General

- No more than 200 lines (formatted) allowed in any stored procedure, user-defined function or view. *If more than 200 lines is unavoidable, please discuss this with a Senior Developer or Database Administrator first.*
- Aliases should be used for tables in SELECTs for readability purposes and alias names should make sense. If alias is too long (>10 characters) , use an abbreviation that makes sense.
- **EditedUser** in a SQL batch job is the name of the stored procedure where it is called from. When the edited user is provide to the procedure the SP name must be appended to e.g. {Username}_MyProc. This must be applied to any parent procedure that perform data modification.
- Script name assigned correctly http://shakespeare/MasterBuilder/Tools/GenerateScript when committing your scripts.
- *Stored Procedure, Views and Functions heading convention* followed (see example below)

> 2018/11/30 : dbotha : 56789 : Added check for qualifications offered by my learning provider.
> {Date} : {Author} : {TP#} Short concise description of change

- **UPDATE** statements should always contain a **WHERE** clause.
- **WITH(NOLOCK)** should be used in SELECTs to prevent unnecessary locking.
- **SQL KEYWORDS** should always be in **CAPS.**
- Dynamic SQL should only be used when absolutely necessary.
- The **SchemaId** column in tables must not be nullable.
- The statement **SET NOCOUNT ON** should be at the top of the stored procedure unless in the unlikely case where the counts obtained from the stored procedure is used in code.
- When performing an INSERT, always specify the column list

> **Correct** : INSERT INTO tmp (Value) SELECT @variable INSERT INTO tmp (Value) VALUES(@variable)
>
> **Not correct** : INSERT INTO tmp SELECT @variable INSERT INTO tmp VALUES(@variable)

- When writing an automation stored procedure in SQL and the code becomes too complex and long-winded, break the stored procedure into multiple stored procedures.
- Using in-line user defined functions in SELECT statements should only be used if absolutely necessary. NOTE : Complex user-defined functions used in-line in large SELECT statements returning many rows (1000+) can potentially slow down your code significantly. In-line user defined functions should be tested on large amounts of data.

Revision #10
Created 17 September 2020 02:10:36 by Theuns Pretorius
Updated 25 June 2021 14:01:58 by Carl