**Практическая работа № 11\_12. Создание SQL запросов.**

**Цель работы:** Научиться писать запросы к БД на языке SQL.

**Оборудование:** компьютер.

**Содержание работы:**

1. Добавил записи в таблицы  
    USE Applications

INSERT INTO app\_cants (applicant\_id, applicant\_FIO,applicant\_phone,applicant\_adress)

VALUES (12, 'Агеев Максим', '+7-946-952-79-95', 'г. Москва'),

(1, 'Березюк Никита', '+7-946-852-78-85', 'г. Москва'),

(32, 'Ганиев Амир', '+7-946-752-77-75', 'г. Москва'),

(8, 'Алямкин Иван', '+7-946-652-76-65', 'г. Москва'),

(16, 'Сергеев Кирилл', '+7-946-552-75-55', 'г. Москва'),

(45, 'Телепов Дмитрий', '+7-946-452-74-45', 'г. Москва'),

(9, 'Якимов Дмитрий', '+7-946-352-73-35', 'г. Москва'),

(78, 'Эверстова Нарияна', '+7-946-252-72-25', 'г. Москва'),

(10, 'Фиатов Денис', '+7-946-152-71-15', 'г. Москва'),

(23, 'Купцов Владимир', '+7-946-052-70-05', 'г. Москва')

1. Вывел количество записей в каждой таблице  
    USE Applications

SELECT \*

FROM app\_cants

USE Applications

SELECT \*

FROM app\_tions

USE Applications

SELECT \*

FROM ob\_cts

1. Соединил таблицы, вывел нужные поля по усмотрению, отсортировал данные в алфавитном порядке. Сформулировал полученный запрос.  
   Формулировка запроса: Вывести номер заявления, название объекта и ФИО заявителя. Отсортировать названия объектов в алфавитном порядке.  
    USE Applications

SELECT app\_tions.application\_id, ob\_cts.obct\_name, app\_cants.applicant\_FIO

FROM app\_tions INNER JOIN app\_cants

ON app\_tions.application\_applicant=app\_cants.applicant\_id

INNER JOIN ob\_cts

ON app\_tions.application\_object=ob\_cts.obct\_id

ORDER BY ob\_cts.obct\_name

1. Напечатал названия или фамилии из первых трёх записей таблицы, упорядоченной по алфавиту. Сформулировал полученный запрос.  
   Формулировка запроса: вывести ФИО трёх заявителей, отсортированных по алфавиту.  
    USE Applications

SELECT TOP(3)applicant\_FIO

FROM app\_cants

ORDER BY applicant\_FIO

1. Из любой из таблиц (или запроса на выборку), в которых дублируются названия, вывести названия так, чтобы они не дублировались, отсортировать их в порядке, обратном алфавитному.  
    USE Applications  
   SELECT DISTINCT app\_cants.applicant\_FIO, ob\_cts.obct\_name

FROM app\_tions

INNER JOIN app\_cants

ON app\_tions.application\_applicant=app\_cants.applicant\_id

INNER JOIN ob\_cts

ON app\_tions.application\_object=ob\_cts.obct\_id

ORDER BY applicant\_FIO DESC

1. Вывести фамилию(название) из последней записи таблицы.  
    USE Applications

SELECT TOP 1 applicant\_FIO

FROM app\_cants

ORDER BY applicant\_id DESC

1. Создать запрос на выборку названий или фамилий, которые начинаются на первую букву вашей фамилии, а заканчиваются на последнюю букву вашей фамилии. Привести несколько вариантов решения. Сравнить результаты.  
   Вариант №1  
    USE Applications

SELECT obct\_name

FROM ob\_cts

WHERE LEFT(obct\_name,1) = 'Т' AND RIGHT(obct\_name,1) = 'в'

Вариант №2  
 USE Applications

SELECT obct\_name

FROM ob\_cts

WHERE obct\_name = 'Т%в'

1. Определить зависимую таблицу. Удалить из зависимой таблицы запись с заданным значением атрибута. Сформулировать полученный запрос.  
   Формулировка запроса: Удалить заявление с ID равным значению «20»

USE Applications

DELETE

FROM app\_tions

WHERE application\_id = 20

1. Определить главную таблицу. Обновить значение первичного ключа в главной таблице.

USE Applications

UPDATE ob\_cts

SET obct\_name = 'ГЭС'

WHERE obct\_id = 9

1. Выбрать записи в заданном временном диапазоне. Например, выбрать расписание загрузки аудиторий с 7.10.19 сроком на 2 недели или выбрать фамилии сотрудников, у которых стаж работы от 5 до 15 лет. Сформулировать полученный запрос. Привести несколько вариантов решения. Сравнить результаты.  
   Формулировка запроса: Вывести все данные о заявлениях в интервале от 2000 до 2002 года.  
   Вариант №1

SELECT \*

FROM app\_tions

WHERE application\_date >= '2000-01-01' AND application\_date <= '2002-01-01'

Вариант №2

SELECT \*

FROM app\_tions

WHERE application\_date BETWEEN '2000-01-01' AND '2002-01-01'

1. Выбрать записи из всех таблиц по нескольким критериям одновременно. Сформулировать полученный запрос. Привести по возможности несколько вариантов решения.  
   Формулировка запроса: Вывести все данные о заявлении 2002 года, о его заявителе, и об объекте заявления.

SELECT \*

FROM app\_tions INNER JOIN app\_cants

ON app\_tions.application\_applicant = app\_cants.ID

INNER JOIN ob\_cts

ON app\_tions.application\_object = ob\_cts.obct\_id

WHERE YEAR(app\_tions.application\_date)=2002 AND app\_cants.ID=9 AND ob\_cts.obct\_id=3

1. Вывести сумму заказа, сумму оценок, сумму доходов и т.д, того, что применимо к Вашей предметной области. Сформулировать полученный запрос.  
   Ответ: не применимо к моей предметной области.
2. Заменить фамилию последнего человека на собственную, при условии, что таблица упорядочена по номеру человека. (Заменить название в последней записи таблицы на новое, при условии, что таблица упорядочена по номеру.)

UPDATE app\_cants

SET Last\_name = 'Телепова'

WHERE ID=78

1. \*Сформулировать и написать собственный запрос с использованием встроенных строковых функций.  
   Формулировка запроса: Вывести первые 5 букв в каждом тексте заявления.

SELECT SUBSTRING(application\_text,1,5)

FROM app\_tions

1. \* Сформулировать и написать собственный запрос с использованием встроенных функций даты.  
   Формулировка запроса: вывести дни недели, в которые ответили на заявления.

SELECT CASE DATEPART(dw,applcation\_answer\_date)

WHEN 7 THEN 'Суббота'

WHEN 6 THEN 'Пятница'

WHEN 5 THEN 'Четверг'

WHEN 4 THEN 'Среда'

WHEN 3 THEN 'Вторник'

WHEN 2 THEN 'Понедельник'

WHEN 1 THEN 'Воскресенье'

END

FROM app\_tions

1. \*Сформулировать и написать собственный запрос с использованием несимметричных соединений.  
   Формулировка запроса: Вывести фамилии тех заявителей, которые подавали заявления на рассмотрение.

SELECT app\_cants.Last\_name

FROM app\_tions LEFT JOIN app\_cants

ON app\_tions.application\_applicant=app\_cants.ID

1. \* Сформулировать и написать собственный запрос с использованием оператора CASE.  
   Формулировка запроса: вывести дни недели, в которые ответили на заявления.

SELECT CASE DATEPART(dw,applcation\_answer\_date)

WHEN 7 THEN 'Суббота'

WHEN 6 THEN 'Пятница'

WHEN 5 THEN 'Четверг'

WHEN 4 THEN 'Среда'

WHEN 3 THEN 'Вторник'

WHEN 2 THEN 'Понедельник'

WHEN 1 THEN 'Воскресенье'

END

FROM app\_tions

1. \* Сформулировать и написать собственный запрос с использованием групповых функций: COUNT, MAX, MIN, AVG, SUM.  
   Формулировка запроса: вывести самую позднюю дату и самую раннюю дату ответа на заявления.

SELECT MAX(applcation\_answer\_date), MIN(applcation\_answer\_date)

FROM app\_tions

1. Сформулировать и написать собственный запрос с использованием подзапроса. Привести по возможности другие варианты решения.  
   Формулировка запроса: вывести номера объектов заявителя с фамилией Телепов

SELECT app\_tions.application\_object

FROM app\_tions

WHERE application\_applicant=(

SELECT ID

FROM app\_cants

WHERE Last\_name='Телепов'

)