1. Выбрать поставщиков, поставляющих детали, поставляемые поставщиками, проживающими в Лондоне.

select distinct t.n\_post   
from spj t   
where t.n\_det in(   
 select distinct n\_det   
 from spj   
 join s on spj.n\_post = s.n\_post   
 where [s.town](https://vk.com/away.php?to=http%3A%2F%2Fs.town&cc_key=" \t "_blank)='Лондон'   
 ) order by 1

1. Вывести без повторений пары городов таких, что была поставка детали из первого города для изделия во втором городе. Упорядочить список по алфавиту.

select distinct [p.town](https://vk.com/away.php?to=http%3A%2F%2Fp.town&cc_key=" \t "_blank), [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
from spj   
join p on p.n\_det = spj.n\_det   
join j on j.n\_izd = spj.n\_izd   
where [p.town](https://vk.com/away.php?to=http%3A%2F%2Fp.town&cc_key=" \t "_blank) != [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
order by [p.town](https://vk.com/away.php?to=http%3A%2F%2Fp.town&cc_key=" \t "_blank)

1. Получить список городов, в которые выполнили поставки ТОЛЬКО поставщики, поставлявшие голубые детали.

select distinct [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
from spj t   
join j on j.n\_izd = t.n\_izd   
where t.n\_post in (   
 select distinct spj.n\_post   
 from spj   
 join p on p.n\_det = spj.n\_det   
 where p.cvet = 'Голубой'   
 )   
except   
select distinct [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
from spj t   
join j on j.n\_izd = t.n\_izd   
where not t.n\_post in (   
 select distinct spj.n\_post   
 from spj   
 join p on p.n\_det = spj.n\_det   
 where p.cvet = 'Голубой'   
 )

4. Вывести полный список городов и для каждого города найти суммарное количество деталей красного цвета, которые были в него поставлены. Города в списке должны быть ВСЕ. Список должен быть упорядочен по алфавиту.

select [town.town](https://vk.com/away.php?to=http%3A%2F%2Ftown.town&cc_key=" \t "_blank), pt.skol   
from ( select town   
 from p   
 union   
 select town   
 from s   
 union   
 select town   
 from j   
 ) town   
left join ( select [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank), sum(spj.kol) skol   
 from spj   
 join p on spj.n\_det = p.n\_det   
 join j on spj.n\_izd = j.n\_izd   
 where p.cvet = 'Красный'   
 group by [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
 ) pt on [pt.town](https://vk.com/away.php?to=http%3A%2F%2Fpt.town&cc_key=" \t "_blank) = [town.town](https://vk.com/away.php?to=http%3A%2F%2Ftown.town&cc_key=)   
order by 1

5. Построить таблицу с упорядоченным списком городов, в которых размещается только один поставщик, или производится только одна деталь, или собирается только одно изделие.

create table town as   
(   
select town   
from s   
group by town   
having count (s.n\_post) = 1   
union   
select town   
from p   
group by town   
having count (p.n\_det) = 1   
union   
select town   
from j   
group by town   
having count (j.n\_izd) = 1   
)

6. Увеличить рейтинг всех поставщиков, имеющих в настоящее время рейтинг меньше, чем у поставщика S5, на величину, равную умноженному на 2 числу выполненных поставщиком поставок.

update s set reiting = reiting+(select sum(kol)   
 from spj   
 where spj.n\_post = s.n\_post)\*2   
where reiting<(select reiting

from s

where(n\_post)='S5' )

1. Выбрать поставщиков, поставляющих детали, поставляемые поставщиками, проживающими в Лондоне.

select distinct t.n\_post   
from spj t   
where t.n\_det in(   
 select distinct n\_det   
 from spj   
 join s on spj.n\_post = s.n\_post   
 where [s.town](https://vk.com/away.php?to=http%3A%2F%2Fs.town&cc_key=" \t "_blank)='Лондон'   
 ) order by 1

1. Вывести без повторений пары городов таких, что была поставка детали из первого города для изделия во втором городе. Упорядочить список по алфавиту.

select distinct [p.town](https://vk.com/away.php?to=http%3A%2F%2Fp.town&cc_key=" \t "_blank), [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
from spj   
join p on p.n\_det = spj.n\_det   
join j on j.n\_izd = spj.n\_izd   
where [p.town](https://vk.com/away.php?to=http%3A%2F%2Fp.town&cc_key=" \t "_blank) != [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
order by [p.town](https://vk.com/away.php?to=http%3A%2F%2Fp.town&cc_key=" \t "_blank)

1. Получить список городов, в которые выполнили поставки ТОЛЬКО поставщики, поставлявшие голубые детали.

select distinct [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
from spj t   
join j on j.n\_izd = t.n\_izd   
where t.n\_post in (   
 select distinct spj.n\_post   
 from spj   
 join p on p.n\_det = spj.n\_det   
 where p.cvet = 'Голубой'   
 )   
except   
select distinct [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
from spj t   
join j on j.n\_izd = t.n\_izd   
where not t.n\_post in (   
 select distinct spj.n\_post   
 from spj   
 join p on p.n\_det = spj.n\_det   
 where p.cvet = 'Голубой'   
 )

10. Вывести полный список городов и для каждого города найти суммарное количество деталей красного цвета, которые были в него поставлены. Города в списке должны быть ВСЕ. Список должен быть упорядочен по алфавиту.

select [town.town](https://vk.com/away.php?to=http%3A%2F%2Ftown.town&cc_key=" \t "_blank), pt.skol   
from ( select town   
 from p   
 union   
 select town   
 from s   
 union   
 select town   
 from j   
 ) town   
left join ( select [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank), sum(spj.kol) skol   
 from spj   
 join p on spj.n\_det = p.n\_det   
 join j on spj.n\_izd = j.n\_izd   
 where p.cvet = 'Красный'   
 group by [j.town](https://vk.com/away.php?to=http%3A%2F%2Fj.town&cc_key=" \t "_blank)   
 ) pt on [pt.town](https://vk.com/away.php?to=http%3A%2F%2Fpt.town&cc_key=" \t "_blank) = [town.town](https://vk.com/away.php?to=http%3A%2F%2Ftown.town&cc_key=)   
order by 1

11. Построить таблицу с упорядоченным списком городов, в которых размещается только один поставщик, или производится только одна деталь, или собирается только одно изделие.

create table town as   
(   
select town   
from s   
group by town   
having count (s.n\_post) = 1   
union   
select town   
from p   
group by town   
having count (p.n\_det) = 1   
union   
select town   
from j   
group by town   
having count (j.n\_izd) = 1   
)

12. Увеличить рейтинг всех поставщиков, имеющих в настоящее время рейтинг меньше, чем у поставщика S5, на величину, равную умноженному на 2 числу выполненных поставщиком поставок.

update s set reiting = reiting+(select sum(kol)   
 from spj   
 where spj.n\_post = s.n\_post)\*2   
where reiting<(select reiting

from s

where(n\_post)='S5' )