Why are my insert, update statements failing with the following error?  
  
Server: Msg 8152, Level 16, State 9, Line 1  
String or binary data would be truncated.  
The statement has been terminated.  
  
What is the T-SQL equivalent of IIF (immediate if/ternary operator) function of other programming languages?  
  
How to programmatically find out when the SQL Server service started?  
  
How to get rid of the time part from the date returned by GETDATE function?  
  
How to upload images or binary files into SQL Server tables?  
  
How to run an SQL script file that is located on the disk, using T-SQL?  
  
How to get the complete error message from T-SQL while error handling?  
  
How to get the first day of the week, last day of the week and last day of the month using T-SQL date functions?  
  
How to pass a table name, column name etc. to the stored procedure so that I can dynamically select from a table?  
  
Error inside a stored procedure is not being raised to my front-end applications using ADO. But I get the error when I run the procedure from Query Analyzer  
  
How to suppress error messages in stored procedures/triggers etc. using T-SQL?  
  
How to save the output of a query/stored procedure to a text file?  
  
How to join tables from different databases?  
  
How to join tables from different servers?  
  
How to convert timestamp data to date data (datetime datatype)?  
  
Can I invoke/instantiate COM objects from within stored procedures or triggers using T-SQL?  
  
Oracle has a rownum to access rows of a table using row number or row id. Is there any equivalent for that in SQL Server? Or how to generate output with row number in SQL Server?  
  
How to specify a network library like TCP/IP using ADO connect string?  
  
[How to generate scripts for repetitive tasks like truncating all the tables in a database, changing owner of all the database objects, disabling constraints on all tables etc?](http://vyaskn.tripod.com/generate_scripts_repetitive_sql_tasks.htm)Is there a way to find out when a stored procedure was last updated?  
  
[How to find out all the IDENTITY columns of all the tables in a given database?](http://vyaskn.tripod.com/sql_server_programming_faq_2.htm#q1)[How to search the code of stored procedures?](http://vyaskn.tripod.com/sql_server_search_stored_procedure_code.htm)[How to retrieve the generated GUID value of a newly inserted row? Is there an @@GUID, just like @@IDENTITY ?](http://vyaskn.tripod.com/retrieve_guid_value_like_identity.htm)

| Why are my insert, update statements failing with the following error?  Server: Msg 8152, Level 16, State 9, Line 1 String or binary data would be truncated. The statement has been terminated. <top> |
| --- |

This error occurs, when the length of the value entered by you into a char, varchar, nchar, nvarchar column is longer than the maximum length of the column. For example, inserting 'FAQ' into a char(2) column would result in this error.  
  
Profiler is handy in troubleshooting this error. If data truncation is okay with you and you don't want to see this error, then turn off ANSI WARNINGS by using the following SET command: SET ANSI\_WARNINGS OFF.   
  
Steps to reproduce the problem:  
  
CREATE TABLE MyTable  
(  
Pkey int PRIMARY KEY,  
Col1 char(10)  
)  
GO  
INSERT INTO MyTable (Pkey, Col1) VALUES (1, 'SQL Server Clustering FAQ')  
GO  
  
Make sure, you restrict the length of input, in your front-end applications. For example, you could use the MAXLENGTH property of the text boxes in HTML forms. E.g:  
  
<INPUT NAME = "Name" TYPE= TEXTBOX MAXLENGTH=20>

| What is the T-SQL equivalent of IIF (immediate if/ternary operator) function of other programming languages? <top> |
| --- |

CASE is the equivalent of IIF function. See SQL Server Books Online for more information. Here's a quick example:  
  
CREATE TABLE People  
(  
[ID] int PRIMARY KEY,  
[Name] varchar(25) NOT NULL,  
Sex bit NULL  
)  
  
INSERT INTO People ([ID],[Name], Sex) VALUES (1,'John Dykes', 1)  
INSERT INTO People ([ID],[Name], Sex) VALUES (2,'Deborah Crook', 0)  
INSERT INTO People ([ID],[Name], Sex) VALUES (3,'P S Subramanyam', NULL)  
  
SELECT [ID], [Name],   
    CASE Sex   
        WHEN 1   
            THEN 'Male'   
        WHEN 0   
            THEN 'Female'   
        ELSE 'Not specified'   
    END AS Sex  
FROM People

| How to programmatically find out when the SQL Server service started? <top> |
| --- |

Everytime SQL Server starts, it recreates the tempdb database. So, the creation date and time of the tempdb database tells us the date and time at which SQL Server service started. This information is stored in the crdate column of the sysdatabases table in master database. Here's the query to find that out:  
  
SELECT crdate AS 'SQL Server service started approximately at:'  
FROM master.dbo.sysdatabases  
WHERE name = 'tempdb'  
  
SQL Server error log also has this information (This is more accurate) and the error log can be queried using xp\_readerrorlog

| How to get rid of the time part from the date returned by GETDATE function? <top> |
| --- |

We have to use the CONVERT function to strip the time off the date. Any of the following commands will do this:  
  
SELECT CONVERT(char,GETDATE(),101)  
SELECT CONVERT(char,GETDATE(),102)  
SELECT CONVERT(char,GETDATE(),103)  
SELECT CONVERT(char,GETDATE(),1)  
  
See SQL Server Books Online for more information on CONVERT function.

| How to upload images or binary files into SQL Server tables? <top> |
| --- |

First of all, if possible, try not to stored images and other binary files in the SQL Server tables, as they slow things down. Instead, store a link (file path) to the file in the tables and let your applications directly access the files. But if you must store these files within SQL Server, use the text/ntext or image datatype columns and consider the following options:

* SQL Server 7.0 and 2000 come with a utility called textcopy.exe. You can locate this file in the Binn folder under your SQL Server installation folder. Run this file from command prompt, and it will prompt you for required input
* Use the GetChunk and AppendChunk methods of ADO Field object. MSDN has examples
* Use the ADO Stream object
* Use the Bulk Insert Image utility (BII) that ships with SQL Server 2000 (Can be found at \Program Files\Microsoft SQL  
  Server\80\Tools\Devtools\Samples\Utils)

| How to run an SQL script file that is located on the disk, using T-SQL? <top> |
| --- |

There's no direct command to read a script file and execute it. But the isql.exe and osql.exe come in handy when you have to execute a script file from within T-SQL. Just call any of these exes using xp\_cmdshell and pass the script file name as parameter to it. See SQL Server Books Online for more information about the input parameters of these exes. Here are some quick examples:  
  
EXEC master..xp\_cmdshell 'osql -Svaio -Usa -Pzaassds1 -ic:\MySQl.sql -n'  
EXEC master..xp\_cmdshell 'isql -Svaio -Usa -Pzaassds1 -ic:\MySQl.sql -n'  
  
See xp\_cmdshell in SQL Server Books Online if you are having permissions problems in getting this technique to work.

| How to get the complete error message from T-SQL while error handling? <top> |
| --- |

Unfortunately, the error handling capabilities of SQL Server are limited. When an error occurs, all you can get is the error number, using the @@ERROR global variable. There is no @@ERROR\_MESSAGE global variable to get the error description.   
  
For a complete error message, you can always query the master..sysmessages table using the error number, but most of these messages have place holders (like %s, %l etc.), and hence we can't get the complete error message.  
  
However, the client applications using an object model such as RDO, ADO have access to the complete error message.

| How to get the first day of the week, last day of the week and last day of the month using T-SQL date functions? <top> |
| --- |

Here's the code:  
  
DECLARE @Date datetime  
SET @Date = '2001/08/31'  
SELECT DATEADD(dd,-(DATEPART(dw, @Date) - 1),@Date) AS 'First day of the week'  
SELECT DATEADD(dd,-(DATEPART(dw, @Date) - 7),@Date) AS 'Last day of the week'  
SELECT DAY(DATEADD(d, -DAY(DATEADD(m,1,@Date)),DATEADD(m,1,@Date))) AS 'Last day of the month'

| How to pass a table name, column name etc. to the stored procedure so that I can dynamically select from a table? <top> |
| --- |

Basically, SELECT and other commands like DROP TABLE won't let you use a variable instead of a hardcoded table name. To overcome this problem, you have to use dynamic sql. But dynamic SQL has some disadvantages. It's slow, as the dynamic SQL statement needs to be parsed everytime it's executed. Further, the user who is executing the dynamic SQL string needs direct permissions on the tables, which defeats the purpose of having stored procedures to mask the underlying tables. Having said that, here are some examples of dynamic SQL: (Also see sp\_executesql in SQL Server Books Online)  
  
CREATE PROC DropTable  
@Table sysname  
AS  
EXEC ('DROP TABLE ' + @Table)  
GO  
  
EXEC DropTable 'MyTable'  
GO  
  
CREATE PROC SelectTable  
@Table sysname  
AS  
EXEC ('SELECT \* FROM ' + @Table)  
GO  
  
EXEC SelectTable 'MyTable'  
  
For a complete discussion on the pros and cons of dynamic SQL check out Erland's article:   
[The curse and blessings of dynamic SQL](http://www.algonet.se/~sommar/dynamic_sql.html)

| Error inside a stored procedure is not being raised to my front-end applications using ADO. But I get the error when I run the procedure from Query Analyzer <top> |
| --- |

This typically happens when your stored procedure is returning multiple resultsets and the offending SQL statement is executed after returning one or more resultsets. ADO will not return an error untill it processes all the recordsets returned before the offending SQL statement got executed. So, to get to the error message returned by your procedure. You have to loop through all the recordsets returned. ADO Recordset object has a method called NextRecordset, which lets you loop through the recordsets.   
  
Having SET NOCOUNT ON at the beginning of the procedure also helps avoid this problem. SET NOCOUNT ON also helps in improving the stored procedure performance. Here's a sample procedure to simulate the problem:  
  
CREATE PROC TestProc  
AS  
SELECT MAX(Col1) FROM TestTable  
SELECT MIN(Col1) FROM TestTable  
INSERT INTO TestTable (Col1, Col2) VALUES (1,'Oracle and SQL Server comparison')  
INSERT INTO TestTable (Col1, Col2) VALUES (1,'How to configure SQL Server?') -- Dupplicate key error occurs  
GO

| How to suppress error messages in stored procedures/triggers etc. using T-SQL? <top> |
| --- |

It's not possible to suppress error messages from within T-SQL. Error messages are always returned to the client. If you don't want your users to see these raw error messages, you should handle them in your front-end applications.  For example, if you are using ADO from ASP to connect to SQL Server, you would do something like the following:  
  
On Error Resume Next  
Set Rs = Conn.Execute ("INSERT INTO MyTable (1,'How to migrate from Oracle to SQL Server','Book'")  
If Err.Number <> 0 Then Response.Write ("Error occurred while inserting new data")  
On Error GoTo 0

| How to save the output of a query/stored procedure to a text file using T-SQL? <top> |
| --- |

T-SQL by itself has no support for saving the output of queries/stored procedures to text files. But you could achieve this using the command line utilities like isql.exe and osql.exe.  You could either invoke these exe files directly from command prompt/batch files or from T-SQL using the xp\_cmdshell command. Here are the examples:  
  
From command prompt:  
osql.exe -S YourServerName -U sa -P secretcode -Q "EXEC sp\_who2" -o "E:\output.txt"  
  
From T-SQL:  
EXEC master..xp\_cmdshell 'osql.exe -S YourServerName -U sa -P secretcode -Q "EXEC sp\_who2" -o "E:\output.txt"'  
  
Query Analyzer lets you save the query output to text files manually. The output of stored procedures that are run as a part of a scheduled job, can also be saved to a text file.  
  
BCP and Data Transformation Services (DTS) let you export table data to text files.

| How to join tables from different databases? <top> |
| --- |

You just have to qualify the table names in your SELECT queries with database name, followed by table owner name. In  the following example, Table1 from pubs database and Table2 from northwind database are being joined on the column i. Both tables are owned by dbo.  
  
SELECT a.i, a.j  
FROM pubs.dbo.Table1 a   
INNER JOIN   
northwind.dbo.Table2 b  
ON a.i = b.i  
GO

| How to join tables from different servers? <top> |
| --- |

To be able to join tables between two SQL Servers, first you have to link them. After the linked servers are setup, you just have to prefix your tables names with server name, database name, table owner name in your SELECT queries. The following example links SERVER\_01 to SERVER\_02. Execute the following commands in SERVER\_02:  
  
EXEC sp\_addlinkedserver SERVER\_01  
GO  
  
/\* The following command links 'sa' login on SERVER\_02 with the 'sa' login of SERVER\_01 \*/  
EXEC sp\_addlinkedsrvlogin @rmtsrvname = 'SERVER\_01', @useself = 'false', @locallogin = 'sa', @rmtuser = 'sa', @rmtpassword = 'sa password of SERVER\_01'  
GO  
  
SELECT a.title\_id  
FROM SERVER\_01.pubs.dbo.titles a  
INNER JOIN SERVER\_02.pubs.dbo.titles b  
ON a.title\_id = b.title\_id  
GO

| How to convert timestamp data to date data (datetime datatype)? <top> |
| --- |

The name timestamp is a little misleading. Timestamp data has nothing to do with dates and times and can not be converted to date data. A timestamp is a unique number within the database and is equivalent to a binary(8)/varbinary(8) datatype. A table can have only one timestamp column. Timestamp value of a row changes with every update of the row. To avoid the confusion, SQL Server 2000 introduced a synonym to timestamp, called rowversion.

| Can I invoke/instantiate COM objects from within stored procedures or triggers using T-SQL? <top> |
| --- |

Yes. SQL Server provides system stored procedures that let you instantiate COM objects using T-SQL from stored procedures, triggers and SQL batches. Search SQL Server Books Online for sp\_OACreate and sp\_OA\* for documentation and examples. Also check out my [code library](http://vyaskn.tripod.com/code.htm) for an example.

| Oracle has a rownum to access rows of a table using row number or row id. Is there any equivalent for that in SQL Server? Or how to generate output with row number in SQL Server? <top> |
| --- |

There is no direct equivalent to Oracle's rownum or row id in SQL Server. Strictly speaking, in a relational database, rows within a table are not ordered and a row id won't really make sense. But if you need that functionality, consider the following three alternatives:

* Add an IDENTITY column to your table. See Books Online for more information
* Use the following query to generate a row number for each row. The following query generates a row number for each row in the authors table of pubs database. For this query to work, the table must have a unique key.  
    
  SELECT        (SELECT COUNT(i.au\_id)   
                 FROM pubs..authors i   
                 WHERE i.au\_id >= o.au\_id ) AS RowID,   
                au\_fname + ' ' + au\_lname AS 'Author name'  
  FROM          pubs..authors o  
  ORDER BY      RowID
* Use a temporary table approach, to store the entire resultset into a temporary table, along with a row id generated by the IDENTITY() function. Creating a temporary table will be costly, especially when you are working with large tables. Go for this approach, if you don't have a unique key in your table. Search for IDENTITY (Function) in SQL Server Books Online.

For more ideas on this topic, [click here](http://support.microsoft.com/support/kb/articles/q186/1/33.asp) to read an informative article from Microsoft Knowledgebase.

| How to specify a network library like TCP/IP using ADO connect string? <top> |
| --- |

To specify TCP/IP net library, append the following to your ADO connect string:  
  
Network=dbmssocn  
  
For more information on specifying other net libraries in ADO connect strings, [click here](http://support.microsoft.com/support/kb/articles/Q238/9/49.ASP) to read the article from Microsoft Knowledgebase.

| Is there a way to find out when a stored procedure was last updated? <top> |
| --- |

Simple answer is 'No'. The crdate column in the sysobjects table always contains the stored procedure create date, not the last updated date. You can use Profiler to trace ALTER PROC calls to the database, but you can't really afford to run a trace for ever, as it's resource intensive. Here is a simple idea! Whenever you have to alter your stored procedure, first drop it, then recreate it with the updated code. This resets the crdate column of sysobjects table. If you can make sure your developers always follow this plan, then the crdate column of sysobjects will always reflect the last updated date of the stored procedure. For example, if I have to modify a procedure named MyProc, instead of doing "ALTER PROC MyProc", here's what I would do:  
  
- Use sp\_helptext to get the current code of MyProc.   
- Change the code as needed.  
- Run the following code to drop the existing version of MyProc:  
  
IF EXISTS(SELECT 1 FROM sysobjects WHERE name = 'MyProc' AND type = 'P' AND USER\_NAME(uid) = 'dbo')  
BEGIN  
DROP PROC dbo.MyProc  
END  
  
- Run the updated code to recreate MyProc  
  
There is a much more powerful way out, if you can use Visual Source Safe (VSS). VSS is a version control software, that lets you manage your code. With VSS in place, you will have to maintain all your object creation scripts as script files and check them into VSS. When you have to modify a particular stored procedure, check out that script from VSS, modify it, test it, create the stored procedure, and check the script back into VSS. VSS can show you when a script got modified, by who and a whole lot of other information.  
  
Advantages of using VSS  
- You can version control your software, as VSS maintains all your changes as different versions  
- You can go back to a previous known good version of your stored procedure, if a developer makes a mistake  
- Using the labelling feature, you can revert back to an entire set of scripts at a particular point in time  
- You can control access to your source code by configuring permissions to your developers  
- By maintaining backups of VSS database, you can secure all your code centrally, instead of worrying about individual script files