Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

По дисциплине: “Теория баз данных”

Лабораторная работа №5

“РАЗРАБОТКА СЛОЖНЫХ ЗАПРОСОВ К БАЗЕ ДАННЫХ.

ЗАПРОСЫ НА ОСНОВЕ НЕСКОЛЬКИХ ТАБЛИЦ.

КОРРЕЛИРОВАННЫЕ ВЛОЖЕННЫЕ ПОДЗАПРОСЫ”

Выполнил:

ст.гр. ИС/б-22

Долженко И.А.

Проверила:

Лебедева М.А.

Севастополь

2019

1 ЦЕЛЬ РАБОТЫ

Выполнение лабораторной работы преследует следующие цели:

* изучить способы получения информации из нескольких таблиц;
* записать запросы, демонстрирующие выборки из нескольких таблиц с использованием оператора JOIN и без него;
* изучить способы выполнения и принцип действия рекурсивных запросов;
* научится использовать вложенные подзапросы;
* ознакомиться с возможностями построения вложенных коррелированных подзапросов с применением кванторов;
* ознакомиться с возможностью формирования отчетов в клиентском приложении.

2 ПОСТАНОВКА ЗАДАЧИ

1. Проанализировать созданную в л/р №4 базу данных.
2. Сформулировать и выполнить с помощью SQL-редактора PhpMyAdmin запросы, соединяющие две таблицы с помощью JOIN и без него.
3. Сформулировать и выполнить с помощью SQL-редактора PhpMyAdmin запросы, соединяющие более чем две таблицы с помощью JOIN и без него.
4. Продемонстрировать с помощью SQL-редактора PhpMyAdmin следующие возможности SQL:

* использование псевдонимов на примере рекурсивного запроса;
* запрос с подзапросом;
* использование агрегатных функций в подзапросе;
* подзапросы, возвращающие единственное и множественные значения;
* подзапросы, использующие вычисление;
* использование подзапросов в HAVING;

1. Ознакомится с принципом и продемонстрировать работу коррелированных подзапросов:

* привести пример соединения таблицы со своей копией;
* привести пример коррелированного запроса, использующего две разные таблицы;
* привести пример запроса с оператором EXIST;
* привести пример запроса с оператором ALL;
* привести пример запроса с оператором ANY.

1. Разработать и выполнить с помощью SQL-редактора PhpMyAdmin обязательные запросы, заданные вариантом (Таблица 1).

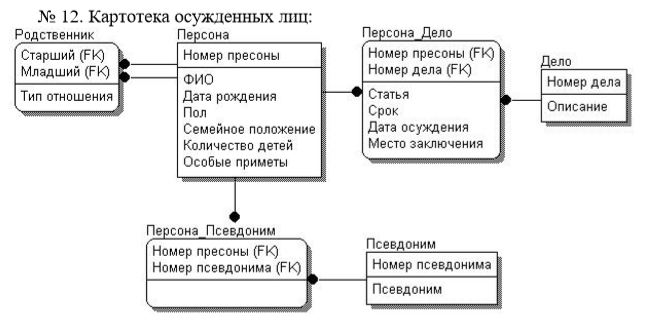


Рисунок 1 – Схема БД по варианту

Таблица 1 – Вариант задания

|  |  |
| --- | --- |
| № Варианта | Запросы к базе данных |
| 3 | 1. Вывести всех осужденных, проходящих по делу №26.  2. Вывести всех осужденных, которые не имеют псевдонима.  3. Вывести Ф.И.О. осужденного на максимальный срок |

3 ХОД РАБОТЫ

1. Сформулировать и выполнить с помощью SQL-редактора PhpMyAdmin запросы, соединяющие две таблицы с помощью JOIN и без него.

Исходные таблицы:



Рисунок 2 – Таблица Персона



Рисунок 3 – Таблица Дело

Вывести записи, содержащие: номер персоны, ФИО, дату рождения, пол и описание дела, у которых номер персоны равен номеру дела.

Использование JOIN:

SELECT `Номер персоны`, `ФИО`, `Дата рождения`, `Пол`, `Дело`.`Описание` FROM `персона` JOIN `дело` ON `персона`.`Номер персоны` = `Дело`.`Номер дела`

Без использования JOIN:

SELECT `Номер персоны`, `ФИО`, `Дата рождения`, `Пол`, `Дело`.`Описание` FROM `персона`, `дело` WHERE `персона`.`Номер персоны` = `Дело`.`Номер дела`



Рисунок 4 – Результат соединения двух таблиц

2. Сформулировать и выполнить с помощью SQL-редактора PhpMyAdmin запросы, соединяющие более чем две таблицы с помощью JOIN и без него.

Исходные таблицы (рис. 2,3 и рис. 5):

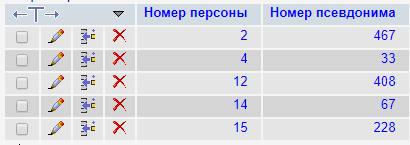


Рисунок 5 – Таблица Персона\_псевдоним

Вывести записи, содержащие: номер персоны, ФИО, описание дела, номер псевдонима у которых номер персоны равен номеру дела и номер персоны из таблицы персона\_псевдоним равен номеру персоны в таблице персона.

Использование JOIN:

SELECT `Персона`.`Номер персоны`, `ФИО`, `Дело`.`Описание`, `Персона\_псевдоним`.`Номер псевдонима` FROM `Персона` JOIN `Дело` ON `Персона`.`Номер персоны` = `Дело`.`Номер дела` INNER JOIN `Персона\_псевдоним` ON `Персона\_псевдоним`.`Номер персоны` = `Персона`.`Номер персоны`

Без использования JOIN:

SELECT `Персона`.`Номер персоны`, `ФИО`, `Дело`.`Описание`, `Персона\_псевдоним`.`Номер псевдонима` FROM `Персона`, `Дело`, `Персона\_псевдоним` WHERE `Персона`.`Номер персоны` = `Дело`.`Номер дела` && `Персона\_псевдоним`.`Номер персоны` = `Персона`.`Номер персоны`

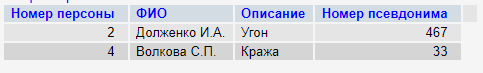


Рисунок 6 – Результат соединения трех таблиц

3. Использование псевдонимов на примере рекурсивного запроса

SELECT one.`Номер персоны`, two.`Номер персоны`, one.`Количество детей` FROM `Персона` AS one, `Персона` AS two WHERE one.`Количество детей` = two.`Количество детей` AND one.`Номер персоны` < two.`Номер персоны`



Рисунок 7 – Результат использование псевдонимов

4. Запрос с подзапросом + агрегатная функция

Выбрать запись где номер персоны равен максимальному номеру дела в таблице Персона\_дело

SELECT \* FROM `Персона` WHERE `Номер персоны` = (SELECT MAX(`Номер дела`) FROM `Персона\_дело`)



Рисунок 8 – Таблица Персона\_дело



Рисунок 9 – Результат использования подзапроса с агрегатной функцией

5. Подзапросы, использующие вычисление

Вывод списка персон, имеющих Срок меньше 3, если его поделить на 5.

SELECT \* FROM `Персона` WHERE EXISTS( SELECT \* FROM `Персона\_дело` WHERE `Персона\_дело`.`Номер персоны` = `Персона`.`Номер персоны` AND `Персона\_дело`.`Срок`/5 < 3)

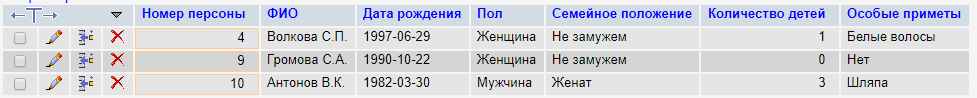


Рисунок 10 – Результат использование подзапросов с вычислениями

6. Использование подзапросов с HAVING

Вывод списка персон, номер которых больше чем среднее число номеров персон в этой таблице.

SELECT `Номер персоны`, `ФИО` FROM `Персона` HAVING `Номер персоны` > (SELECT AVG(`Номер персоны`) FROM `Персона`)

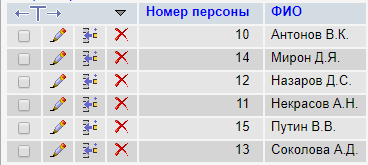


Рисунок 11 – Результат использование подзапросов с HAVING

7. Привести пример коррелированного запроса, использующего две разные таблицы.

Вывод список персон, номер дела которых больше 5.

SELECT DISTINCT `Персона`.`Номер персоны`, `Персона`.`ФИО`, `Персона`.`Пол`, `Персона\_дело`.`Номер дела`, `Персона\_дело`.`Статья` FROM `Персона`, `Персона\_дело` WHERE `Персона`.`Номер персоны` = `Персона\_дело`.`Номер персоны` AND `Персона\_дело`.`Номер дела` > 5



Рисунок 12 – Пример коррелированного запроса

8. Привести пример запроса с оператором ALL.

Вывести всех персон, у которых количество детей выше чем у всех персон, чей номер < 4.

SELECT \* FROM `Персона` WHERE `Количество детей` > ALL (SELECT `Количество детей` FROM `Персона` WHERE `Номер персоны` < 4)



Рисунок 13 – Пример использования оператора ALL

9. Привести пример запроса с оператором ANY.

Вывод персон, у которых есть записи о номерах персон в таблице Персона\_псевдоним.

SELECT \* FROM `Персона` WHERE `Персона`.`Номер персоны` = ANY(SELECT `Персона\_псевдоним`.`Номер персоны` FROM `Персона\_псевдоним`)



Рисунок 14 – Пример использования оператора ANY

**Задания по варианту**

10. Вывести всех осужденных, проходящих по делу №26.

SELECT `Персона`.`Номер персоны`, `Персона`.`ФИО`, `Персона`.`Дата рождения`, `Персона`.`Пол`, `Персона`.`Номер дела`

FROM `Персона`

JOIN `Персона\_дело` ON `Персона\_дело`.`Номер персоны` = `Персона`.`Номер персоны` && `Номер дела` = 26



Рисунок 15 – Таблица Персона\_дело для проверки результата



Рисунок 16 – Результат вывода осужденных, проходящих по делу №26

11. Вывести всех осужденных, которые не имеют псевдонима.

SELECT \*

FROM `Персона`

WHERE `Персона`.`Номер персоны` != ALL (

SELECT `Персона\_псевдоним`.`Номер персоны`

FROM `Персона\_псевдоним`

)

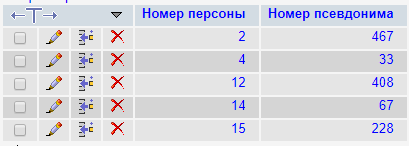


Рисунок 17 – Таблица Персона\_псевдоним



Рисунок 18 – Таблица Персона



Рисунок 19 – Результат поиска персон, не имеющих псевдонима

12. Вывести Ф.И.О. осужденного на максимальный срок.

SELECT `Персона`.`ФИО`, `Персона\_дело`.`Срок`

FROM `Персона`

JOIN `Персона\_дело` ON `Персона\_дело`.`Номер персоны` = `Персона`.`Номер персоны` && `Персона\_дело`.`Срок` >= ALL (

SELECT `Персона\_дело`.`Срок`

FROM `Персона\_дело`

)



Рисунок 20 – Результат поиска персоны, осужденной на максимальный срок

ВЫВОД

В ходе выполнения лабораторной работы были изучены способы получения информации из нескольких таблиц; записаны запросы, демонстрирующие выборки из нескольких таблиц с использованием оператора JOIN и без него; изучены способы выполнения и принцип действия рекурсивных запросов; освоены навыки использования вложенных подзапросов.