**Лабораторная работа № 8. Основы программирования на T-SQL**

**Вариант 12**

Цель: изучить основные возможности T-SQL; изучить работу с переменными, операции работы с переменными, основные функции, условия IF, оператор RETURN, блок обработки ошибок try-catch; ознакомится с глобальные переменные.

Использование программ на языке T-SQL позволяет расширить круг решаемых задач, возникающих при работе с базами данных.

Первое задание заключалось в создании переменных нескольких типов и выполнения действий над ними.

Имя переменной должно начинаться с символа @.

С помощью оператора SET можно переменной присвоить значение и выполнять вычисления. Оператор SELECT позволяет нескольким переменным присвоить значения.

Вывод данных в T-SQL возможен двумя способами: оператором SELECT можно сформировать выходной результирующий набор и с помощью оператора PRINT можно вывести строку в стандартный выходной поток – рисунок 1.1.

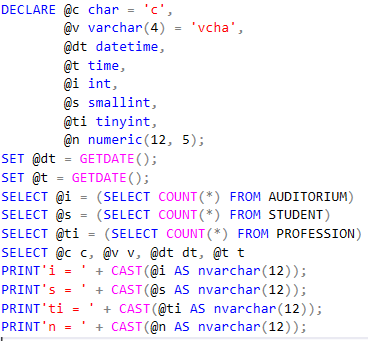


Рисунок 1.1 ­– Работа с переменными

Если одновременно выводятся данные, сформированные оператором SELECT и данные, сформированные оператором PRINT, то посмотреть последние можно на вкладке Messages.

Далее требовалось разработать скрипт, в котором определяется общая вместимость аудиторий. Когда общая вместимость превышает 200, то вывести количество аудиторий, среднюю вместимость аудиторий, количество аудиторий, вместимость которых меньше средней, и процент таких аудиторий. Когда общая вместимость аудиторий меньше 200, то вывести сообщение о размере общей вместимости. – рисунок 1.2.

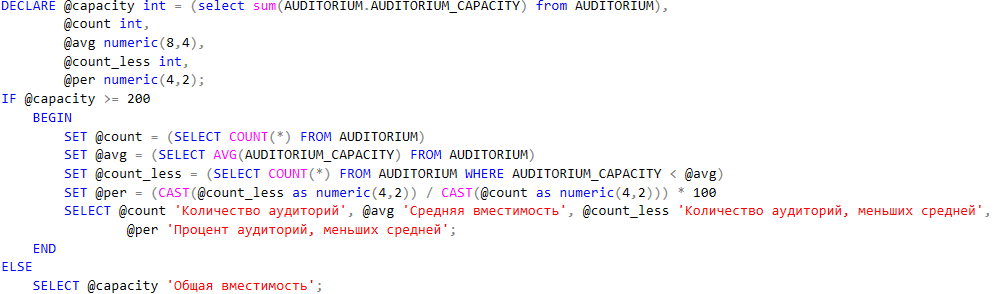


Рисунок 1.2 ­– Задание 2

1. Третье задание заключалось разработке T-SQL-скрипта, который выводит на печать глобальные переменные:

− @@ROWCOUNT (число обработанных строк);

− @@VERSION (версия SQL Server);

− @@SPID (возвращает системный идентификатор процесса, назначенный сервером текущему подключению);

− @@ERROR (код последней ошибки);

− @@SERVERNAME (имя сервера);

− @@TRANCOUNT (возвращает уровень вложенности транзакции);

− @@FETCH\_STATUS (проверка результата считывания строк результирующего набора);

− @@NESTLEVEL (уровень вложенности текущей процедуры).

Скрипт представлен на рисунке 1.3.

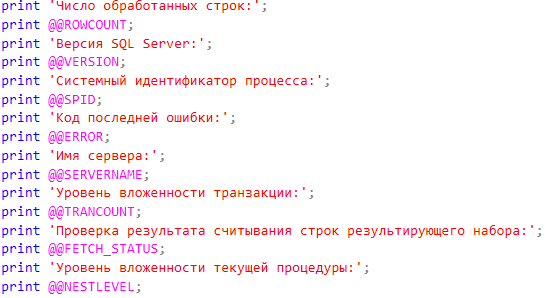


Рисунок 1.3 ­– Итоговый скрипт

Далее требовалось разработать скрипт для:

вычисления значения переменной z, в функции , для различных значений исходных данных;

преобразования полного ФИО в сокращенное;

поиска студентов, у которых день рождения в следующем месяце, и определение их возраста;

поиска дня недели, в который студенты некоторой группы сдавали экзамен по СУБД. Скрипт – рисунок 1.4.

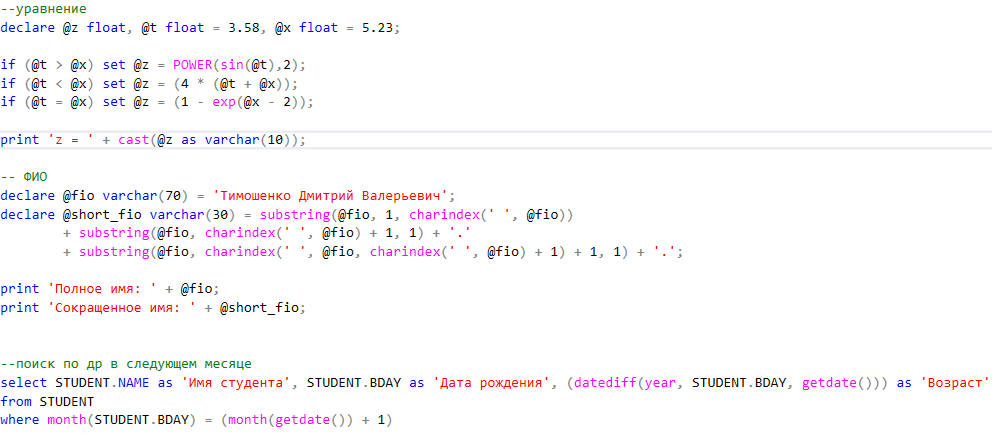


Рисунок 1.4 ­– Задание 4

В 5 задании надо было продемонстрировать работу с if…else – рисунок 1.5.

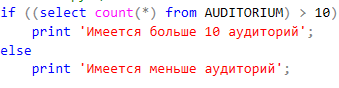


Рисунок 1.5 ­– Задание 5

Шестое задание – сценарий с CASE– рисунок 1.6.

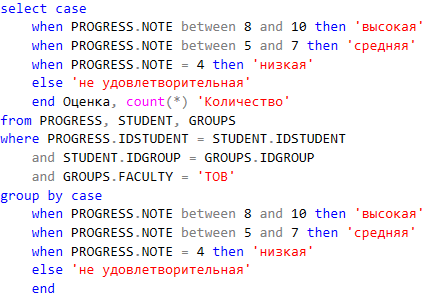


Рисунок 1.6 – Работа с CASE

Седьмое задание – создание временной локальной таблицы и заполнение, с использованием WHILE– рисунок 1.7.

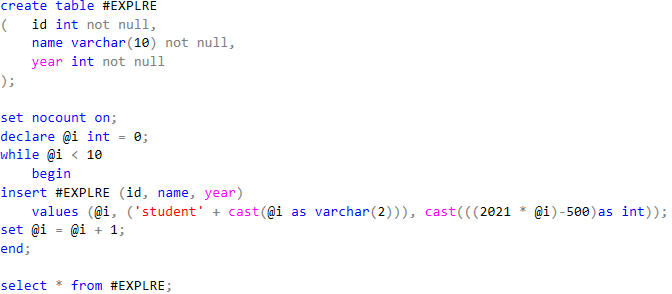


Рисунок 1.7 – Использование WHILE

В восьмом задании нужно было продемонстрировать использование оператора RETURN – рисунок 1.8.

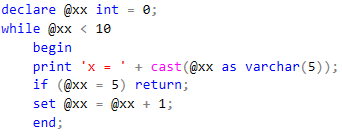


Рисунок 1.8 – Использование RETURN

В девятом задании требовалось разработать сценарий с ошибками, в котором используются для обработки ошибок блоки TRY и CATCH. Применить функции ERROR\_NUMBER (код последней ошибки), ERROR\_ES-SAGE (сообщение об ошибке), ERROR\_LINE (код последней ошибки), ERROR\_PROCEDURE (имя процедуры или NULL), ERROR\_SEVERITY (уровень серьезности ошибки), ERROR\_ STATE (метка ошибки) – рисунок 1.10.

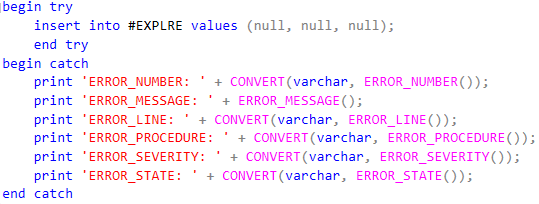


Рисунок 1.10 – Использование HAVING

**Вывод**

В ходе лабораторной работы были изучены и отработаны навыки работы с основными возможностями T-SQL; изучена работа с переменными, операции работы с переменными, основные функции, условия IF, оператор RETURN, блок обработки ошибок try-catch; ознакомились с глобальные переменные.