**DB Coding/Standards & Project Expectations**

**Best practices**

* Use the brackets around objects, so the query engine excplicitly knows a field when it sees it
* Use THE SAME CASE as table object names and field names
* When calling SPs from application, use the fully qualified [dbo].[procName] with correct owner AND case. Not Kidding! Read the articles above!
* Reference the owner of the object so security is explicitly known and doesn't have to be figured out
* DON'T us "sp\_" as this refers to system stored procs, and overhead
* Use SET NOCOUNT ON and SET NOCOUNT OFF to eliminate the extra overhead to keep track of how many records are updated in the stored proc unless you need them. Normally, you don't and you can gain a huge increase in performance.

**Preferences**

* Prefix stored procs with proc
* Suffix every stored proc with SEL, UPD, DEL, INS (or SELECT, UPDATE, DELETE, INSERT)
* Capitalize reserved words
* Main keywords on new line (scripting)
* Use commas before columns (scripting)
* Prefix views with vw
* Don't prefix tables
* Table names singular
* Add a suffix to the standard names like "\_ByPK", "\_OrderByLastName", or "\_Top15Orders" for variations on the stock SP

**Naming**

**Tables:** Rules: Pascal notation; end with an ‘s’

* Examples: Products, Customers
* Group related table names1

**Stored Procs:** Rules: sp<App Name>\_[<Group Name >\_]<Action><table/logical instance>

* Examples: spOrders\_GetNewOrders, spProducts\_UpdateProduct

**Triggers:** Rules: TR\_<TableName>\_<action>

* Examples: TR\_Orders\_UpdateProducts
* Notes: The use of triggers is discouraged

**Indexes:** Rules: IX\_<TableName>\_<columns separated by \_>

* Examples: IX\_Products\_ProductID

**Primary Keys:** Rules: PK\_<TableName>

* Examples: PK\_Products

**Foreign Keys:** Rules: FK\_<TableName1>\_<TableName2>

* Example: FK\_Products\_Orderss

**Defaults:** Rules: DF\_<TableName>\_<ColumnName>

* Example: DF\_Products\_Quantity

**Columns:** If a column references another table’s column, name it <table name>ID

* Example: The Customers table has an ID column
* The Orders table should have a CustomerID column

**General Rules:**

* Do not use spaces in the name of database objects
  + Do not use SQL keywords as the name of database objects
  + In cases where this is necessary, surround the
* object name with brackets, such as [Year]
* Do not prefix stored procedures with ‘sp\_’2
* Prefix table names with the owner name3

**Structure**

* Each table must have a primary key
  + In most cases it should be an IDENTITY column named ID
* Normalize data to third normal form
  + Do not compromise on performance to reach third normal form. Sometimes, a little de-normalization results in better performance.
* Do not use TEXT as a data type; use the maximum allowed characters of VARCHAR instead
* In VARCHAR data columns, do not default to NULL; use an empty string instead
* Columns with default values should not allow NULLs
* As much as possible, create stored procedures on the same database as the main tables they will be accessing

**Formatting**

* Use upper case for all SQL keywords
  + SELECT, INSERT, UPDATE, WHERE, AND, OR, LIKE, etc.
* Indent code to improve readability
* Comment code blocks that are not easily understandable
  + Use single-line comment markers(–)
  + Reserve multi-line comments (/\*.. ..\*/) for blocking out sections of code
* Use single quote characters to delimit strings.
  + Nest single quotes to express a single quote or apostrophe within a string
    - For example, SET @sExample = ‘SQL”s Authority’
* Use parentheses to increase readability
  + WHERE (color=’red’ AND (size = 1 OR size = 2))
* Use BEGIN..END blocks only when multiple statements are present within a conditional code segment.
* Use one blank line to separate code sections.
* Use spaces so that expressions read like sentences.
  + fillfactor = 25, not fillfactor=25
* Format JOIN operations using indents
  + Also, use ANSI Joins instead of old style joins4
* Place SET statements before any executing code in the procedure.

**Reference:**

1) Group related table names:

Products\_USA

Products\_India

Products\_Mexico

2) The prefix sp\_ is reserved for system stored procedures that ship with SQL Server. Whenever SQL Server encounters a procedure name starting with sp\_, it first tries to locate the procedure in the master database, then it looks for any qualifiers (database, owner) provided, then it tries dbo as the owner. Time spent locating the stored procedure can be saved by avoiding the “sp\_” prefix.

3) This improves readability and avoids unnecessary confusion. Microsoft SQL Server Books Online states that qualifying table names with owner names helps in execution plan reuse, further boosting performance.

4)

*False code:*

SELECT \*

FROM Table1, Table2

WHERE Table1.d = Table2.c

*True code:*

SELECT \*

FROM Table1

INNER JOIN Table2 ON Table1.d = Table2.c