

Комментарий_1. В ответах используется диалект Oracle SQL.

1. Выведите в один общий список без повторений наименование производителей (Vendor) из двух таблиц.

T1

| Vendor_id | Vendor |
|-----------|---------|
| 1 | Asus |
| 2 | Sony |
| 3 | Samsung |
| 4 | Acer |
| 5 | LG |
| 6 | HP |

T2

| Vendor_id | Vendor |
|-----------|--------|
| 1 | Epson |
| 2 | HP |
| 3 | Canon |
| 4 | Xerox |

```
select Vendor from T1
union
select Vendor from T2;
```

2. В базе данных имеется таблица **BOOKS**

| ID | Name | Author | Publisher | Year |
|----|-----------------------|---------------|-----------|------|
| 1 | Война и мир | Толстой Л.Н. | Мысль | 2014 |
| 2 | Мертвые души | Гоголь Н.В. | Азбука | 2013 |
| 3 | День опричника | Сорокин В.Г. | АСТ | 2006 |
| 4 | Финансовая математика | Четыркин Е.М. | Мысль | 2015 |

Выберите запросы, возвращающие всю информацию о книге, выпущенной издательствами "Азбука" или "Мысль".

- ☒ A. SELECT * FROM BOOKS WHERE PUBLISHER IN ('Мысль', 'Азбука')
- ☒ B. SELECT * FROM BOOKS WHERE PUBLISHER ='Мысль' or PUBLISHER = 'Азбука'
- ☐ C. SELECT * FROM BOOKS WHERE PUBLISHER ='Мысль' and PUBLISHER = 'Азбука'
- ☐ D. SELECT * FROM BOOKS WHERE PUBLISHER ='Мысль', PUBLISHER = 'Азбука'
- ☒ E. SELECT * FROM BOOKS WHERE PUBLISHER ='Мысль' UNION select * from BOOKS where PUBLISHER = 'Азбука'

3. Имеются 2 таблицы **T1** и **T2**, содержащие колонки **NUM** типа **NUMBER**. Напишите запрос, отбирающий из таблицы **T1** все уникальные числа **NUM**, отсутствующие в колонке **NUM** таблицы **T2**.

a. Не используя конструкцию left join

```
select
    DISTINCT "NUM"
from
    T1
where
    "NUM" not in (select "NUM" from T2);
```

b. Используя конструкцию left join

```
select
    DISTINCT T1."NUM"
from
    T1
left outer join T2 on T1."NUM" = T2."NUM"
where
    T1."NUM" not in (SELECT "NUM" FROM
T2);
```

4. В таблице **T** имеется одна колонка **NUM** типа **NUMBER**. Таблица заполнена некоторыми числами, которые могут повторяться. Написать оператор DELETE, удаляющий за один проход из таблицы T все записи дубликаты (остаться должны только неповторяющиеся числа).

```
delete from T
where
    "NUM" in (
        /* Создаем список дублирующихся значений столбца NUM */
        select
            "NUM"
        from
            T
        group by
            "NUM"
        having
            count("NUM") > 1
    );
```

5. В таблице **T1** Имеются данные по продажам и выполнению плана за последний отчетный период по каждому дополнительному офису. Определить 3 лучших филиала по выполнению плана и вывести общую сумму продаж по каждому из них. Результаты необходимо представить в виде таблицы **T2**.

T1

| Fil | DO | Sales | Plan, % |
|-----------------|-------------|-------|---------|
| Центральный | Москва 1 | 80 | 120% |
| Центральный | Москва 2 | 120 | 100% |
| Центральный | Тула | 30 | 80% |
| Дальневосточный | Хабаровск | 20 | 101% |
| Дальневосточный | Владивосток | 40 | 103% |
| Южный | Сочи | 60 | 95% |
| Южный | Анапа 2 | 80 | 105% |
| Северный | Спб 1 | 70 | 105% |

T2

| Place | Fil | Sales | Plan, % |
|-------|-----|-------|---------|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Комментарий_2. В рамках тестового задания рейтинг филиалов будет определяться по среднему значению процента выполнения плана среди доп. офисов.

В рамках реальной задачи необходимо более детально подойти к механизму составления рейтинга. Возможна необходимость применения иных мер центральной тенденции или разработка специализированной метрики.

Комментарий_3. При выводе 3 филиалов с наилучшими показателями запрос учитывает возможность наличия у филиалов одинаковых значений рейтинга.

```
with inter_tab as(
    select
        LISTAGG(DO, ', ') as Place,
        FIL,
        sum(Sales) as Sales,
        /* Поскольку значения поля "Plan, %" являются строковыми предварительно надо
        избавиться от последнего символа и перевести в числовой тип данных */
        avg(to_number(substr("Plan, %", 1, length("Plan, %") - 1))) as "Plan, %"
    from T1
    group by FIL)

select *
from inter_tab
where "Plan, %" in (select "Plan, %" from inter_tab order by "Plan, %" DESC FETCH
FIRST 3 ROWS ONLY )
order by "Plan, %" desc;
```

Есть таблица с зарплатой сотрудников компании **Salary**

В ней хранятся поля: уникальный идентификатор записи **Id** (uniqueidentifier, не NULL), дата выплаты зарплаты **Date** (datetime2(7)), идентификатор сотрудника **EmployeeId** (uniqueidentifier) - поле ссылается на поле **Id** из таблицы **Employees**, сумма зарплаты **Salary**(int)

Salary

| Id | Date | EmployeeId | Salary |
|----|------|------------|--------|
| | | | |

Есть таблица с информацией о сотрудниках компании **Employees** со столбцами (уникальный идентификатор записи **Id** (uniqueidentifier, не NULL) , дата оформления трудоустройства **Date** (datetime2(7)), ФИО **FullName**(nvarchar(500)), название департамента **Department** (nvarchar(250)), электронная почта сотрудника **Email** (nvarchar(250)), телефон **Phone**(nvarchar(250)))

Employees

| Id | Date | FullName | Department | Email | Phone |
|----|------|----------|------------|-------|-------|
| | | | | | |

Задачи

- 1) Вывести названия департаментов с наибольшим и наименьшим количеством сотрудников

```
with inter_tab as(
    select
        Department,
        count(Id) as Amount_Employees
    from Employees
    group by
        Department)

select Department, Amount_Employees
from inter_tab
where Amount_Employees = (select min(Amount_Employees) from inter_tab) or
    Amount_Employees = (select max(Amount_Employees) from inter_tab);
```

- 2) Вывести сумму выплаченной зарплаты в разбивке по месяцам.

```
/* Применена конструкция TO_CHAR("Date", 'MM.YYYY') поскольку неизвестно
содержит ли таблица данные более чем за один год */

select
    TO_CHAR("Date", 'MM.YYYY'),
    sum(salary) as salary
from Salary
group by TO_CHAR("Date", 'MM.YYYY');
```

- 3) Вывести ФИО людей, получивших зарплату за апрель меньшую, чем 50 тыс рублей в поле **Less50**; ФИО оставшихся людей вывести в другое поле - **More50**.

```

with inter_tab_1 as(
    select
        ROW_NUMBER() over (order by Employees.FullName) AS id_,
        Employees.FullName as Less50

    from Salary
    inner join Employees on Salary.EmployeeId = Employees."Id"
    where (EXTRACT(MONTH FROM Salary."Data")=4) and (Salary.Salary<50000)),

inter_tab_2 as(
    select
        ROW_NUMBER() over (order by Employees.FullName) AS id_,
        Employees.FullName as More50

    from Salary
    inner join Employees on Salary.EmployeeId = Employees."Id"
    where (EXTRACT(MONTH FROM Salary."Data")=4) and (Salary.Salary>=50000))

select
    inter_tab_1.Less50,
    inter_tab_2.More50

from inter_tab_1
full outer JOIN inter_tab_2 on inter_tab_1.id_ = inter_tab_2.id_;

```

- 4) Изменить название таблицы **Employees** на **Workers**, добавить поле **Comment**(nvarchar(255))

```

alter table Employees rename to Workers;
alter table Workers add (Comment nvarchar(255));

```

- 5) Присвоить сотрудникам из таблицы **Salary** порядковый номер исходя из величины полученной зарплаты в апреле.

```

select
    EmployeeId,
    dense_rank() over (order by Salary) as dense_rank_salary

from Salary
where EXTRACT(MONTH FROM "Date")=4;

```