**Execute Dynamic SQL commands in SQL Server**

**Problem**

In some applications having hard coded SQL statements is not appealing, because of the dynamic nature of the queries being issued against the database server. Because of this sometimes there is a need to dynamically create a SQL statement on the fly and then run that command.  This can be done quite simply from the application perspective where the statement is built on the fly whether you are using ASP.NET, ColdFusion or any other programming language.  But how do you do this from within a SQL Server stored procedure?

**Solution**

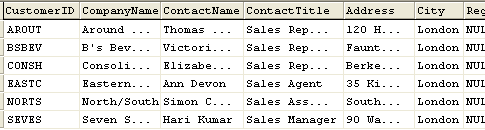
SQL Server offers a few ways of running a dynamically built SQL statement.  These ways are:

1. Writing a query with parameters
2. Using [EXEC](http://msdn2.microsoft.com/en-us/library/ms189915.aspx)
3. Using [sp\_executesql](http://msdn2.microsoft.com/en-us/library/ms188001.aspx)

**1. Writing a query with parameters**

This first approach is pretty straight forward if you only need to pass parameters into your WHERE clause of your SQL statement.  Let's say we need to find all records from the customers table where City = 'London'.  This can be done easily such as the following example shows.

DECLARE @city varchar(75)  
SET @city = 'London'  
SELECT \* FROM customers WHERE City = @city



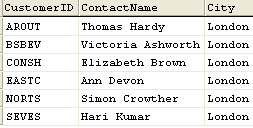
**2. Using EXEC**

With this approach you are building the SQL statement on the fly and can pretty much do whatever you need to in order to construct the statement.  Let's say we want to be able to pass in the column list along with the city.

For this example we want to get columns CustomerID, ContactName and City where City = 'London'.

As you can see from this example handling the @city value is not at straight forward, because you also need to define the extra quotes in order to pass a character value into the query.  These extra quotes could also be done within the statement, but either way you need to specify the extra single quotes in order for the query to be built correctly and therefore run.

DECLARE @sqlCommand varchar(1000)  
DECLARE @columnList varchar(75)  
DECLARE @city varchar(75)  
SET @columnList = 'CustomerID, ContactName, City'  
SET @city = '''London'''  
SET @sqlCommand = 'SELECT ' + @columnList + ' FROM customers WHERE City = ' + @city  
EXEC (@sqlCommand)



**3. sp\_exectesql**

With this approach you have the ability to still dynamically build the query, but you are also able to still use parameters as you could in example 1. This saves the need to have to deal with the extra quotes to get the query to build correctly.  In addition, with using this approach you can ensure that the data values being passed into the query are the correct datatypes.

DECLARE @sqlCommand nvarchar(1000)  
DECLARE @columnList varchar(75)  
DECLARE @city varchar(75)  
SET @columnList = 'CustomerID, ContactName, City'  
SET @city = 'London'  
SET @sqlCommand = 'SELECT ' + @columnList + ' FROM customers WHERE City = @city'  
EXECUTE sp\_executesql @sqlCommand, N'@city nvarchar(75)', @city = @city

