**SQL**

*Задача №1*

Написать запрос, приводящий таблицу 1 Динамики курса валют (CurrencyDynamics) к виду таблицы 2 (CurrencyDynamicsRes).

Таблица 1

|  |  |  |
| --- | --- | --- |
| CURR | DT | VAL |
| USD | 24.10.2019 | 63,7997 |
| USD | 25.10.2019 | 63,8600 |
| USD | 26.10.2019 | 63,9966 |
| USD | 29.10.2019 | 69,8700 |
| USD | 30.10.2019 | 63,8320 |
| USD | 31.10.2019 | 63,8734 |
| EUR | 24.10.2019 | 70,9644 |
| EUR | 25.10.2019 | 71,1400 |
| EUR | 26.10.2019 | 71,1194 |
| EUR | 29.10.2019 | 70,8382 |
| EUR | 30.10.2019 | 70,7769 |
| EUR | 31.10.2019 | 71,0081 |

Таблица 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CURR | DT\_FROM | VAL\_FROM | DT\_TO | VAL |
| EUR | 24.10.2019 | 70,9644 | 25.10.2019 | 71,1400 |
| EUR | 25.10.2019 | 71,1400 | 26.10.2019 | 71,1194 |
| EUR | 26.10.2019 | 71,1194 | 29.10.2019 | 70,8382 |
| EUR | 29.10.2019 | 70,8382 | 30.10.2019 | 70,7769 |
| EUR | 30.10.2019 | 70,7769 | 31.10.2019 | 71,0081 |
| EUR | 31.10.2019 | 71,0081 |  |  |
| USD | 24.10.2019 | 63,7997 | 25.10.2019 | 63,8600 |
| USD | 25.10.2019 | 63,8600 | 26.10.2019 | 63,9966 |
| USD | 26.10.2019 | 63,9966 | 29.10.2019 | 69,8700 |
| USD | 29.10.2019 | 69,8700 | 30.10.2019 | 63,8320 |
| USD | 30.10.2019 | 63,8320 | 31.10.2019 | 63,8734 |
| USD | 31.10.2019 | 63,8734 |  |  |

*Задача №2*

В БД определены таблицы:

[dbo].[rep\_BR\_complaints] - таблица с обращениями клиентов (жалобы) со следующими полями:

 1.1 ID - уникальный идентификатор (Primary Key)  
 1.2 RequestSBDate - дата поступления обращения в Сбербанк  
 1.3 BaseCategoryCode - code направления обращения, который принимает значения из словаря:  
         0: Исполнительное производство  
         1: Кредиты  
         2: Банковские карты  
         3: Вклады   
         4: Платежи / переводы  
         5: Прочее  
 1.4 CategoryId - идентификатор категории обращения (Foreign Key)

[dbo].[rep\_BR\_complaints\_category] - таблица с категориями обращений клиентов со следующими полями:

 2.1 ID - уникальный идентификатор (Primary Key)  
 2.2 BaseCategoryCode -code направления обращения (аналогии с BaseCategoryCode из таблицы [dbo].[rep\_BR\_complaints])  
 2.3 Name - "наименование" категории обращения

Задача: Необходимо написать sql-запрос, который сформирует отчет следующего вида:

| **Год** | **Месяц** | **Направление обращения** | **Категория обращения** | **Количество обращений** |
| --- | --- | --- | --- | --- |
| 2017 | 1 | Кредиты | Реструктуризация/изменение % ставки | 100 |
| 2017 | 10 | Прочее | Прочее | 90 |
| ... | ... | ... | ... | ... |
| 2017 | 2 | Банковские карты | Технический сбой УС | 30 |

\* Только на данных за 2017 год; вывести результаты на экран в порядке уменьшения количества обращений.

*Задача №3*

В БД определена таблица [dbo].[Problem\_2\_Test] - таблица со следующими полями:

1 ID - уникальный идентификатор (Primary Key)  
2 Name - ФИО  
  
Некоторые записи удалены. То есть, существуют пропущенные ID.  
  
Задача: Необходимо написать sql-запрос, который выводит на экран минимальный пропущенный ID.  
  
\* В рамках данной задачи считается, что всегда существуют запись с ID = 1 и между 1 и max(ID) существует хотя бы один пропуск.  
  
Пример: Пропущенные значения ID - 2, 4, 6, 9. Ответ - 2.

| **ID** | **ФИО** |
| --- | --- |
| 1 | Иванов А. А. |
| 3 | Иванов А. Б. |
| 5 | Иванов А. В. |
| 7 | Иванов А. Г. |
| 8 | Иванов А. Д. |
| 10 | Иванов А. Ж. |

Для решения данной задачи предлагаем воспользоваться ресурсом: <https://sqliteonline.com/>.  
Код создания таблицы ниже:

create table [dbo].[Problem\_2\_Test]  
(  
  ID int primary key,  
  FIO nvarchar(255) not null  
)  
insert into [dbo].[Problem\_2\_Test]  
values(1, 'Ivanov A. A.'),  
(3, 'Ivanov A. B.'),  
(5, 'Ivanov A. C.'),  
(7, 'Ivanov A. D.'),  
(8, 'Ivanov A. E.'),  
(10, 'Ivanov A. F.')

*Задача №4*

Таблица [dbo].[turnover]

В БД магазина имеется таблица [dbo].[turnover], в которой хранится информация об обороте товара на складах.  
Это сеть магазинов, поэтому складов несколько.  
В отдел аналитики поступил запрос на поиск случаев, когда какой-нибудь товар полностью пропадал со всех складов.  
Менеджеры хотят выявить такие периоды и разобраться почему так получилось.

Необходимо вывести:

* ID товара
* с какой даты пропал товар с полок
* по какую дату товар отсутствовал на полках

Работать предлагаем в <https://sqliteonline.com/>  
Просто вставьте туда код, расположенный ниже

create table [dbo].[turnover]

(

    ID int primary key identity(1, 1),

    branch\_id int not null, --ID филиала

    good\_id int not null, --ID товара

    oper\_date datetime not null, --дата операции

    oper\_type bit not null, --тип операции 1 - приход, 0 - расход

    good\_count int not null --количество товара в резултате операции

)

insert into [dbo].[turnover] (branch\_id, good\_id, oper\_date, oper\_type, good\_count)

values(1, 1, '20200201 10:05:25', 1, 20),

(2, 1, '20200201 11:24:48', 1, 10),

(1, 2, '20200201 10:20:38', 1, 15),

(1, 1, '20200201 14:26:28', 0, 4),

(2, 1, '20200204 11:48:29', 0, 8),

(1, 1, '20200205 20:13:57', 0, 9),

(1, 1, '20200206 16:15:38', 1, 4),

(1, 1, '20200215 09:00:26', 0, 10),

(2, 1, '20200216 12:27:59', 0, 2),

(1, 1, '20200217 22:16:36', 0, 1),

(1, 2, '20200219 14:28:49', 0, 5),

(1, 1, '20200301 14:25:36', 1, 10),

(2, 1, '20200301 15:26:36', 1, 10),

(1, 2, '20200305 11:12:34', 0, 10),

(2, 1, '20200307 16:16:46', 0, 7)

*Задача №5*

Дан набор данных по маршрутам инкассации следующей структуры:

Таблица ROUTES

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Описание |
| ROUTE\_ID | number | Уникальный идентификатор маршрута |
| ROUTE\_START\_TIME | date | Дата и время выезда маршрута с базы |
| ROUTE\_END\_TIME | date | Дата и время возвращения маршрута на базу |

Требуется определить максимальное количество одновременно работающих маршрутов по всей выборке (даже если маршруты пересекаются на 1 секунду они считаются одновременно работающими).

Например: если маршрут 1 выехал в 10.00  и вернулся в 14.00, а маршрут 2 выехал в 13.00 и вернулся в 18.00 и больше никаких маршрутов не было, то в период с 13.00 до 14.00 одновременно работали 2 маршрута.

Во вложении тестовый набор данных dataset5.csv.

*Задача №6*

Дан набор данных по маршрутам инкассации следующей структуры:

Таблица OPERATIONS

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Описание |
| ROUTE\_ID | number | Уникальный идентификатор маршрута |
| OBJECT\_NUM | number | Порядковый номер объекта на маршруте |
| START\_TIME | date | Дата и время начала обслуживания клиента |
| END\_TIME | date | Дата и время окончания обслуживания клиента |

Требуется посчитать:

* 1. время операции в минутах
  2. как соотносится время текущей операции к среднему времени операции на маршруте (если больше -1, если меньше – 0)
  3. как соотносится время текущей операции к среднему времени трех предыдущих операций на маршруте (если больше -1, если меньше – 0)

Во вложении тестовый набор данных dataset6.csv и образец с решением в Excel (task 6 example.xlsx)

**Python**

*Задача №1*

Необходимо реализовать метод сортировки списка целых (int) элементов. Метод должен быть реализован без использования "готовых" методов (sorted, numpy.sort и т.д.).  
  
input: list[int, ... , int].  
  
output: отсортированный в порядке убывания list.

*Задача №2*

Даны координаты 4-х точек на плоскости: (x1, y1), (x2, y2), (x3, y3), (x4, y4). Определить, можно ли по данным точкам достроить квадрат.  
  
input: (int, int), (int, int), (int, int), (int, int)  
  
output: YES | NO