**10. Переменные.**

Имя локальной переменной начинается с одного символом @ в начале имени. Общий вид объявления переменной:

DECLARE {@local\_variable datatype} [,..n]

С помощью DECLARE можно объявлять и табличную переменную:

DECLARE @table TABLE (

Id int NOT NULL,

AUTHOR nvarcchar(max))

Пример установки переменной значения:

SET @имя = значение

Содержимое переменной можно использовать и в качестве имени транзакции.   
Глобальные переменные начинаются с двух символов @@.   
Существует несколько глобальных переменных, через которые можно определить полезную информацию о сервере или о времени выполнения.  
@@SERVERNAME – содержит имя экземпляра MS SQL Server;

@@VERSION – информация о версии SQL Server;

@@ROWCOUNT – содержит количество строк, обработанных последней выполненной командой;

@@SPID – идентификатор текущего процесса;

@@TRANCOUNT – уровень вложенности транзакций;

@@NESTLEVEL – уровень вложенности при выполнении хранимых процедур и триггеров;

@@OPTIONS – текущие настройки серверных параметров;

@@ERROR – содержит код ошибки последней выполненной команды;

@@IDENTITY – последнее значение счетчика, которое было использовано после добавления строки.

@@CURSOR\_ROWS – возвращает количество записей, полученных последней командой открытия курсора.

В реализациях языка [**SQL**](http://www.sql-tutorial.ru/ru/termins/sql.html) может быть выполнено неявное преобразование типов. Так, например, в [SQL Server](http://www.sql-tutorial.ru/ru/termins/sql_server.html) и Sybase ASE.Transact-SQL при сравнении или комбинировании значений типов smallint и int, данные типа smallint неявно преобразуются к типу int.

**Преобразование данных**

Имеется 2 функции для преобразования данных CAST и CONVERT

CAST ( expression AS data\_type [ ( length ) ] )

* expression - исходные данные
* type - datatype к которому мы хотим привести. Возможен один из вариантов: bigint, int, smallint, tinyint, bit, decimal, numeric, money, smallmoney, float, real, datetime, smalldatetime, char, varchar, text, nchar, nvarchar, ntext, binary, varbinary, or image.
* length - длина результирующего типа, такого как char, varchar, nchar, nvarchar, binary and varbinary.

Результат в CAST может округлиться или обрезаться

*Примеры*

SELECT CAST(14.85 AS int);  
*Result:* 14 *(result is truncated)*

SELECT CAST(14.85 AS float);  
*Result:* 14.85 *(result is not truncated)*  
  
SELECT CAST(15.6 AS varchar);  
*Result:* '15.6'  
  
SELECT CAST(15.6 AS varchar(4));  
*Result:* '15.6'  
  
SELECT CAST('15.6' AS float);  
*Result:* 15.6  
  
SELECT CAST('2014-05-02' AS datetime);  
*Result:* '2014-05-02 00:00:00.000'

CONVERT ( data\_type [ ( length ) ] , expression [ , style ] )

* *data\_type(length) -* тип в который преобразовать
* *expression -* исходные данные
* *style -* формат вывода для date/time

*Примеры*

1. CONVERT(VARCHAR(19),GETDATE())
2. CONVERT(VARCHAR(10),GETDATE(),10)
3. CONVERT(VARCHAR(10),GETDATE(),110)
4. CONVERT(VARCHAR(11),GETDATE(),6)
5. CONVERT(VARCHAR(11),GETDATE(),106)
6. CONVERT(VARCHAR(24),GETDATE(),113)
7. Nov 04 2014 11:45 PM
8. 11-04-14
9. 11-04-2014
10. 04 Nov 14
11. 04 Nov 2014
12. 04 Nov 2014 11:45:34:243

**Отличия**

Как видно, CONVERT более гибкая функция.

CAST - [ANSI-standard](http://searchdatacenter.techtarget.com/definition/ANSI), что означает большую совместимость с дркгими модификациями sql языка, но при этом меньшую гибкость .

CAST и CONVERT можно использовать вместе, например преобразовали дату в определенном формате,

SELECT CONVERT(CHAR(10), CURRENT\_TIMESTAMP, 102)

Теперь хотим явно указать тип, к которому привели дату. Допустим [datetime](http://searchsqlserver.techtarget.com/tip/Out-of-range-datetime-value-error-in-SQL-Server-when-working-with-dates) или smalldatetime:

SELECT CAST(CONVERT(CHAR(10),CURRENT\_TIMESTAMP,102) AS DATETIME