|  |  |
| --- | --- |
|  | МИНПРОСВЕЩЕНИЯ РОССИИ  Федеральное государственное бюджетное образовательное учреждение  высшего образования  «Тульский государственный педагогический университет им. Л.Н. Толстого»  (ТГПУ им. Л.Н. Толстого) |

**ОТЧЕТ**

**по дисциплине:**

**«Базы данных»**

**Лабораторная работа № 11**

Выполнил:

студент группы №1521731

института передовых информационных технологий

Шкатов Никита Николаевич

Тула – 2024

**Оглавление**

[Задание №1: 3](#_Toc190934440)

[Задание №2: 4](#_Toc190934441)

[Задание №3: 6](#_Toc190934442)

[Задание №4: 6](#_Toc190934443)

[Задание №5: 9](#_Toc190934444)

[Задание №6: 10](#_Toc190934445)

[Задание №7: 11](#_Toc190934446)

[Задание №8: 13](#_Toc190934447)

[Задание №9: 14](#_Toc190934448)

[Задание №10: 15](#_Toc190934449)

[Задание №11: 16](#_Toc190934450)

# Задание №1:

1. Напишите запрос, демонстрирующий соединение двух таблиц с помощью конструкции SELECT ... FROM TABLE1, TABLE2 WHERE... . Перепишите тот же запрос с помощью конструкции JOIN. Убедитесь, что результаты выполнения запросов одинаковы.

**Решение:**

Запрос, соединяющий таблицы «Сотрудник» и «Отдел» для получения информации о сотрудниках и их отделах.

Программный код:

SELECT "ФИО", "Название\_отдела"

FROM "Сотрудник", "Отдел"

WHERE "Сотрудник"."ID\_Сотрудника" = "Отдел"."ID\_Сотрудника";

SELECT "ФИО", "Название\_отдела"

FROM "Сотрудник"

JOIN "Отдел" ON "Сотрудник"."ID\_Сотрудника" = "Отдел"."ID\_Сотрудника";

Таблица:



# Задание №2:

2. Напишите запрос, демонстрирующий смысл и назначение конструкции LEFT JOIN. Перепишите его с помощью конструкции RIGHT JOIN. Убедитесь, что результаты выполнения запросов одинаковы.

**Решение:**

Запрос, объединяющий данные о сотрудниках и их отделах, возвращая всех сотрудников и соответствующие отделы. Если сотрудник не принадлежит ни одному отделу, то в результирующем наборе будут пропущенные значения (NULL) для столбцов из таблицы «Отдел».

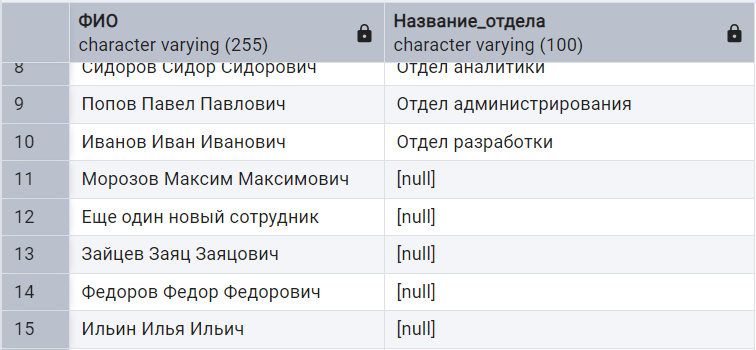
Программный код:

SELECT "ФИО", "Название\_отдела"

FROM "Сотрудник"

LEFT JOIN "Отдел" ON "Сотрудник"."ID\_Сотрудника" = "Отдел"."ID\_Сотрудника";

Таблица:



Запрос, объединяющий данные о сотрудниках и их отделах, возвращая все отделы и соответствующих сотрудников. Если в отделе нет сотрудников, то в результирующем наборе будут пропущенные значения (NULL) для столбцов из таблицы «Сотрудник».

SELECT "ФИО", "Название\_отдела"

FROM "Сотрудник"

RIGHT JOIN "Отдел" ON "Сотрудник"."ID\_Сотрудника" = "Отдел"."ID\_Сотрудника";

Таблица:



# Задание №3:

3. Напишите запрос, в котором таблица соединяется (JOIN) сама с собой.

**Решение:**

Запрос, соединяющий таблицу «Сотрудник» саму с собой.

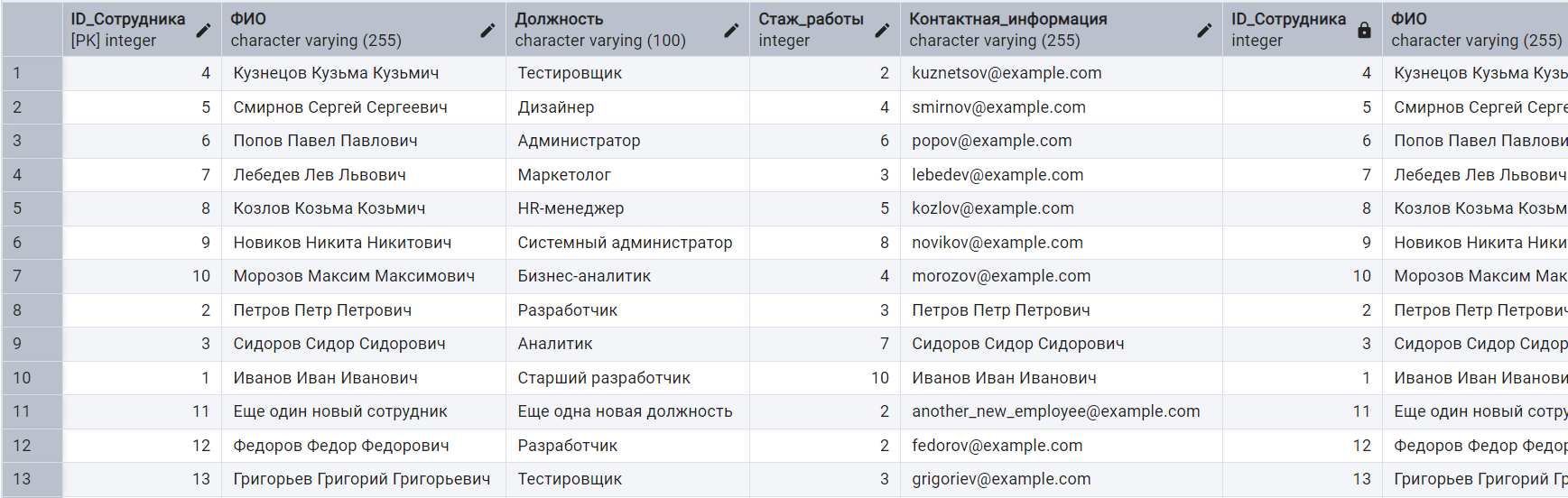
Программный код:

SELECT \*

FROM "Сотрудник" s1

JOIN "Сотрудник" s2 ON s1."ID\_Сотрудника" = s2."ID\_Сотрудника";

Таблица:



# Задание №4:

4. Напишите запрос, в котором агрегация происходит по результату соединения таблиц. То есть, в запросе должны присутствовать агрегирующая функция (SUM, AVG, MAX, MIN или COUNT), GROUP BY и HAVING, WHERE и JOIN (внутренний или внешний).

**Решение:**

Агрегация в SQL — это процесс объединения данных из нескольких строк в одну строку с использованием агрегатных функций.

Запрос получает информацию о клиентах, которые совершили заказы (услуги) на сумму, превышающую 5,000. Он позволяет агрегировать данные из нескольких связанных таблиц, чтобы получить суммарную стоимость выполненных заданий для каждого клиента.

Программный код:

SELECT

"Клиент"."ФИО",

SUM("Задание\_Стоимость"."Общая\_стоимость") AS "Суммарная\_стоимость\_заданий"

FROM

"Клиент"

JOIN

"Услуга" ON "Клиент"."ID\_Клиента" = "Услуга"."ID\_Клиента"

JOIN

"Задание" ON "Услуга"."ID\_Услуги" = "Задание"."ID\_Услуги"

JOIN

"Задание\_Стоимость" ON "Задание"."ID\_Задания" = "Задание\_Стоимость"."ID\_Задания"

WHERE

"Задание"."Статус\_задания" = 'Выполнено'

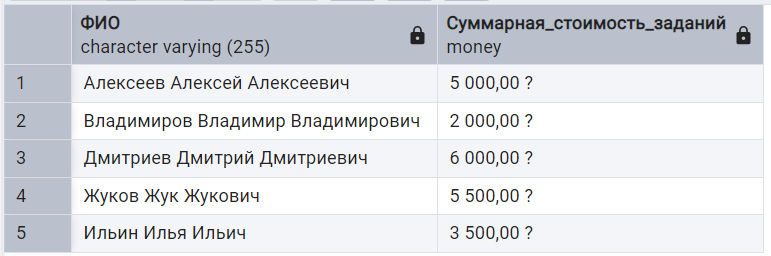
GROUP BY

"Клиент"."ФИО"

HAVING

SUM("Задание\_Стоимость"."Общая\_стоимость")::numeric > 1000;

Таблица:



# Задание №5:

5. Напишите пример запроса, в котором вместо любой из констант выражения, определяющего условие WHERE, используется скалярный подзапрос.

**Решение:**

Скалярный подзапрос — это подзапрос, который возвращает одно значение (один столбец и одну строку).

Запрос, выбирающий сотрудников, чей стаж работы больше среднего стажа всех сотрудников. Вместо константы в условии WHERE используется скалярный подзапрос для вычисления среднего стажа.

Программный код:

SELECT "ФИО", "Должность", "Стаж\_работы"

FROM "Сотрудник"

WHERE "Стаж\_работы" > (

SELECT AVG("Стаж\_работы")

FROM "Сотрудник"

);

Таблица:



# Задание №6:

6. Напишите запрос с векторным подзапросом (ключевое слово IN).

**Решение:**

Векторный подзапрос — это подзапрос, который возвращает несколько значений (один столбец и несколько строк).

Запрос, выбирающий всех сотрудников, которые работают в отделах, связанных с разработкой или тестированием.

Программный код:

SELECT "ФИО", "Должность"

FROM "Сотрудник"

WHERE "ID\_Сотрудника" IN (

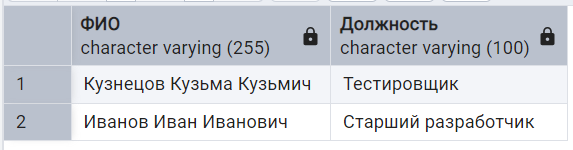
SELECT "ID\_Сотрудника"

FROM "Отдел"

WHERE "Название\_отдела" IN ('Отдел разработки', 'Отдел тестирования')

);

Таблица:



# Задание №7:

7. Напишите два запроса с ключевыми словами ANY и ALL, делающие одно и то же.

**Решение:**

1) Запрос, выбирающий всех сотрудников-разработчиков, чей стаж работы больше, чем хотя бы у одного из сотрудников с должностью «Разработчик». (ANY)

2) Запрос, выбирающий всех сотрудников-разработчиков, чей стаж работы больше, чем у всех сотрудников с должностью «Аналитик». (ALL)

Программный код:

1) SELECT "ФИО", "Стаж\_работы"

FROM "Сотрудник"

WHERE "Должность" = 'Разработчик'

AND "Стаж\_работы" > ANY (

SELECT "Стаж\_работы"

FROM "Сотрудник"

WHERE "Должность" = 'Разработчик'

);

2) SELECT "ФИО", "Стаж\_работы"

FROM "Сотрудник"

WHERE "Должность" = 'Разработчик'

AND "Стаж\_работы" > ALL (

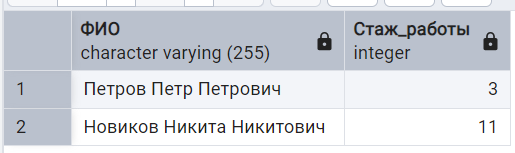
SELECT "Стаж\_работы"

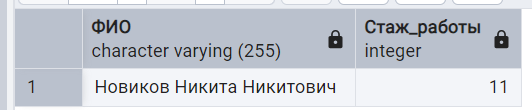
FROM "Сотрудник"

WHERE "Должность" = 'Аналитик'

);

Таблица:





# Задание №8:

8. Напишите пример запроса с табличным подзапросом (ключевое слово EXISTS).

**Решение:**

Табличный подзапрос — это подзапрос, который возвращает таблицу и может быть использован в основном запросе для фильтрации строк.

Запрос, выбирающий всех сотрудников, которые работают в отделах, где есть хотя бы один сотрудник с должностью «Разработчик».

Программный код:

SELECT s."ФИО", s."Должность"

FROM "Сотрудник" s

WHERE EXISTS (

SELECT 1

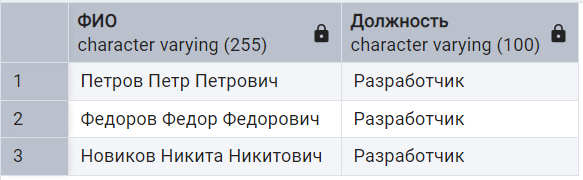
FROM "Сотрудник" s2

WHERE s2."Должность" = 'Разработчик'

AND s."ID\_Сотрудника" = s2."ID\_Сотрудника"

);

Таблица:



# Задание №9:

9. Напишите пример запроса несколькими уровнями вложенности.

**Решение:**

Запрос, выбирающий всех сотрудников, которые работают в отделах, где есть хотя бы один сотрудник с должностью «Разработчик» и стаж работы которого больше среднего стажа работы всех разработчиков.

Программный код:

SELECT s."ФИО", s."Должность"

FROM "Сотрудник" s

WHERE EXISTS (

SELECT 1

FROM "Сотрудник" s2

WHERE s2."Должность" = 'Разработчик'

AND s."ID\_Сотрудника" = s2."ID\_Сотрудника"

AND s2."Стаж\_работы" > (

SELECT AVG("Стаж\_работы")

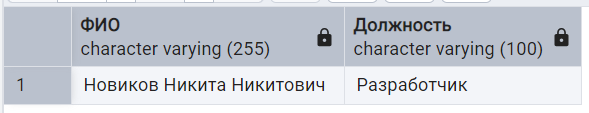
FROM "Сотрудник"

WHERE "Должность" = 'Разработчик'

)

);

Таблица:



# Задание №10:

10. Напишите пример запроса, в котором вместо таблицы, указываемой после ключевого слова FROM, используется подзапрос.

**Решение:**

Запрос, выбирающий ФИО и стаж работы сотрудников, которые работают в отделах, связанных с разработкой или тестированием, и имеют стаж работы более 3 лет.

Программный код:

SELECT "ФИО", "Стаж\_работы"

FROM (

SELECT С."ФИО", С."Стаж\_работы"

FROM "Сотрудник" С

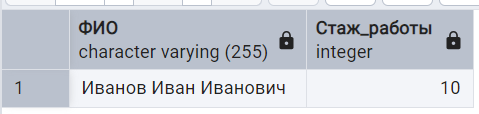
JOIN "Отдел" О ON С."ID\_Сотрудника" = О."ID\_Сотрудника"

WHERE О."Название\_отдела" IN ('Отдел разработки', 'Отдел тестирования')

) AS Subquery

WHERE "Стаж\_работы" > 3;

Таблица:



# Задание №11:

11. Напишите пример связанного подзапроса.

**Решение:**

Связанный подзапрос — это подзапрос, который ссылается на столбцы из внешнего запроса.

Запрос, выбирающий всех сотрудников, которые работают в отделах, где средняя зарплата сотрудников выше средней зарплаты по всей компании.

Программный код:

SELECT "ФИО", "Должность", "Отдел"."Название\_отдела"

FROM "Сотрудник"

JOIN "Отдел" ON "Сотрудник"."ID\_Сотрудника" = "Отдел"."ID\_Сотрудника"

WHERE (

SELECT AVG("Стаж\_работы")

FROM "Сотрудник" AS sub

WHERE sub."ID\_Сотрудника" = "Отдел"."ID\_Сотрудника"

) > (

SELECT AVG("Стаж\_работы")

FROM "Сотрудник"

);

Таблица:

