***Гуркова Виктория* *381 группа***

***Library Database***

**Задание № 3. *Создать следующие виды запросов к базе:***

1. **Запросы с использованием различных видов соединений таблиц (Inner join, Outer join, Cross join, Cross apply, самосоединение).**

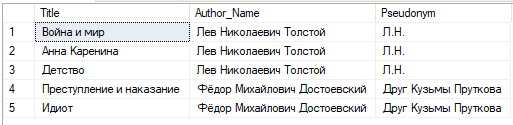
-- INNER JOIN --

SELECT b.Title, a.Full\_Name AS Author\_Name, a.Pseudonym

FROM Book b

INNER JOIN Author a ON b.Author\_ID = a.Author\_ID

Результат:

**

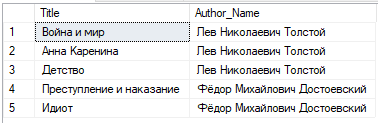
-- LEFT OUTER JOIN --

SELECT b.Title, a.Full\_Name AS Author\_Name

FROM Book b

LEFT OUTER JOIN Author a ON b.Author\_ID = a.Author\_ID

Результат:



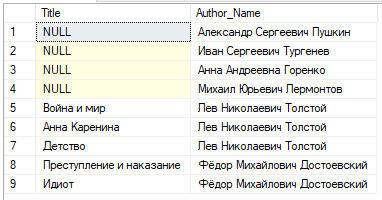
-- RIGHT OUTER JOIN --

SELECT b.Title, a.Full\_Name AS Author\_Name

FROM Book b

RIGHT OUTER JOIN Author a ON b.Author\_ID = a.Author\_ID

Результат:

****

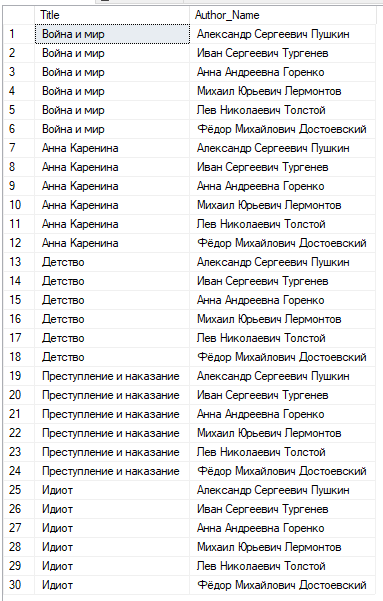
-- CROSS JOIN --

SELECT b.Title, a.Full\_Name AS Author\_Name

FROM Book b

CROSS JOIN Author a

Результат:

****

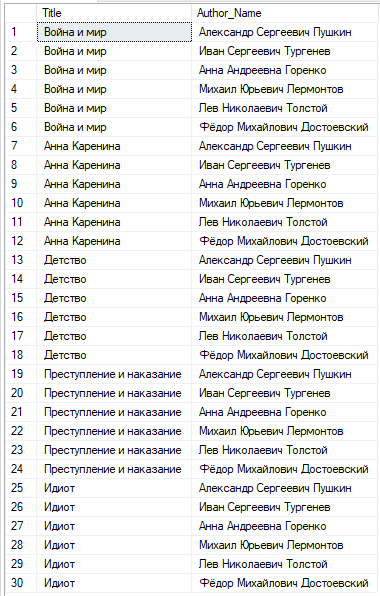
-- CROSS APPLY --

SELECT b.Title, a.Full\_Name AS Author\_Name

FROM Book b

CROSS APPLY Author a

Результат:

****

-- САМОСОЕДИНЕНИЕ --

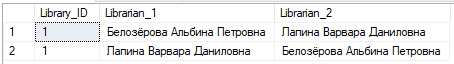
SELECT l1.Library\_ID, l1.Full\_Name AS Librarian\_1, l2.Full\_Name AS Librarian\_2

FROM Librarian l1 JOIN Librarian l2

ON l1.Library\_ID = l2.Library\_ID

WHERE l1.Librarian\_ID <> l2.Librarian\_ID

Результат:



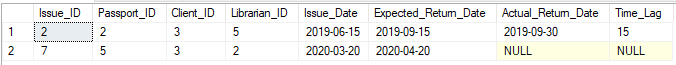
1. **Фильтрация данных в запросах с использованием предикатов (EXISTS, IN, ALL, SOME/ANY, BETWEEN, LIKE).**

-- EXISTS --

SELECT \* FROM Issue WHERE Client\_ID = 3 AND

EXISTS (SELECT Client\_ID FROM Client)

Результат:



SELECT Title FROM Book WHERE Author\_ID = 3 AND

EXISTS (SELECT Author\_ID FROM Author)

Результат: результата нет, так как у автора с ID = 3 нет книг в базе

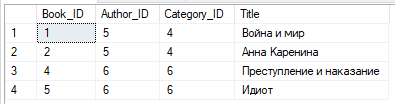


-- IN --

SELECT \* FROM Book

WHERE Category\_ID IN (4, 6)

Результат:



-- ANY --

SELECT Full\_Name, Pseudonym FROM Author

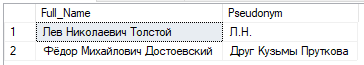
WHERE Author\_ID = ANY(SELECT Author\_ID FROM Book)

--SOME --

SELECT Full\_Name, Pseudonym FROM Author

WHERE Author\_ID = SOME(SELECT Author\_ID FROM Book)

Результат: предикаты **SOME** и **ANY** выдают одинаковый результат



-- ALL --

SELECT \* FROM Issue

WHERE Issue\_Date >= ALL(SELECT Issue\_Date FROM Issue)

Результат:

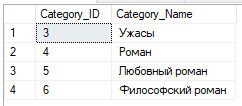


-- BETWEEN --

SELECT Category\_ID, Category\_Name FROM Category

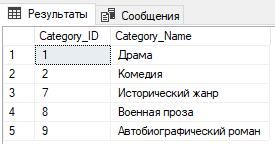
WHERE Category\_ID BETWEEN 3 AND 6

Результат:



SELECT Category\_ID, Category\_Name FROM Category

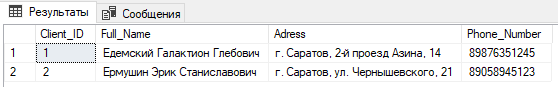
WHERE Category\_ID NOT BETWEEN 3 AND 6



-- LIKE --

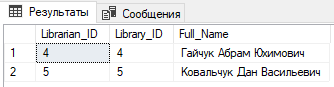
SELECT \* FROM Client

WHERE Full\_Name LIKE 'е%'



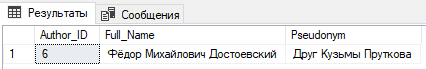
SELECT \* FROM Librarian

WHERE Full\_Name LIKE '%ч'



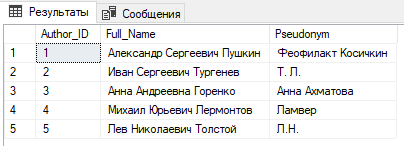
SELECT \* FROM Author

WHERE Pseudonym LIKE '% % %'



SELECT \* FROM Author

WHERE Pseudonym NOT LIKE '% % %'



1. **Запросы с использованием выражения CASE.**

-- CASE --

SELECT Instance\_ID, Publisher\_ID, Year\_of\_Publication,

CASE

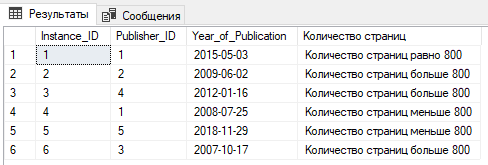
WHEN Amount\_of\_Page > 800 THEN 'Количество страниц больше 800'

WHEN Amount\_of\_Page = 800 THEN 'Количество страниц равно 800'

ELSE 'Количество страниц меньше 800'

END AS 'Количество страниц'

FROM Instance

****

SELECT Passport\_ID, Client\_ID, Librarian\_ID,

CASE

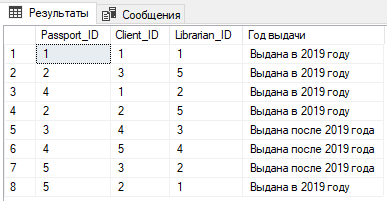
WHEN Issue\_Date > '31.12.2019' THEN 'Выдана после 2019 года'

WHEN Issue\_Date BETWEEN '01.01.2019' AND '31.12.2019' THEN 'Выдана в 2019 году'

ELSE 'Выдана до 2019 года'

END AS 'Год выдачи'

FROM Issue

****

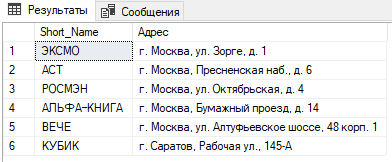
1. **Использование встроенных функций (функций преобразования типов (CAST, CONVERT), функций для проверки значений NULL**

**(ISNULL, NULLIF), COALESCE логических функций (CHOOSE,  IIF)).**

-- CAST --

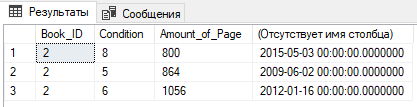
SELECT Short\_Name, CAST(Adress AS TEXT) AS 'Адрес'

FROM Publish\_House

****

SELECT Book\_ID, Condition, Amount\_of\_Page, CAST(Year\_of\_Publication AS DATETIME2)

FROM Instance WHERE Book\_ID = 2

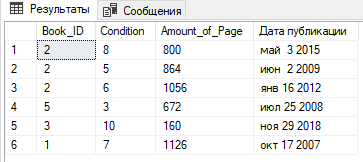


-- CONVERT --

SELECT Book\_ID, Condition, Amount\_of\_Page,

CONVERT(VARCHAR, Year\_of\_Publication, 100) AS 'Дата публикации'

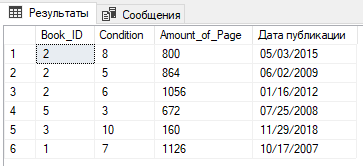
FROM Instance



SELECT Book\_ID, Condition, Amount\_of\_Page,

CONVERT(VARCHAR, Year\_of\_Publication, 101) AS 'Дата публикации'

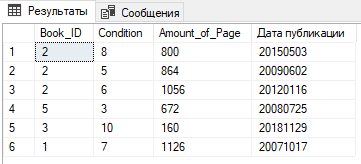
FROM Instance



SELECT Book\_ID, Condition, Amount\_of\_Page,

CONVERT(VARCHAR, Year\_of\_Publication, 112) AS 'Дата публикации'

FROM Instance

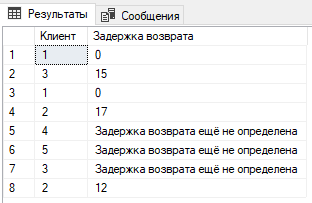


-- ISNULL --

SELECT Client\_ID AS 'Клиент', ISNULL(CAST(Time\_Lag AS VARCHAR(50)),

'Задержка возврата ещё не определена') AS 'Задержка возврата'

FROM Issue

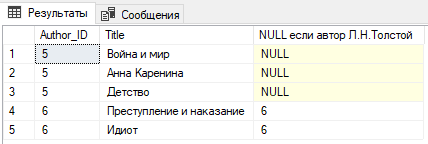


-- NULLIF --

SELECT Author\_ID, Title,

NULLIF(Author\_ID, 5) 'NULL если автор Л.Н.Толстой'

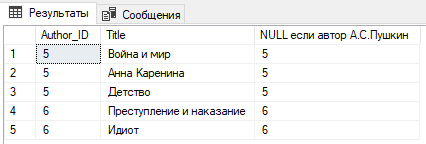
FROM Book



SELECT Author\_ID, Title,

NULLIF(Author\_ID, 1) 'NULL если автор А.С.Пушкин'

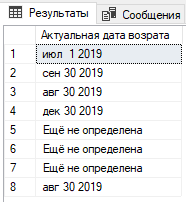
FROM Book



-- COALESCE --

SELECT COALESCE(CONVERT(VARCHAR(50),Actual\_Return\_Date,100), 'Ещё не определена') AS 'Актуальная дата возрата'

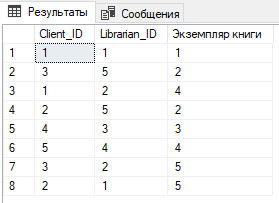
FROM Issue



-- CHOOSE --

SELECT Client\_ID, Librarian\_ID, CHOOSE(Passport\_ID, '1', '2', '3', '4', '5') AS 'Экземпляр книги'

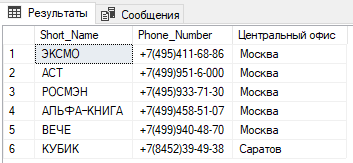
FROM Issue



-- IIF --

SELECT Short\_Name, Phone\_Number, IIF(City\_ID = 495, 'Москва', 'Саратов') AS 'Центральный офис'

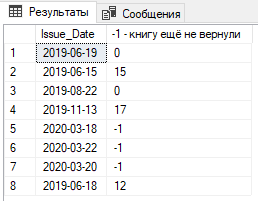
FROM Publish\_House



-- COALESCE and IIF--

SELECT Issue\_Date, COALESCE(IIF(Time\_Lag IS NULL, -1, Time\_Lag), Time\_Lag ) AS '-1 - книгу ещё не вернули'

FROM Issue

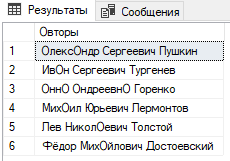


1. **Запросы с использованием функций для работы со строками (REPLACE, SUBSTRING, STUFF, STR, UNICODE, LOWER, UPPER).**

-- REPLACE --

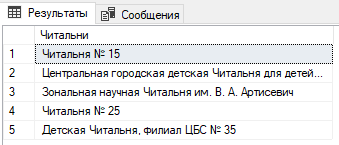
SELECT REPLACE(Full\_Name, 'А', 'О') AS 'Овторы'

FROM Author

****

SELECT REPLACE(Full\_Name, 'Библиотека', 'Читальня') AS 'Читальни'

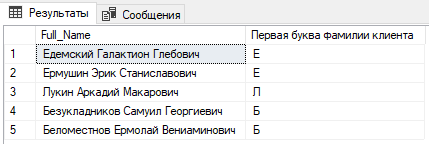
FROM Library

****

-- SUBSTRING --

SELECT Full\_Name, SUBSTRING(Full\_Name, 1, 1) AS 'Первая буква фамилии клиента'

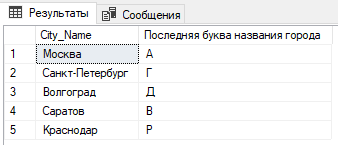
FROM Client

****

SELECT City\_Name, UPPER(SUBSTRING(City\_Name, LEN(City\_Name), 1)) AS

'Последняя буква названия города'

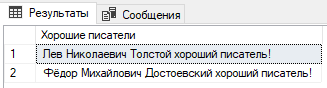
FROM City



-- STUFF --

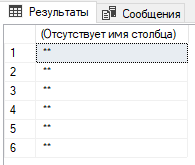
SELECT STUFF(Full\_Name, LEN(Full\_Name), 1, 'й хороший писатель!') AS 'Хорошие писатели'

FROM Author WHERE Author\_ID >= 5

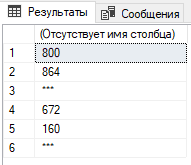
****

-- STR --

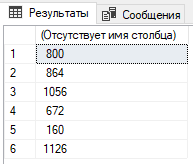
SELECT STR(CAST(Amount\_of\_Page AS FLOAT), 2, 0) FROM Instance

****

SELECT STR(CAST(Amount\_of\_Page AS FLOAT), 3, 0) FROM Instance

****

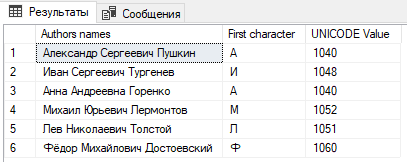
SELECT STR(CAST(Amount\_of\_Page AS FLOAT), 4, 0) FROM Instance

****

-- UNICODE --

SELECT Full\_Name AS 'Authors names', SUBSTRING(Full\_Name, 1, 1) AS 'First character', UNICODE(Full\_Name) AS 'UNICODE Value'

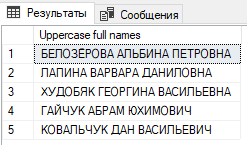
FROM Author



-- UPPER --

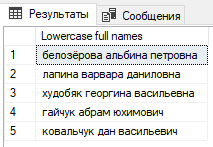
SELECT UPPER(Full\_Name) AS 'Uppercase full names'

FROM Librarian



-- LOWER --

SELECT LOWER(Full\_Name) AS 'Lowercase full names' FROM Librarian

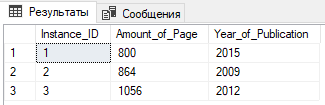
****

1. **Запросы с использованием функций даты и времени (DATEPART, DATEADD, DATEDIFF, GETDATE(), SYSDATE, TIMEOFFSET() и т.п.).**

-- DATEPART --

SELECT Instance\_ID, Amount\_of\_Page, DATEPART(YEAR, Year\_of\_Publication) AS 'Year\_of\_Publication'

FROM Instance WHERE Book\_ID = 2

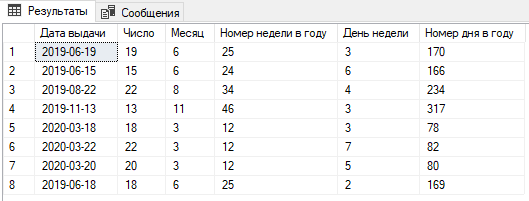
****

SELECT Issue\_Date AS 'Дата выдачи', DATEPART(DAY, Issue\_Date) AS 'Число', DATEPART(MONTH, Issue\_Date) AS 'Месяц',

DATEPART(WEEK, Issue\_Date) AS 'Номер недели в году', DATEPART(WEEKDAY, Issue\_Date)AS 'День недели',

DATEPART(DAYOFYEAR, Issue\_Date) AS 'Номер дня в году'

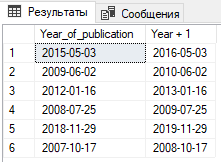
FROM Issue

****

-- DATEADD --

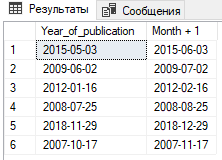
SELECT Year\_of\_publication, DATEADD(YEAR, 1, Year\_of\_Publication) AS 'Year + 1'

FROM Instance

****

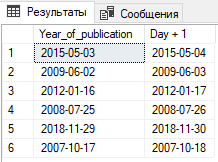
SELECT Year\_of\_publication, DATEADD(MONTH, 1, Year\_of\_Publication) AS 'Month + 1'

FROM Instance

****

SELECT Year\_of\_publication, DATEADD(DAY, 1, Year\_of\_Publication) AS 'Day + 1'

FROM Instance



-- DATEDIFF --

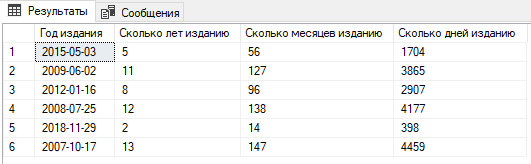
SELECT Year\_of\_Publication AS 'Год издания',

DATEDIFF(YEAR, Year\_of\_Publication, '01.01.2020') AS 'Сколько лет изданию',

DATEDIFF(MONTH, Year\_of\_Publication, '01.01.2020') AS 'Сколько месяцев изданию',

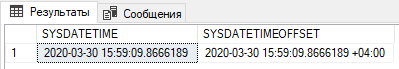
DATEDIFF(DAY, Year\_of\_Publication, '01.01.2020') AS 'Сколько дней изданию'

FROM Instance

****

SELECT SYSDATETIME() AS 'SYSDATETIME',

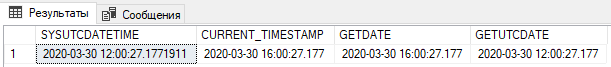
SYSDATETIMEOFFSET() AS 'SYSDATETIMEOFFSET'



SELECT SYSUTCDATETIME() AS 'SYSUTCDATETIME',

CURRENT\_TIMESTAMP AS 'CURRENT\_TIMESTAMP',

GETDATE() AS 'GETDATE', GETUTCDATE() AS 'GETUTCDATE'

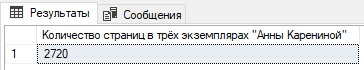


1. **Запросы с использованием агрегатных функций, группировок (GROUP BY) и фильтрации групп (HAVING).**

-- Агрегатные функции --

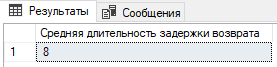
SELECT SUM(Amount\_of\_Page) AS 'Количество страниц в трёх экземплярах "Анны Карениной"'

FROM Instance WHERE Book\_ID = 2



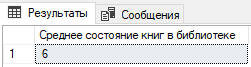
SELECT AVG(Time\_Lag) AS 'Средняя длительность задержки возврата'

FROM Issue



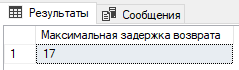
SELECT AVG(Condition) AS 'Среднее состояние книг в библиотеке'

FROM Instance



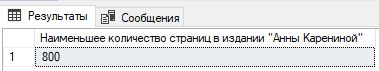
SELECT MAX(Time\_Lag) AS 'Максимальная задержка возврата'

FROM Issue



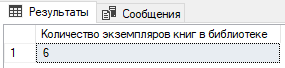
SELECT MIN(Amount\_of\_Page) AS 'Наименьшее количество страниц в издании "Анны Карениной"'

FROM Instance WHERE Book\_ID = 2



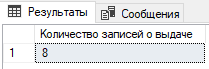
SELECT COUNT(Instance\_ID) AS 'Количество экземпляров книг в библиотеке'

FROM Instance



SELECT COUNT(\*) AS 'Количество записей о выдаче'

FROM Issue

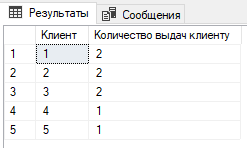
****

-- GROUP BY --

SELECT Client\_ID AS 'Клиент', COUNT(Issue\_ID) AS 'Количество выдач клиенту'

FROM Issue

GROUP BY Client\_ID

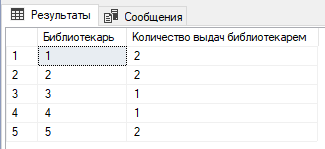


SELECT Librarian\_ID AS 'Библиотекарь', COUNT(Issue\_ID) AS

'Количество выдач библиотекарем'

FROM Issue

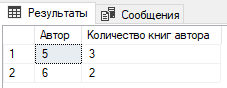
GROUP BY Librarian\_ID



SELECT Author\_ID AS 'Автор', COUNT(Book\_ID) AS 'Количество книг автора'

FROM Book

GROUP BY Author\_ID



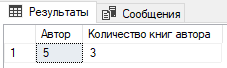
-- HAVING --

SELECT Author\_ID AS 'Автор', COUNT(Book\_ID) AS 'Количество книг автора'

FROM Book

GROUP BY Author\_ID

HAVING COUNT(Book\_ID) > 2



SELECT Book\_ID, COUNT(Instance\_ID) AS 'Количество экземпляров', AVG(Condition) AS 'Среднее состояние её экземпляро'

FROM Instance

GROUP BY Book\_ID

HAVING AVG(Condition) > 6

