**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**

по Лабораторной Работе № 2

по дисциплине «**Проектирование и реализация баз данных**»

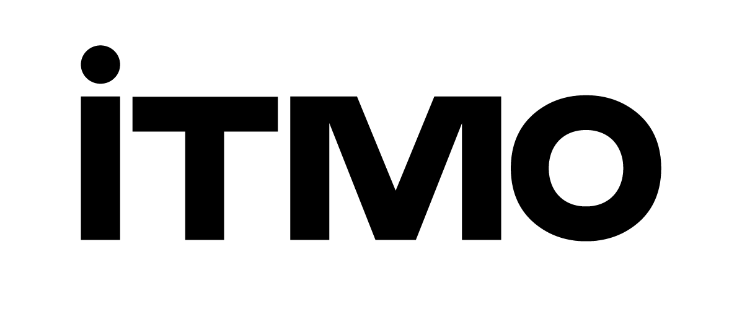
Вариант 2

Автор: Барышников Владимир Романович

Факультет: ФИКТ

Группа: K32401

Преподаватель: Говорова Марина Михайловна



Санкт-Петербург

**Создание запросов**

**1. Список дисциплин для заданной группы с указанием дат сдачи и фамилий преподавателей:**

SELECT D.discipline\_name AS Discipline, ES.exam\_date AS ExamDate, T.lastname AS TeacherLastName

FROM scheme1234.exam\_schedule ES

JOIN scheme1234.discipline D ON ES.discipline\_name = D.discipline\_name

JOIN scheme1234.exam E ON ES.exam\_code = E.exam\_code

JOIN scheme1234.teacher T ON E.teacher\_code = T.teacher\_code

JOIN scheme1234.studying\_student SS ON E.sstudent\_code = SS.sstudent\_code

JOIN scheme1234.student\_class SC ON SS.group\_code = SC.group\_code

WHERE SC.group\_number = '14A’;

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

**2. Список студентов, получивших двойки на первой попытке с указанием фамилии преподавателя, которым они должны пересдать экзамен:**

SELECT S.fullname AS StudentName, T.lastname AS TeacherLastName

FROM scheme1234.exam E

JOIN scheme1234.teacher T ON E.teacher\_code = T.teacher\_code

JOIN scheme1234.studying\_student SS ON E.sstudent\_code = SS.sstudent\_code

JOIN scheme1234.student S ON SS.record\_book\_number = S.record\_book\_number

WHERE E.mark = '2' AND E.try\_number = '1';

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

**3. Фамилии студентов, получивших оценки по дисциплине, которые выше среднего балла по этой дисциплине:**

SET search\_path TO scheme1234;

SELECT s.fullname AS StudentName, ss.group\_code, e.mark, es.exam\_date

FROM scheme1234.exam\_schedule es

JOIN scheme1234.exam e ON es.exam\_id = e.exam\_id

JOIN scheme1234.studying\_student ss ON e.sstudent\_code = ss.sstudent\_code

JOIN scheme1234.student s ON ss.record\_book\_number = s.record\_book\_number

WHERE (es.discipline\_name, es.exam\_date) IN (

SELECT discipline\_name, MAX(exam\_date)

FROM scheme1234.exam\_schedule

GROUP BY discipline\_name

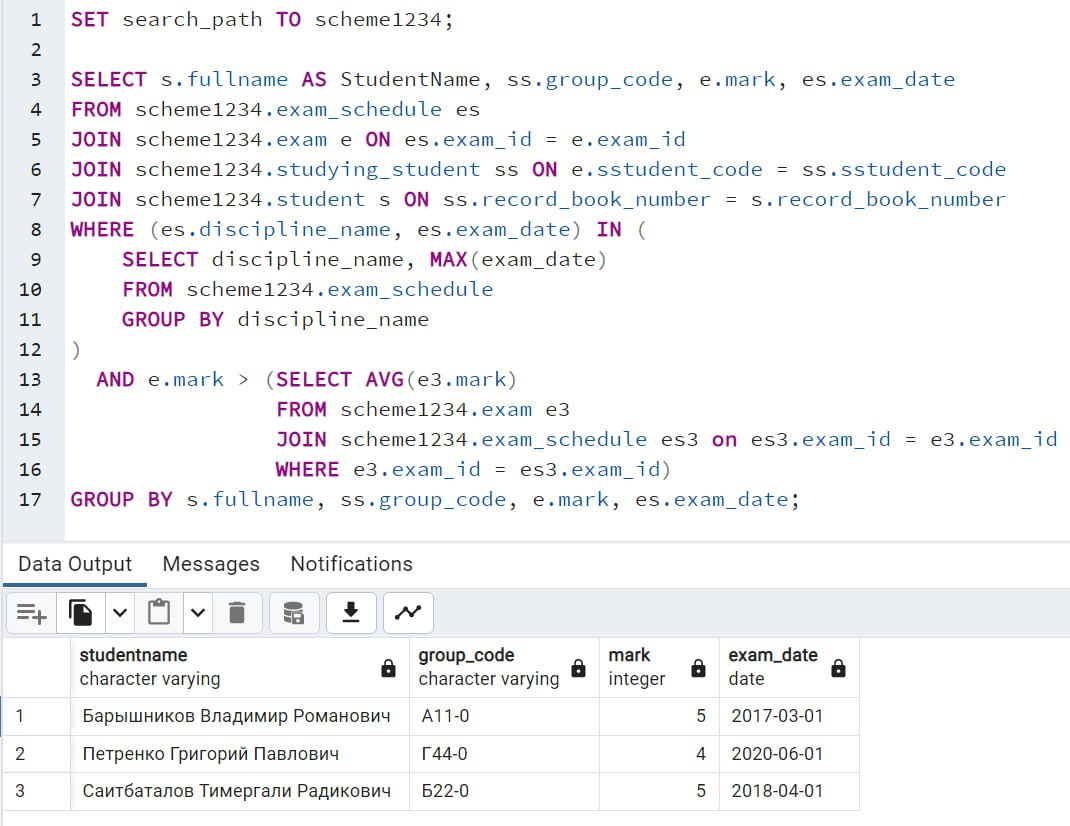
)

AND e.mark > (SELECT AVG(e3.mark)

FROM scheme1234.exam e3

JOIN scheme1234.exam\_schedule es3 on es3.exam\_id = e3.exam\_id

WHERE e3.exam\_id = es3.exam\_id)

GROUP BY s.fullname, ss.group\_code, e.mark, es.exam\_date;

**4. Рейтинговый список групп по заданному направлению по результатам сдачи сессии, упорядоченный по убыванию:**

SELECT SC.group\_number, AVG(CAST(E.mark AS decimal)) AS AvgMark

FROM scheme1234.exam E

JOIN scheme1234.studying\_student SS ON E.sstudent\_code = SS.sstudent\_code

JOIN scheme1234.student\_class SC ON SS.group\_code = SC.group\_code

GROUP BY SC.group\_number

ORDER BY AvgMark DESC;

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

**5. Список студентов, которые назначены на стипендию исходя из того, что их средний балл больше 3:**

SELECT SC.group\_number, S.fullname AS StudentName

FROM scheme1234.student\_class SC

JOIN scheme1234.studying\_student SS ON SC.group\_code = SS.group\_code

JOIN scheme1234.student S ON SS.record\_book\_number = S.record\_book\_number

WHERE SS.sstudent\_code NOT IN (

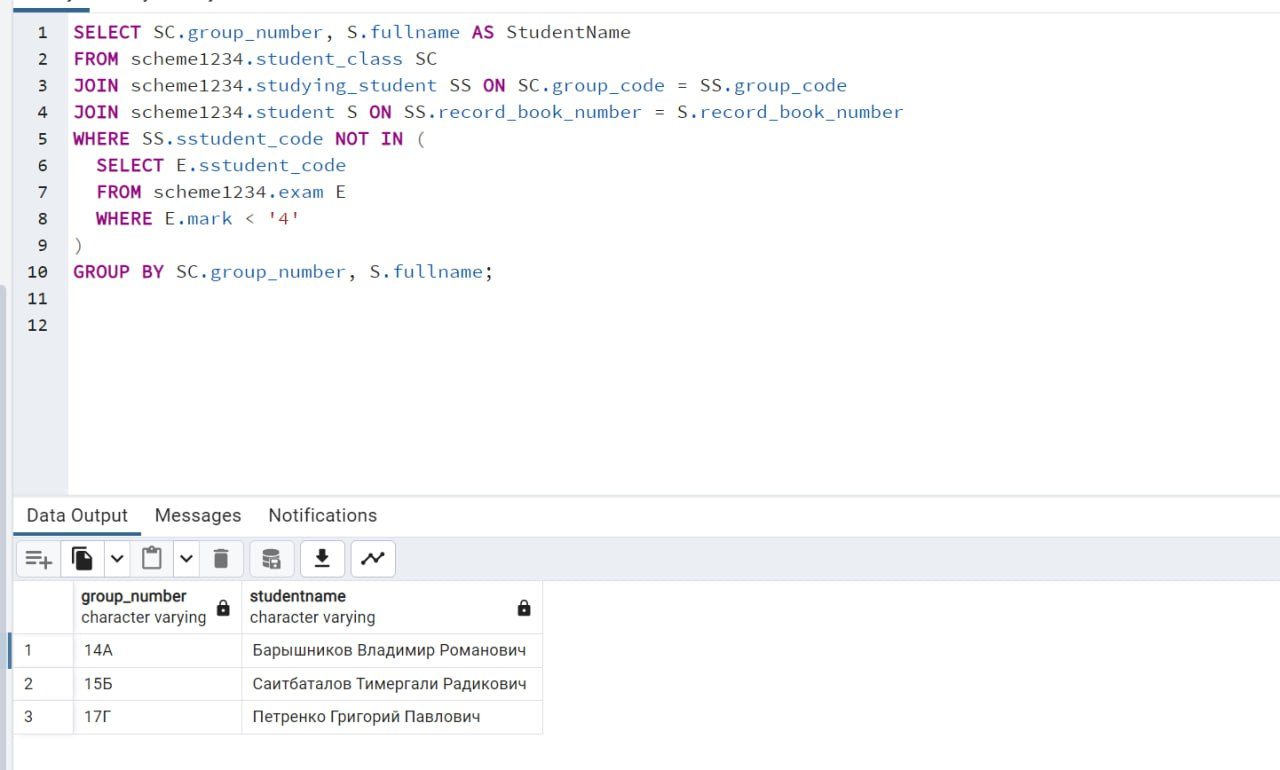
SELECT E.sstudent\_code

FROM scheme1234.exam E

WHERE E.mark < '4'

)

GROUP BY SC.group\_number, S.fullname;



**6. Список студентов, сдавших все положенные экзамены:**

SELECT distinct S.fullname AS StudentName

FROM scheme1234.student S

JOIN scheme1234.studying\_student SS ON

S.record\_book\_number = SS.record\_book\_number

JOIN scheme1234.exam E ON E.sstudent\_code = SS.sstudent\_code

JOIN scheme1234.exam\_schedule ES ON E.exam\_id = ES.exam\_id

WHERE NOT EXISTS (

SELECT E2.sstudent\_code

FROM scheme1234.exam E2

WHERE E2.sstudent\_code = SS.sstudent\_code AND E2.try\_number = '3' AND E2.mark = '2'

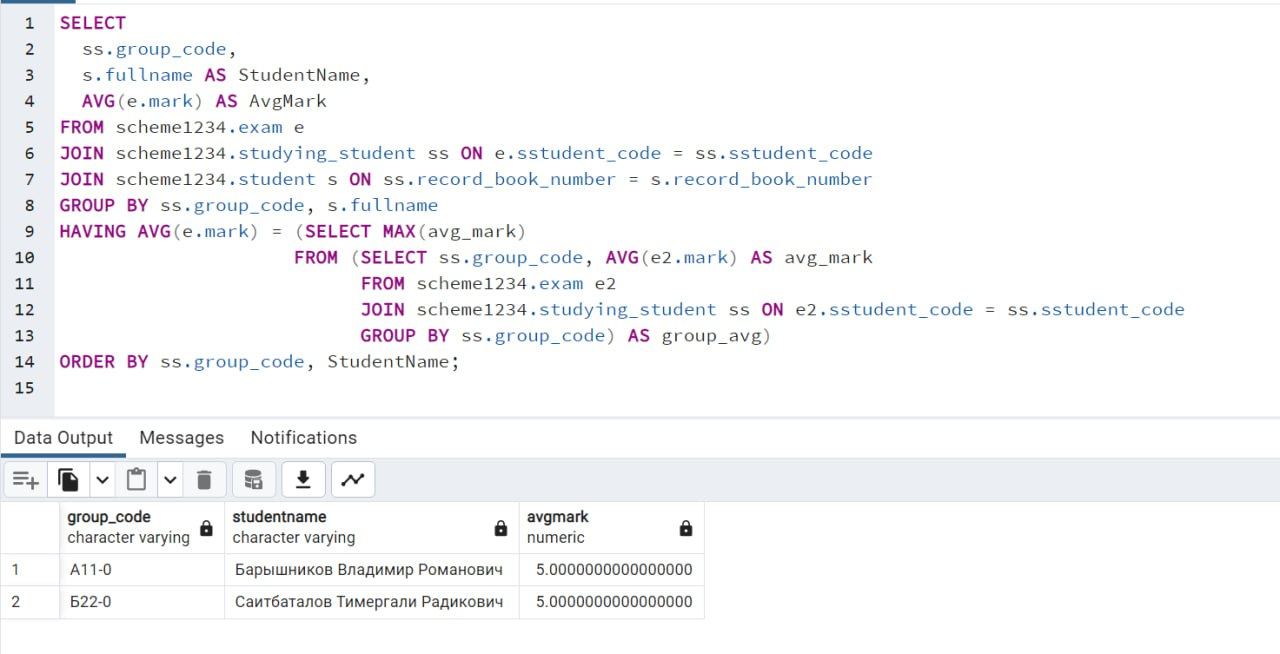
)

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

**7. Список студентов, получивших максимальный средний балл в своей группе:**

SELECT  
  ss.group\_code,  
  s.fullname AS StudentName,  
  AVG(e.mark) AS AvgMark  
FROM scheme1234.exam e  
JOIN scheme1234.studying\_student ss ON e.sstudent\_code = ss.sstudent\_code  
JOIN scheme1234.student s ON ss.record\_book\_number = s.record\_book\_number  
GROUP BY ss.group\_code, s.fullname  
HAVING AVG(e.mark) = (SELECT MAX(avg\_mark)  
                     FROM (SELECT ss.group\_code, AVG(e2.mark) AS avg\_mark  
                           FROM scheme1234.exam e2  
                           JOIN scheme1234.studying\_student ss ON e2.sstudent\_code = ss.sstudent\_code  
                           GROUP BY ss.group\_code) AS group\_avg)  
ORDER BY ss.group\_code, StudentName;

****

**Задание 3. Создание представлений**

**1. Список студентов, получивших двойки на первой попытке с указанием фамилии преподавателя, которым они должны пересдать экзамен:**

CREATE VIEW StudentsWithGrade2OnFirstAttempt AS

SELECT S.fullname AS StudentName, T.lastname AS TeacherLastName

FROM scheme1234.exam E

JOIN scheme1234.teacher T ON E.teacher\_code = T.teacher\_code

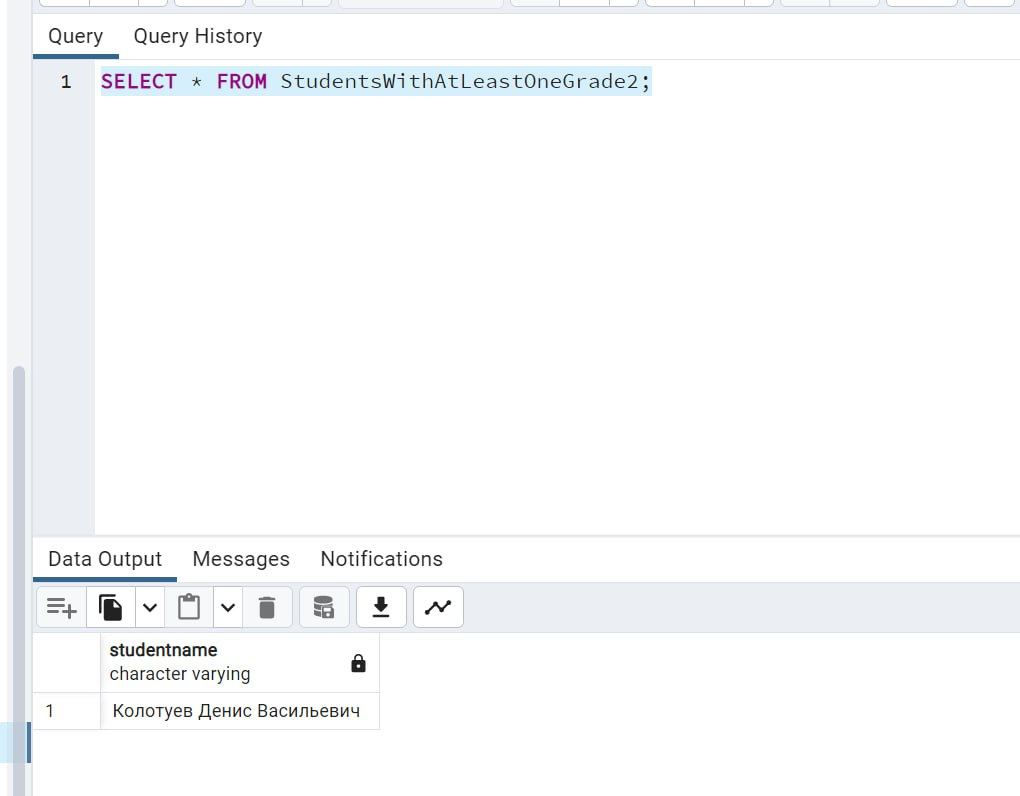
JOIN scheme1234.studying\_student SS ON E.sstudent\_code = SS.sstudent\_code

JOIN scheme1234.student S ON SS.record\_book\_number = S.record\_book\_number

WHERE E.mark = '2' AND E.try\_number = '1';

Изображение выглядит как текст, Шрифт, число, снимок экрана

Автоматически созданное описание



**2. Данных о студентах при получении ими хотя бы одной оценки 2 (после 2-й попытки):**

CREATE VIEW StudentsWithAtLeastOneGrade2 AS

SELECT DISTINCT S.fullname AS StudentName

FROM scheme1234.exam E

JOIN scheme1234.studying\_student SS ON E.sstudent\_code = SS.sstudent\_code

JOIN scheme1234.student S ON SS.record\_book\_number = S.record\_book\_number

WHERE E.mark = '2' AND E.try\_number > '3';

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

**INSERT, DELETE, UPDATE**

1. Вставить несколько записей о преподавателях в таблицу, обозначив все свойства.

**Команда:**

INSERT INTO teacher (lastname,teacher\_code,surname,teachers\_name,title)

VALUES ('Теремов',51,'Евгеньевич','Алексей','декан программирования'),

('Денисов',52,'Иванович','Олег','эксперт по БД'),('Отшушаренко',53,'Анатольевич','Павел','Ректор'),

('Тарасова',54,'Генадьевна','Татьяна','Лучший\_дизайнер\_СПб'),('Вдыбенко',56,'Павловна','Ольга',

( SELECT title FROM teacher WHERE teacher\_code = 22))

Изображение выглядит как текст, Шрифт, линия, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

2. Изменение даты экзамена.

Команда:

UPDATE scheme1234.exam\_schedule ES

SET exam\_date = '2023-09-20'

WHERE classroom\_code = 222222

AND ES.exam\_id IN (

SELECT exam\_id

FROM scheme1234.exam

WHERE mark >= '4'

);

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

3. Удалить все записи о студентах, которые не принадлежат ни к одной группе.

**Команда:**

DELETE FROM scheme1234.student

WHERE record\_book\_number NOT IN (

SELECT SS.record\_book\_number

FROM scheme12.studying\_student SS

);

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

**Индексы**

Простой индекс для ускорения поиска даты рождения студента:

CREATE INDEX idx\_student\_year\_of\_birth ON student (year\_of\_birth);

***До:***

113ms

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

***После:***

***71ms***

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Ускорили в ***1.59раз***

**Удаление индексов:**

DROP INDEX idx\_teacher\_practice;

DROP INDEX idx\_exam\_schedule\_info;

**Составной индекс:**

Индекс для ускорения запросов по расписанию экзаменов:

CREATE INDEX idx\_exam\_schedule\_info

ON exam\_schedule (exam\_date, classroom\_code, discipline\_name);

**До:**

106ms

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

**После:**

70ms

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

**Вывод**

В рамках работы были выполнены операции модификации данных, такие как UPDATE, INSERT и DELETE. Также были созданы простые и составные индексы, улучшающие производительность запросов к базе данных. В результате выполнения лабораторной работы были приобретены навыки работы с SQL для формирования запросов, обработки данных и оптимизации их производительности.