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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **T1** |  |  | **T2** |  |
| **Vendor\_id** | **Vendor** |  | **Vendor\_id** | **Vendor** |
| 1 | Asus |  | 1 | Epson |
| 2 | Sony |  | 2 | HP |
| 3 | Samsung |  | 3 | Canon |
| 4 | Acer |  | 4 | Xerox |
| 5 | LG |  |  |  |
| 6 | HP |  |  |  |

|  |
| --- |
| 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’ |
|  |
|  |
| 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 | b. Используя конструкцию left join |
| WITH subq AS (SELECT NUM  FROM T2)  SELECT DISTINCT NUM  FROM T1  WHERE NUM NOT IN (SELECT \* FROM subq); | SELECT DISTINCT T1.NUM  FROM T1 LEFT JOIN  T2 ON T1.NUM = T2.NUM  WHERE T2.NUM IS NULL; |

**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).

|  |
| --- |
| WITH CTE AS  (SELECT \*,ROW\_NUMBER() OVER (PARTITION BY NUM ORDER BY NUM) AS RN  FROM T)  DELETE FROM CTE WHERE RN<>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** |  |  |  |  | **T2** |  |  |  |
| **Fil** | **DO** | **Sales** | **Plan, %** |  | **Place** | **Fil** | **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% |  |  |  |  |  |

|  |
| --- |
| WITH CTE AS (SELECT Fil, SUM(Sales) AS Sales, SUM(Plan, %) AS Plan  FROM T1  GROUP BY Fil  ORDER BY Plan DESC  LIMIT 3)  SELECT \*, ROW\_NUMBER() OVER (ORDER BY Plan DESC) AS Place  FROM CTE; |

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

* Вывести названия департаментов с наибольшим и наименьшим количеством сотрудников

|  |
| --- |
| WITH CTE AS (SELECT Department, COUNT(id) AS cnt\_id  FROM Employees GROUP BY Department)  SELECT Department FROM CTE  WHERE cnt\_id IN ((SELECT MAX(cnt\_id) FROM CTE), (SELECT MAX(cnt\_id) FROM CTE)) |

* Вывести сумму выплаченной зарплаты в разбивке по месяцам.

|  |
| --- |
| SELECT DATE\_PART(‘month’, Date::DATE) AS Months, SUM(Salary) AS Month\_salary  FROM Salary  GROUP BY Months  ORDER BY Months DESC; |

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

|  |
| --- |
| SELECT DISTINCT CASE WHEN Salary < 50000 THEN FullName END AS Less50,  CASE WHEN Salary > 50000 THEN FullName END AS More50  FROM Salary s INNER JOIN Employees e ON s.Employeeid = e.id |

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

|  |
| --- |
| RENAME TABLE Employees TO Workers;  ALTER TABLE Workers ADD Comment nvarchar(255) NULL; |

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

|  |
| --- |
| SELECT Employeeid, ROW\_NUMBER() OVER (ORDER BY Salary DESC) AS seq\_num  FROM Salary  WHERE DATE\_PART(‘month’, Date::DATE) = 4 |