Информатика.

**Вариант 11**

1. Список спектаклей на следующую неделю.
2. Название спектакля, на который продано максимальное количество билетов.
3. Список спектаклей, на которые не продано ни одного билета на прошлой неделе.
4. Список свободных мест на спектакль «Отелло» на завтра.
5. Список свободных мест на спектакль «Отелло» на завтра в 1, 2 и 5 рядах.

**Вариант 12**

1. Список студентов, размещенных в общежитии вчера.
2. Список студентов, размещенных в двухместных комнатах.
3. Список номеров комнат, в которых размещены студенты группы «Группа1».
4. Номер комнаты, в которой проживает студент Петров.
5. Номера групп, в которых учатся студенты, проживающие в комнате номер 99.

**Вариант 13**

1. Список дисциплин, читаемых кафедрой ПОВТ.
2. Список дисциплин, по которым в 5 семестре сдают экзамен.
3. Список дисциплин, изучаемых на 1 и 2 курсах.
4. Название дисциплины, количество часов на изучение которой максимально.
5. Список дисциплин модуля «Модуль1».

**Вариант 14**

1. Список районов города Минска.
2. Город, имеющий максимальное количество районов.
3. Список типов недвижимости в Первомайском районе г. Минска.
4. Список типов недвижимости в Минске, Бресте и Гомеле.
5. Список районов с самой дешевой недвижимостью.

**Вариант 15**

1. Список клиентов, записанных на сегодня.
2. Список клиентов, записанных на самые дорогие услуги.
3. Список клиентов, которых обслуживал сотрудник Иванов вчера.
4. Список услуг оказанных клиентам сотрудниками Иванов, Петров, Сидоров вчера.
5. Список услуг, не востребованных клиентами сегодня.

**Содержание отчета о лабораторной работе**

1. Название и цель работы.
2. Номер группы, фамилия и инициалы студента.
3. Задание на лабораторную работу.
4. Перечень запросов и результатов их выполнения.
5. Выводы.

**Контрольные вопросы:**

1. Что такое подзапрос?
2. Как используется оператор IN в подзапросе?
3. Для чего предназначены операторы ALL и ANY?
4. Для чего предназначена функция EXIST?

|  |  |
| --- | --- |
|  |  |