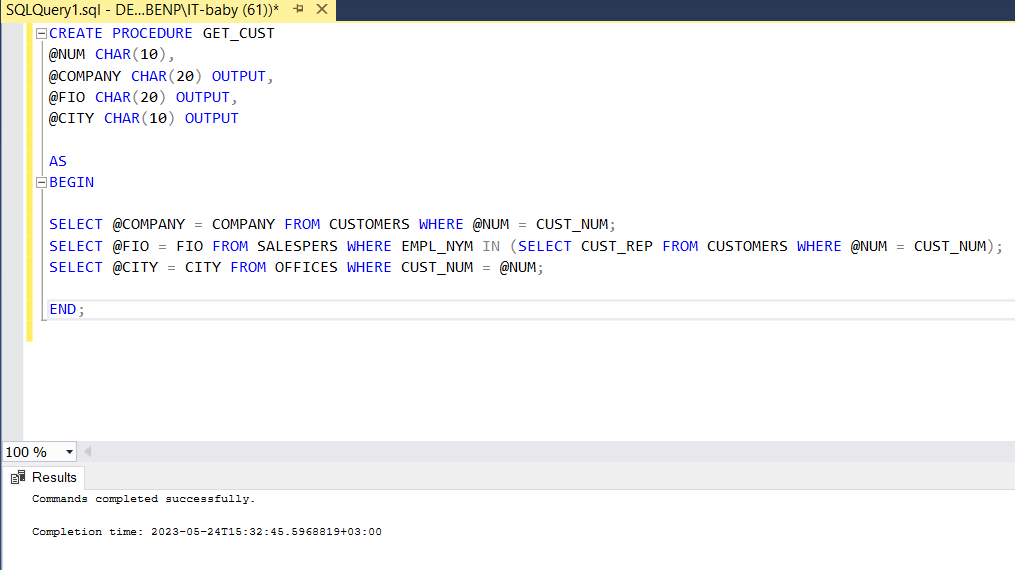
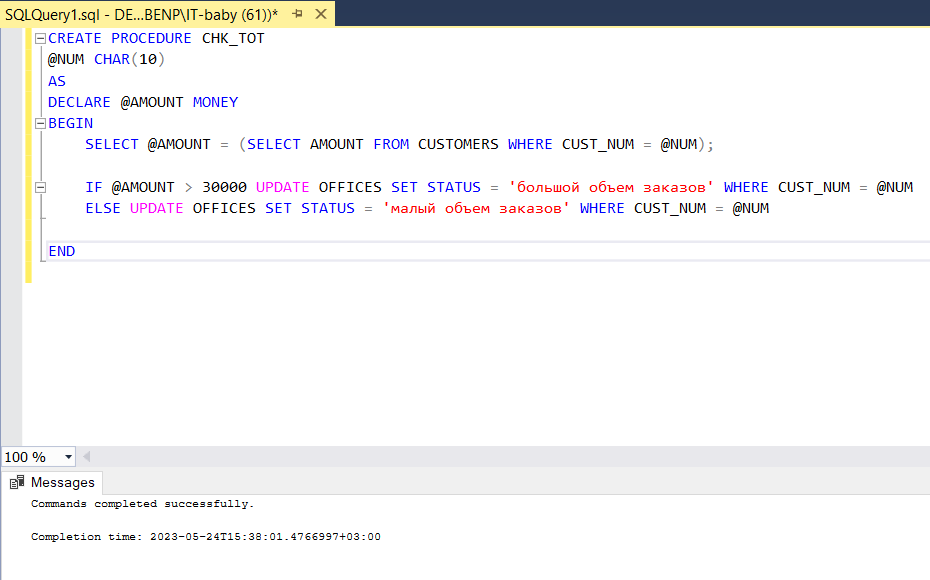
Отчёт: Лабораторная работа 10

Задание 1.1. Написать хранимую процедуру GET\_CUST, которая получает идентификатор клиента и возвращает его имя (поле COMPANY таблицы CUSTOMERS), имя закрепленного за ним служащего (поле FIO таблицы SALESPERS) и название города, в котором расположен офис этого клиента (поле CITY таблицы OFFICES). Первый из передаваемых параметров - входной, остальные три – выходные (используются для передачи запрошенных данных вызывающей процедуре). Продемонстрировать вызов данной процедуры из другой процедуры Transact-SQL. Например, GET\_CUST1 (идентификатор\_клиента).



Задание 1.2. Написать хранимую процедуру CHK\_TOT, которая получает идентификатор клиента и вычисляет общую стоимость его заказов (поле AMOUNT таблицы CUSTOMERS) и в зависимости от того, превысит ли эта сумма 30 000$, заносит в поле STATUS таблицы OFFICES одно из двух примечаний – “большой объем заказов”, ”малый объем заказов”.

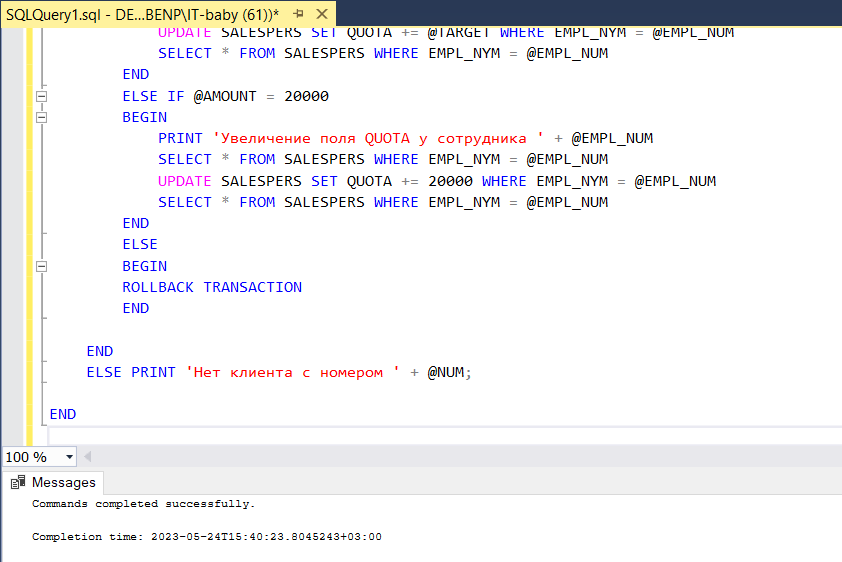


Задание 1.3. Написать хранимую процедуру для добавления данных о новом клиенте в таблицу OFFICES.

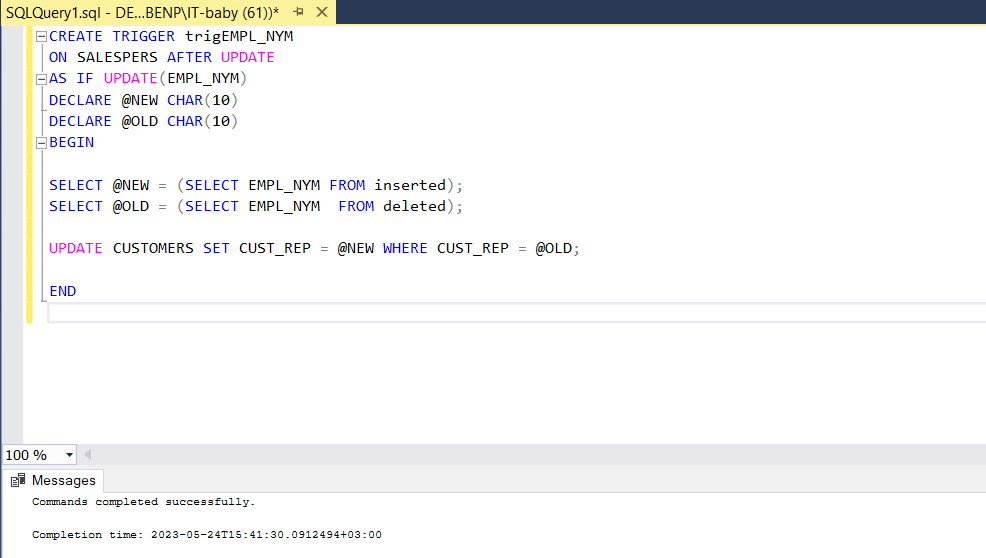
 Добавить новую строку в таблицу OFFICES. Разрешается добовление только для клиентов, уже имеющих запись в таблице CUSTOMERS.

 Обновить запись в SALESPERS, увеличив поле QUOTA для соответствующего служащего (для каждого, с которым работает клиент) на величину объема продаж добавленной в предыдущем пункте записи. Плановый объем продаж служащего не может быть увеличен более чем на определенную величину. Если сумма заказов рассматриваемого клиента составляет менее 20 000$ (вызов

процедуры CHK\_TOT, которая должна возвращать эту сумму, например, в качестве выходного параметра), то величина объема продаж (поле TARGET таблицы OFFICES) будет добавлена к плану служащего. Если сумма заказов рассматриваемого клиента равна 20 000$ к плану будут добавлены фиксированные 20 000$. В противном случае – запретить добавление новой записи в OFFICES и обновление в SALESPERS. Предусмотреть промежуточный вывод суммы заказов рассматриваемого клиента.



Задание 1.4. Создать триггер, распространяющий любое обновление столбца EMPL\_NUM в таблице SALESPERS на столбец CUST\_REP таблицы CUSTOMERS.



Задание 1.5. Создать триггер, который запускается каждый раз, когда запись вставляется в таблицу CUSTOMERS или модифицируется. Если заказ сделан не в первые 15 дней месяца, то запись не принимается.

Для выделения определенной части даты (дня) в виде целого значения используйте функцию DATEPART(date\_part, date), где date\_part- **day|month|year.**

Например:

SELECT DATEPART(month, GETDATE()) AS 'Month Number'

GO

Month Number

------------

11

