МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

ГЛАВНОЕ УПРАВЛЕНИЕ ПО ОБРАЗОВАНИЮ

МОГИЛЕВСКОГО ОБЛАСТНОГО ИСПОЛНИТЕЛЬНОГО КОМИТЕТА

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

“МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ”

Специальность 2-40 01 01

Учебный предмет Базы данных и системы

управления базами данных

УТВЕРЖДАЮ

Заместитель директора

по учебной работе

\_\_\_\_\_\_\_\_\_М.М. Федоськова

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ЛАБОРАТОРНАЯ РАБОТА № 9**

**Исследование простых SELECT-запросов на выборку данных**

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ**

Разработал преподаватель

Базанова А.Д.

2022

Обсуждено и одобрено

на заседании цикловой комиссии

специальностей в области

программного обеспечения

информационных систем

Протокол № \_\_ от \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## 1 Цель работы

1.1 Формирование умений создавать простые SELECT-запросы с операторами IN, BETWEEN, LIKE, IS NULL.

**2 Методическое и материальное обеспечение**

2.1 Методические рекомендации по выполнению лабораторной работы

2.2 Персональный компьютер IBM PC

2.3 СУБД MySQL

**3 Последовательность выполнения работы**

3.1 Изучить теоретическое обоснование лабораторной работы

3.2 Выполнить индивидуальное задание

3.3 Составить ответы на контрольные вопросы

3.4 Составить отчет

**4 Теоретическое обоснование**

Рассмотрим пример **MySQL условия IN**, используя символьные значения.

Ниже приведен пример MySQL оператора SELECT, который использует условие IN для сравнения значений символов:

SELECT \*

FROM contacts

WHERE last\_name IN ('Bernard', 'Boy', 'Tomas');

Этот пример MySQL условия IN возвращает все строки из таблицы contacts, где last\_name - это Bernard, Boy или Tomas. Поскольку в SELECT используется \*, то все поля из таблицы contacts будут отображаться в результирующем наборе.

Вышеприведенный пример IN эквивалентен следующему оператору SELECT:

SELECT \*

FROM contacts

WHERE last\_name = 'Bernard'

OR last\_name = 'Boy'

OR last\_name = 'Tomas';

Как вы можете видеть, использование MySQL условия IN облегчает чтение и повышает эффективности.

Рассмотрим пример MySQL условия IN, используя числовые значения.

SELECT \*

FROM suppliers

WHERE supplier\_id IN (20, 23, 31, 40);

Этот пример использования MySQL условия IN будет возвращать всех suppliers, где supplier\_id равен 20, 23, 31 или 40.

Рассмотрим пример условия IN, используя оператор NOT.

SELECT \*

FROM contacts

WHERE last\_name NOT IN ('Bernard', 'Boy', 'Tomas');

Этот пример MySQL условия IN возвращает все строки из таблицы contacts, где last\_name не является Bernard, Boy или Tomas. Иногда более эффективно перечислять значения, которые вы не хотите видеть в результате, в отличие от желаемых значений.

Рассмотрим некоторые примеры **MySQL условий BETWEEN**, используя числовые значения. Следующий пример использует условие BETWEEN для извлечения значений в числовом диапазоне.

SELECT \*

FROM contacts

WHERE contact\_id BETWEEN 50 AND 100;

Этот MySQL пример BETWEEN возвращает все строки из таблицы contacts, где contact\_id находится между 50 и 100 (включительно). Это эквивалентно следующему оператору SELECT:

SELECT \*

FROM contacts

WHERE contact\_id >= 50

AND contact\_id <= 100;

Рассмотрим, как вы будете использовать MySQL условие BETWEEN с датами. При использовании условия BETWEEN в MySQL с датами обязательно используйте функцию CAST для явного преобразования значений в даты.

В следующем примере используется условие BETWEEN для извлечения значений в диапазоне дат.

SELECT \*

FROM order\_details

WHERE order\_date BETWEEN CAST('2017-12-01' AS DATE) AND CAST('2017-12-31' AS DATE);

Этот пример MySQL условия BETWEEN возвращает все записи из таблицы order\_details, где order\_date находится между 1 декабря 2017 года и 31 декабря 2017 года (включительно). Это будет эквивалентно следующему оператору SELECT:

SELECT \*

FROM order\_details

WHERE order\_date >= CAST('2017-12-01' AS DATE)

AND order\_date <= CAST('2017-12-31' AS DATE);

Рассмотрим как % работает в **MySQL условии LIKE**. Мы хотим найти всех customers, last\_name которых начинается с «Ber».

SELECT customer\_name

FROM customers

WHERE last\_name LIKE 'Ber%';

Вы также можете использовать % несколько раз в одной строке. Например:

SELECT customer\_name

FROM customers

WHERE last\_name LIKE '%ns%';

В этом примере MySQL условия LIKE мы ищем всех customers, у которых last\_name содержит символы 'ns'.

Рассмотрим, как знак \_ (подстановочный символ подчеркивания) работает в MySQL условии LIKE. Помните, что подстановочный символ \_ означает только один символ. Например:

SELECT supplier\_name

FROM suppliers

WHERE supplier\_name LIKE 'Ber\_ard';

Этот пример MySQL условия LIKE возвращает всех suppliers, supplier\_name которых составляет 7 символов, причем первые три символа - «Ber», а последние три символа - «ard». Например, он может вернуть всех, supplier\_name которых - 'Bernard', 'Berzard', 'Bermard', 'Bersard' и т.д.

И еще один пример:

SELECT \*

FROM suppliers

WHERE account\_number LIKE '12345\_';

Вам может понадобиться найти номер учетной записи, но у вас есть только 5 из 6 цифр. В приведенном выше примере будет извлечено потенциально 10 записей (где отсутствующее значение могло бы равняться чему угодно от 0 до 9). Например, запрос может вернуть suppliers, чьи номера учетной записи: 123450, 123451, 123452, 123453, 123454, 123455, 123456, 123457, 123458, 123459

**MySQL функция** **ISNULL** проверяет, является ли выражение NULL.

Рассмотрим примеры MySQL функции ISNULL, чтобы понять, как использовать функцию ISNULL в MySQL.

Например:

mysql> SELECT ISNULL('google.ru');

#Результат: 0

mysql> SELECT ISNULL('');

#Результат: 0

mysql> SELECT ISNULL(NULL);

#Результат: 1

mysql> SELECT ISNULL(28);

#Результат: 0

**5 Индивидуальное задание**

5.1 В созданной базе данных из лабораторной работы № 6, произвести выборку всех данных.

5.2 Создать три простых SELECT-запроса с операторами IN.

5.3 Создать три простых SELECT-запроса с операторами BETWEEN.

5.4 Создать три простых SELECT-запроса с операторами LIKE.

5.5 Создать три простых SELECT-запроса с операторами IS NULL.

**6 Содержание отчета** **(в электронном виде, в рабочей папке учащегося)**

6.1 Тема работы

6.2 Цель работы

6.3 Выполненное индивидуальное задание

6.4 Ответы на контрольные вопросы

1. **Контрольные вопросы**
   1. Для чего используется оператор ISNULL?
   2. Какие запросы используются для извлечения значений в числовом диапазоне?
   3. Какой запрос используется для сравнения значений?

**Список использованных источников**

1 Дюбуа П. MySQL/ П. Дюбуа.– 2-е издание – М.: Вильямс, 2004. – 198-211 с.

2 Хомоненко, А.Д. Базы данных: учеб./ А.Д. Хомоненко [и др.]: под ред. проф. А.Д. Хомоненко. Санкт-Петербург, 2003, с. 277-364.