**ТРЗБД.**

**Задание 1.**

1. Запрос на выборку.

Выбрать фамилии студентов родившихся в промежутке с 2004 по 2006 год рождения.

SELECT ФАМИЛИЯ

FROM СТУДЕНТ

WHERE ГОД РОЖДЕНИЯ= 2004 BETWEEN 2006;

1. Подзапросы.

Нужно узнать оценки студента по фамилии Белева, не зная его номер

SELECT\*

FROM EXAM\_MARX

WHERE STUDENT\_ID=(SELECT STUDENT ID

FROM STUDENT

WHERE SURNAME=»Белева»);

1. Операция объединения таблиц.

Выбрать сведения обо всех предметах обучения, по которым проводился экзамен 06 ноября 2023 г.

SELECT DISTINCT SU.SUBJ\_ID, SUBJ\_NAME, HOUR, SEMESTER

FROM SUBJECT FIRST, EXAM\_MARKS SECOND

WHERE FIRST.SUBJ\_ID = SECOND.SUBJ\_ID

AND SECOND.EXAM\_DATE = '06/11/2023';

1. Подзапросы, связывающие таблицу со своей собственной копией.

Например, надо найти идентификаторы, фамилии и стипендии студентов, получающих стипендию выше средней на курсе, на котором они учатся.

SELECT DISTINCT STUDENT\_ID, SURNAME, STIPEND

FROM STUDENT El

WHERE STIPEND >

(SELECT AVG(STIPEND)

FROM STUDENT E2

WHERE El.KURS = E2.KURS);

1. Связанные подзапросы в HAVING.

По данным из таблицы EXAM\_MARKS определить сумму полученных студентами оценок и исключить те дни, когда число студентов, сдававших в течение дня экзамены, было меньше 5.

SELECT EXAM\_DATE, SUM(MARK)

FROM EXAM\_MARKS A

GROUP BY EXAM\_DATE

HAVING 5 < (SELECT COUNT(MARK)

FROM EXAM\_MARKS В

WHERE A.EXAM\_DATE = В.EXAM\_DATE);

1. Использовании операторов EXIST.

Нужно извлечь из таблицы EXAM\_MARKS данные о студентах, получивших хотя бы одну неудовлетворительную оценку.

SELECT DISTINCT STUDENT\_ID

FROM EXAM\_MARKS A

WHERE EXISTS

(SELECT \* FROM EXAM\_MARKS В

WHERE MARK < 3

AND B.STUDENT\_ID = A.STUDENT\_ID);

1. Операторы сравнения с множеством значений All.

Выбирать данные о названиях всех университетов с рейтингом более высоким, чем рейтинг любого университета Санкт-Петербурга.

SELECT \*

FROM UNIVERSITY

WHERE RATING > ALL

(SELECT RATING

FROM UNIVERSITY

WHERE CITY = ' Санкт-Петербурга ');

1. Особенности применения операторов ANY, EXISTS для NULL-значений.

Найти все данные об университетах, рейтинг которых меньше рейтинга любого университета в Санкт-Петербурге.

SELECT \* FROM UNIVERSITY

WHERE RATING < ANY

(SELECT RATING FROM UNIVERSITY

WHERE CITY = ' Санкт-Петербург');

SELECT \* FROM UNIVERSITY A

WHERE NOT EXISTS

(SELECT \* FROM UNIVERSITY В

WHERE A.RATING >= B.RATING

AND B.CITY = ' Санкт-Петербург');

1. Оператор объединения UNION.

Получить в одной таблице фамилии и идентификаторы студентов и преподавателей из Санкт-Петербурга, можно использовать следующий запрос.

SELECT 'Студент ', SURNAME, STUDENT\_ID

FROM STUDENT

WHERE CITY = ' Санкт-Петербург'

UNION

SELECT 'Преподаватель', SURNAME, LECTURER\_ID

FROM LECTURER

WHERE CITY = ' Санкт-Петербург';

1. Соединение таблиц с использованием JOIN.

Получить фамилии студентов и для каждого студента названия университетов, расположенных в городе, где живет студент.

SELECT STUDENT.SURNAME, UNIVERSITY.UNIV\_NAME, STUDENT.CITY

FROM STUDENT INNER JOIN UNIVERSITY

ON STUDENT.CITY = UNIVERSITY.CITY;

**Задание 2.**

1. Петров В.В. "Базы данных. Учебник и практикум". 2018.

2. Конноли Т., Бегг К. "Системы баз данных: проектирование, реализация и управление". 2015.

3. Шкапоуцкая Н.Ю. "Физическое проектирование баз данных". 2019.

4. "SQL: A Beginner's Guide" by Andy Oppel. 2014.

5. Мельников В.А. "Проектирование баз данных". 2017.

6. Матвеев А.В., Проскуряков В.И. "Администрирование баз данных PostgreSQL и MySQL". 2016.

7. Григорьев В., Кореневский А. "Высоконагруженные системы. Проектирование и администрирование". 2017.

8. Занин С.В., Жукова Ю.В. "Администрирование СУБД Microsoft SQL Server 2017". 2018.

9. Лондон К., Жук Б., Хит Ж. "Oracle Database 12c. SQL. Fundamentals". 2015.

10. Кауп Д. "Проектирование реляционных баз данных с использованием Microsoft Access 2000. Методическое пособие". 2021.

11. Лариэн Дж., Бэйтс М., Уилкерсон Р. "SQL. Программирование для профессионалов". 2016.

12. "Essential SQL: Fast Genie: Practical Guide for SQL Server" by Seyed Ali Tabatabaei Nasab. 2018.

13. Шапошников А. "PostgreSQL. Администрирование и разработка". 2019.

14. Павлов А.А., Петрачков И.А. "PostgreSQL. Базы данных". 2021.

15. Карнеги Н. "Изучаем MySQL". 2015.