ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | Н.В Путилова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №6  Разработка SQL запросов: запросы с подзапросами |
| **по дисциплине: Проектирование баз данных** |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 4134к |  |  |  | Костяков Н.А. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2023

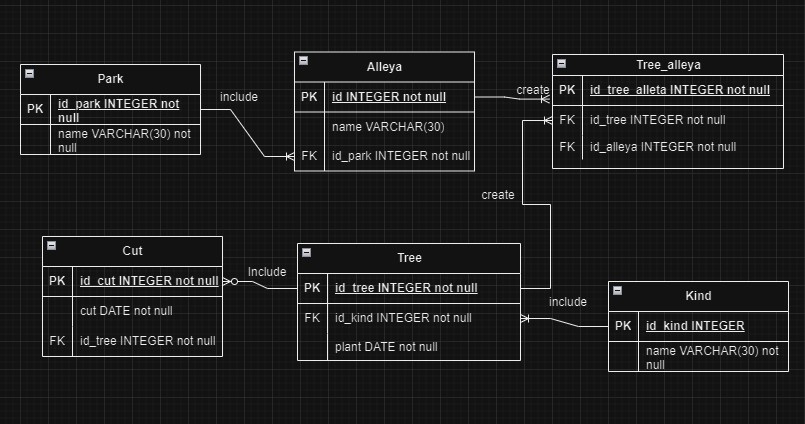
Задание

По аналогии с примерами, приведенными в п. 1 реализовать запросы г) .. ж), указанные в варианте задания. Один из запросов на максимум/минимум реализовать с помощью директивы all Запрос на «все» (реляционное деление) реализовать с помощью 2 not exists Запросы на разность реализовать в 3 вариантах: Not in,except(MySQL не поддерживает, поэтому только синтаксис), с использованием левого/правого соединения

д. породы, деревья которой обрезали меньше всего

е. порода дерева, встречающаяся на всех аллеях заданного парка

ж. аллея, на которой растут дубы, но нет лип



Скрипты

Г) г. аллея, на которой растут деревья, которое было посажено позже всех

select alleya.name from alleya

join tree\_alleya on tree\_alleya.id\_alleya = alleya.id\_alleya

join tree on tree.id\_tree = tree\_alleya.id\_tree

where tree.plant >= all(select plant from tree );

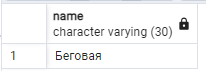
////////////////

select alleya.name from alleya

join tree\_alleya on tree\_alleya.id\_alleya = alleya.id\_alleya

join tree on tree.id\_tree = tree\_alleya.id\_tree

where tree.plant in (select max(tree.plant) from tree);



д) . породы, деревья которой обрезали меньше всего

SELECT kind.name, COUNT(cut.id\_tree) AS cut\_count

FROM kind

JOIN tree ON tree.id\_kind = kind.id\_kind

LEFT JOIN cut ON cut.id\_tree = tree.id\_tree

GROUP BY kind.name

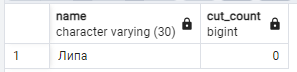
having COUNT(cut.id\_tree) <= all (select COUNT(cut.id\_tree)

FROM kind

JOIN tree ON tree.id\_kind = kind.id\_kind

LEFT JOIN cut ON cut.id\_tree = tree.id\_tree GROUP BY kind.name)

ORDER BY cut\_count ASC;



е)

порода дерева, встречающаяся на всех аллеях заданного парка

количество аллей на которых встерчается порода дерева

=посчитать колво аллей

select distinct kind.name from kind

join tree on tree.id\_kind = kind.id\_kind

where not exists

(Select \* from alleya

join park on park.id\_park=alleya.id\_park

where park.name='Космический' and not exists

(select \* from tree as tr

join kind as ki on ki.id\_kind = tr.id\_kind

join tree\_alleya as t\_a on t\_a.id\_tree = tr.id\_tree

join park on park.id\_park = alleya.id\_park

where t\_a.id\_alleya = alleya.id\_alleya

and tr.id\_kind = tree.id\_kind

));

SELECT kind.name AS number\_of\_alleys

FROM kind

JOIN tree ON tree.id\_kind = kind.id\_kind

JOIN tree\_alleya ON tree\_alleya.id\_tree = tree.id\_tree

JOIN alleya ON alleya.id\_alleya = tree\_alleya.id\_alleya

JOIN park ON park.id\_park = alleya.id\_park

WHERE park.name = 'Космический'

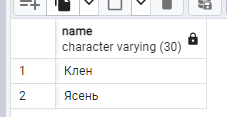
GROUP BY kind.name

HAVING COUNT(DISTINCT alleya.id\_alleya) =

(SELECT COUNT(DISTINCT alleya.id\_alleya)

from alleya join park on park.id\_park = alleya.id\_park

where park.name = 'Космический' );



ж1) аллея, на которой растут дубы, но нет лип

select alleya.name from alleya

join tree\_alleya as t\_a on t\_a.id\_alleya= alleya.id\_alleya

join tree on tree.id\_tree =t\_a.id\_tree

join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Дуб'

except

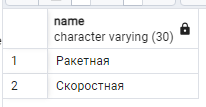
select alleya.name from alleya

join tree\_alleya as t\_a on t\_a.id\_alleya= alleya.id\_alleya

join tree on tree.id\_tree =t\_a.id\_tree

join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Липа';



2)select distinct alleya.name from alleya

join tree\_alleya as t\_a on t\_a.id\_alleya= alleya.id\_alleya

join tree on tree.id\_tree =t\_a.id\_tree

join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Дуб'

and alleya.id\_alleya NOT IN (

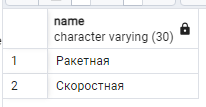
select alleya.id\_alleya from alleya

join tree\_alleya as t\_a on t\_a.id\_alleya= alleya.id\_alleya

join tree on tree.id\_tree =t\_a.id\_tree

join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Липа');



3)select distinct alleya.name from alleya

join tree\_alleya as t\_a on t\_a.id\_alleya= alleya.id\_alleya

join tree on tree.id\_tree =t\_a.id\_tree

join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Дуб'

and not exists (

select \* from alleya as a2

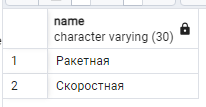
join tree\_alleya as t\_a on t\_a.id\_alleya= a2.id\_alleya

join tree on tree.id\_tree =t\_a.id\_tree

join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Липа'

and alleya.id\_alleya = a2.id\_alleya);



4) select distinct alleya.name from alleya

join tree\_alleya on tree\_alleya.id\_alleya = alleya.id\_alleya

join tree on tree.id\_tree = tree\_alleya.id\_tree

join kind on kind.id\_kind = tree.id\_kind

left join

(select \* from alleya2 as a2

inner join tree\_alleya on tree\_alleya.id\_alleya = a2.id\_a inner

join tree on tree.id\_tree = tree\_alleya.id\_tree

inner join kind on kind.id\_kind = tree.id\_kind

where kind.name = 'Липа') as qq

on alleya.id\_alleya = qq.id\_a

where kind.name = 'Дуб'

and qq.id\_a is NULL;

