Министерство науки и высшего образования РФ

Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа «Киберфизических систем и управления»

**ОТЧЕТ**

по дисциплине «Корпоративные базы данных»

Задания по материалу занятия 3

**Выполнил:**

студент гр. 5132703/00101 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. В. Морозов

подпись, дата

**Проверил:**

доцент, к.т.н \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. А. Нестеров

подпись, дата

Санкт-Петербург

2023 г.

Часть 1

1. Запустите приложение Management Studio и выполните подключение к указанному экземпляру SQL Server (в классе это saiu-sqlsrv).

2. Используя базу данных BestDatabase (под тем именем, как вы ее создавали в предыдущей работе), ознакомьтесь со структурой таблицы [BestDatabase].[dbo].[PEOPLE].

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 1

Напишите код, проверяющий наличие временной таблицы #T и удаляющий ее, если она есть в базе данных.



Рисунок 2

Создайте временную таблицу #T со столбцами COD, NAME1, NAME3, BIRTHDAY, SEX, куда поместите данные из соответствующих столбцов исходной таблицы.

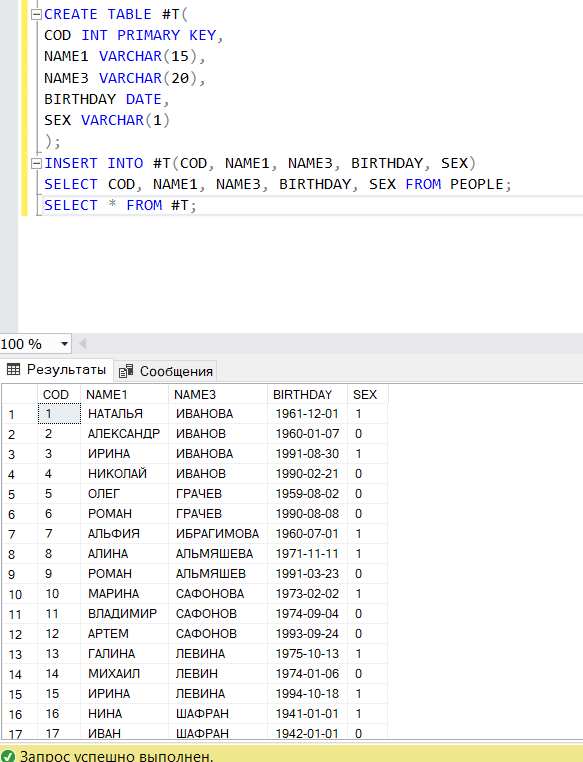


Рисунок 3

3. Создайте табличную переменную @T1, куда из таблицы #T поместите перечень встречающихся имен (именно имен, а не фамилий), указание на пол в виде «мужской», «женский» или «не определено», количество записей, в которых встречалось это имя. Оператором SELECT выведите содержимое табличной переменной.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 4

4. Создайте целочисленную переменную @MaxI, присвойте ей значение, соответствующее максимальному числу повторений имени из таблицы @T1. Выведите значение оператором print.

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 5

5. Предложите и реализуйте пример, иллюстрирующий различия в области видимости локальной временной таблицы и табличной переменной.

Реализуем пример на основе предыдущего задания.

Объявим табличную переменную @T и выведем её содержимое.

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 6

Запрос выполняется успешно, так как находится в одном пакете заданий.

Теперь с помощью ключевого слова GO закроем пакет до применения select

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 7

Табличная переменная успешно создаётся, но select из неё не выполняется.

Это связано с тем, что табличная переменная создаётся локально в своём пакете заданий, а select находится в другом пакете и ничего не знает про @T.

Локальная временная таблица же видна в создавшем ее сеансе, на уровне ее создания и на всех внутренних уровнях стека вызовов (внутренние процедуры, функции и т. д.). Поэтому вплоть до окончания сеанса мы можем с ней работать.

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

Рисунок 8

6. Удалите временную таблицу #T

Изображение выглядит как текст, электроника, снимок экрана, дисплей

Автоматически созданное описание

Рисунок 9

Изображение выглядит как текст, снимок экрана, Шрифт, дисплей

Автоматически созданное описание

Рисунок 10

Часть 2

1. Запустите приложение Management Studio и выполните подключение к указанному экземпляру SQL Server (в классе это saiu-sqlsrv).

2. Используя результаты предыдущей лабораторной работы получите временную таблицу #T со столбцами COD, NAME1, NAME3, BIRTHDAY, SEX, куда поместите данные из соответствующих столбцов таблицы [BestDatabase].[dbo].[PEOPLE].

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11

Создайте табличную переменную @T2, поместите в нее перечень встречающихся имен и количество записей, в которых встречалось это имя.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 12

Создайте целочисленную переменную @MaxI, присвойте ей значение, соответствующее максимальному числу повторений имени из таблицы @T2.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 13

3. В цикле выведите отдельным запросом имя и к-во повторений имени для всех значений от @MaxI до 1 (перебираем значения от большего к меньшему).

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Рисунок 14

Измените ход выполнения цикла, так чтобы пропускался вывод имен, встреченных 3 раза.

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Рисунок 15

Снова измените ход выполнения цикла, так чтобы прерывался на цифре 4, т.е. выводились только имена с числом повторений от @MaxI до 5 (или ничего бы не выводилось, если @MaxI<=4).

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 16

4. Используя системные хранимые процедуры получите:

- Список учетных записей пользователей SQL Server

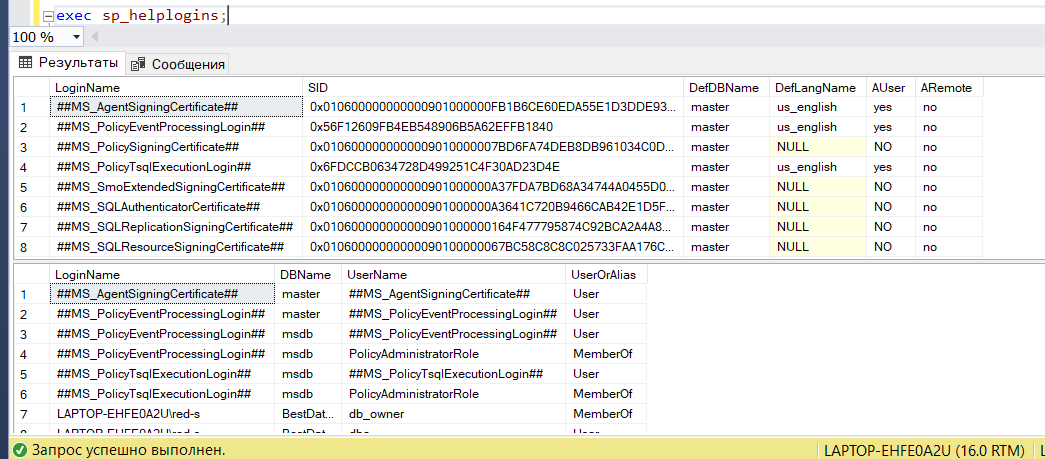


Рисунок 17

- Информацию о таблице [dbo].[PEOPLE]: владельца, дату создания, информацию о столбцах….

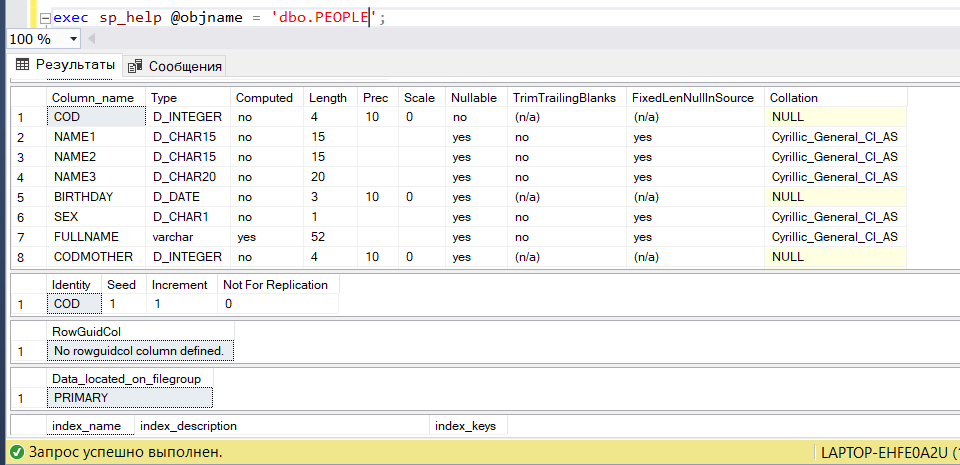


Рисунок 18

- Перечень хранимых процедур, определенных в базе BestDatabase.

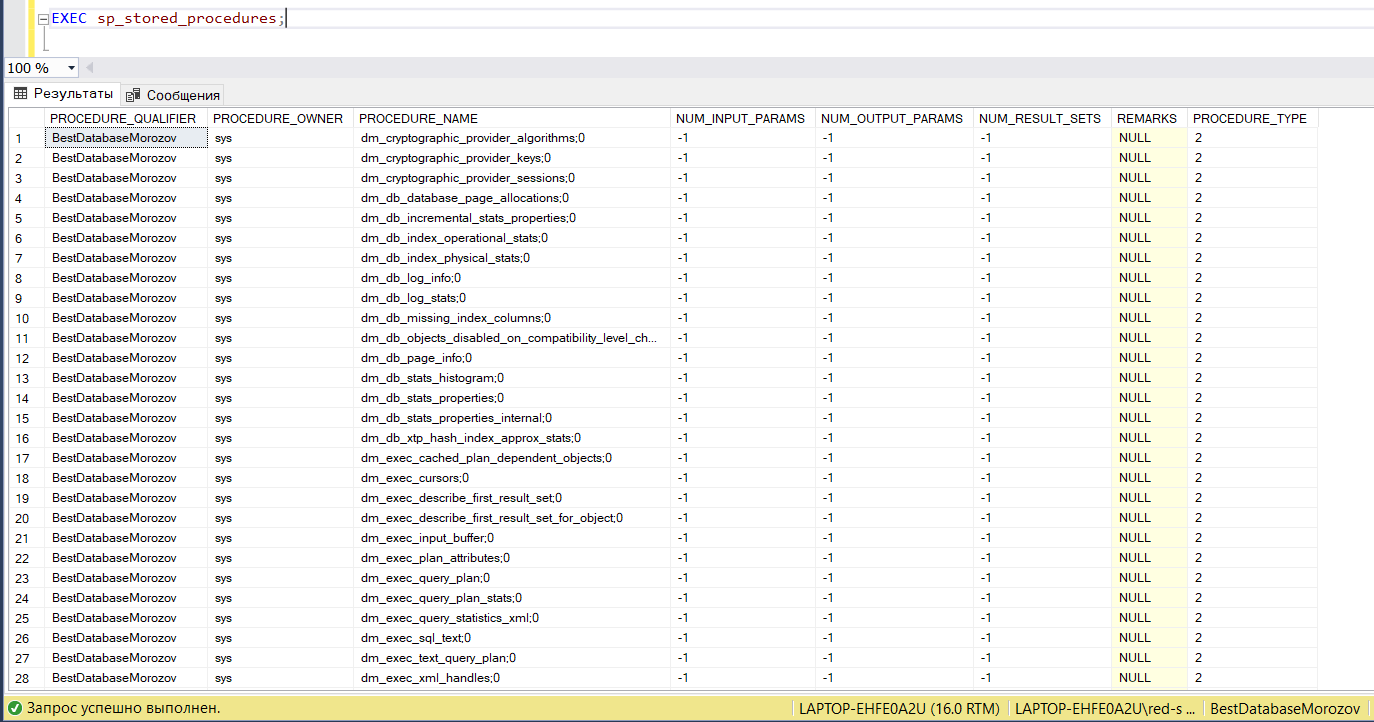


Рисунок 19

5. Оформите действия, выполненные на шагах 2,3 в виде хранимой процедуры dbo.spMyProc. В качестве входного параметра процедура принимает значение 1, 2 или 3, которое соответствует первому, второму или третьему варианту реализации цикла while. В качестве выходного значения возвращается значение переменой @MaxI. При нормальном завершении работы процедура возвращает код 0. Если временная таблица #T уже существует, процедура заканчивает работу, возвращая код 1. Если список имен не сформирован (в #T пустое множество записей), процедура возвращает 2. Если в качестве входного параметра получено значение, отличное от 1, 2 или 3 – процедура возвращает 4. Выполните вызов процедуры с анализом возвращаемого значения, если процедура вернула 0, выведите значение выходного параметра с максимальным числом повторения имени. Протестируйте возможные варианты работы.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 20

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 21

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 22

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 23

6. Для таблицы [dbo].[PEOPLE] создайте триггер на обновление и изменение, не позволяющий вносить в список людей с датой рождения позже текущей (GETDATE()), а также мужчин с возрастом более 100 лет и женщин в возрасте более 110 лет. Разберитесь с функциями для обработки дат, чтобы триггер использовал текущую дату сервера и делал подсчет с точностью до дня (т.е. за день до его 100-летия мужчину в список добавить можно). При попытке внести «неправильную» запись, надо сгенерировать ошибку с сообщением о причине и откатить транзакцию. Протестируйте работу триггера на разных комбинациях вводимых данных. Проверьте правильность работы триггера в случае, когда одним оператором insert вносится несколько записей.