



Microsoft®
SQL Server®

ESTANDAR DE PROGRAMACIÓN SQL SERVER

Versión 1.0

Arquitectura y Estándares TI

04/07/2019



Tabla de contenido

1. Objetivos	1
2. Normas Generales de nomenclatura de Base de Datos	1
4. Bases de Datos	1
5. Schemas.	2
6. Permisos.	2
7. Cadenas de Conexión.....	2
8. Definición de Variables, parámetros y tipos de datos.	3
9. Tablas.	4
10. Constraints.	6
11. Índices.	7
12. Triggers.	7
13. Stored Procedures.	7
14. Vistas.....	9
15. Programación.	10
16. BackUp Tablas Transaccionales.	11
17. Creación de Usuarios.	11
18. Restricciones y Notas.....	11

1. Objetivos

- Estandarizar la nomenclatura de objetos de base de datos.
- Estandarizar la estructura de codificación del SQL Transact.

2. Normas Generales de nomenclatura de Base de Datos

Dentro de algunas normas generales tenemos:

- Mantener nombres cortos y descriptivos (por ejemplo, CLIENTE)
- Mantener nombres de objetos únicos, por ejemplo, evitar crear la tabla SESSION y un rol o vista con el mismo nombre.
- No se permiten los espacios en blanco ni puntos(.) como parte de un identificador.
- Para la definición de nombre de objetos de base de datos de acuerdo al caso se usará el carácter underscore “_” para separar las palabras_del_nombre, por ejemplo, HISTORIAL_SESSION.

3. Estándar de Configuración de instalación de Base de Datos.

- **Collation:** Latin1_General_BIN
- **Sort Order:** Binary Order, Case Sensitive.

Particiones de disco asignadas para la creación de BDD:

Unidad <C:> Unidad exclusiva para el sistema operativo

Unidad <D:> Unidad para Archivos/Logs/BackUps

Unidad <E:> Solo archivos Mdf y Ndf

Unidad <L:> Solo archivos Ldf

Por ejemplo:

```
CREATE DATABASE [API_BCP] ON PRIMARY
( NAME = N'API_BCP_data', FILENAME = N'E:\SQLData\API_BCP_data.mdf' , SIZE =
3072KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )
LOG ON
( NAME = N'API_BCP_log', FILENAME = N'L:\SQLLog\API_BCP_log.ldf' , SIZE = 1024KB
, MAXSIZE = 2048GB , FILEGROWTH = 10%)
GO
```

4. Bases de Datos

Nomenclatura:

El nombre de la base de datos debe estar en **MAYÚSCULAS**.

Ejemplo:

Base de Datos	:	API_BCP
Archivo MDF	:	API_BCP_data
Archivo LDF	:	API_BCP_log

5. Schemas.

Cada aplicación debe tener un esquema asociado y el usuario de conexión tener permisos al esquema

El esquema debe crearse en minúsculas (por ejemplo: api)

```
/***** Object: Schema [api]    Script Date: 08/05/2019 17:49:28 *****/  
CREATE SCHEMA [api]  
GO
```

6. Permisos.

El ABM de las tablas de la aplicación debe ser realizada a través de Store Procedures y a su vez el usuario de conexión a la BDD debe tener permisos específicos/granulares en función del perfil de conexión

```
use [API_BCP]  
GO  
GRANT EXECUTE ON [api].[USUARIO_Add] TO [UsrBD_API]  
GO  
  
use [API_BCP]  
GO  
GRANT EXECUTE ON [api].[USUARIO_Delete] TO [UsrBD_API]  
GO
```

7. Cadenas de Conexión

Las cadenas de conexión a base de datos deben tener la siguiente estructura

Conexión Integrada por grupo de dominio

Data Source=BTBSCD10;Initial Catalog=BIZTALK;Integrated Security=True;Application Name = [Nombre_del_aplicativo];

Donde:

Data Source:	Nombre del servidor de Base de datos
Initial Catalog:	Nombre de la Base de Datos
Integrated Security:	“True” se utiliza las credenciales del usuario de dominio para la conexión
Application Name:	El nombre de la aplicación bajo la estructura XXXX_Nombre_del_aplicativo

Donde XXXX:

- WINS_Nombre_de_aplicativo .- Para aplicaciones que corren como servicio Windows
- WEBS_Nombre_de_aplicativo.- Para aplicaciones web y web services.
- EJEC_Nombre_de_aplicativo.- Para ejecutables.

Conexión con usuario de base de datos SQL

Data Source=DEVDBR00;Initial Catalog=RepExt; User ID=UshrBD_HUB_BIZTALK;
Password=myPassword; Name = [Nombre_del_aplicativo];

Donde:

Data Source: Nombre del servidor de Base de datos
User ID: Nombre del usuario de Base de datos
Initial Catalog: Nombre de la Base de Datos
Password: El password del usuario SQL que se conectara.
Application Name: El nombre de la aplicación

Donde:

- WINS_Nombre_de_aplicativo.- Para aplicaciones que corren como servicio Windows
- WEBS_Nombre_de_aplicativo.- Para aplicaciones web y web services.
- EJEC_Nombre_de_aplicativo.- Para ejecutables.

El parámetro Application Name nos permite identificar el nombre de la aplicación que se conecta al motor de base de datos y es de bastante utilidad al momento de realizar un profiler para monitoreo, troubleshooting, migraciones, etc.

S...	U...	Login	Database	Tas...	Com...	Application	Wait Tim...	Wait...	Wat...	B...	H...	Me...	Host Name	Wor...
130	1	BTBNET\S69593	BD_Banca_Movil2			Microsoft SQL Server Management Studio - Qu...	0						16 W204SISF35131A	default
131	1	BTBNET\S69593	BD_Banca_Movil2			Microsoft SQL Server Management Studio - Qu...	0						16 W204SISF35131A	default
82	1	BTBNET\HMC1	Netwix_CommonDB			Netwix Auditor	0						16 W204SISF32052	default
62	1	BTBNET\UshrAgtSqlSIS00Des	msdb			SQLAgent - Alert Engine	0						16 DEVDBS10	default
57	1	BTBNET\UshrAgtSqlSIS00Des	msdb			SQLAgent - Email Logger	0						16 DEVDBS10	default
56	1	BTBNET\UshrAgtSqlSIS00Des	msdb			SQLAgent - Generic Refresher	0						16 DEVDBS10	default
61	1	BTBNET\UshrAgtSqlSIS00Des	msdb			SQLAgent - Job invocation engine	0						16 DEVDBS10	default
77	1	BTBNET\UshrAgtSqlSIS00Des	BDCONCILIA	RUNNING	SELECT	SQLAgent - TSQL Job Step (Job 0x8388B20C5...	0						16 DEVDBS10	default
109	1	BTBNET\FEM0	BCP_ACH			Test	0						0 W204SISF32101	default

8. Definición de Variables, parámetros y tipos de datos.

Aplica a:

- Stored procedures
- Triggers
- Columnas de Tabla

Nomenclatura:

@YYYY_XX

YYYY: Nombre de la Variable

XX: Identificador del tipo de dato

Ejemplo:

@ NOMBRE_VC
@ CONDICIONES_USO_BT
@ FECHA_NACIMIENTO_DA

Nota: Los nombres son igual a las columnas de la tabla para facilitar el uso del Profiler, en caso de que la variable no se utilice para modificar alguna columna de la tabla quitar el identificador, por ejemplo: @FLAG_BT

TIPO DE DATO	IDENTIFICADOR
Bigint	BI
Int	IN
Smallint	SI
Tinyint	TI
Bit	BT
Decimal	DC
Numeric	NU
Money	MO
Float	FL
Real	RE
Datetime	DT
Smalldatetime	SD
Date	DA
Time	TM
Char	CH
Varchar	VC
Text	TX
Nchar	NC
Nvarchar	NV
Ntext	NT
Binary	BY
Varbinary	VY
Image	IM
Timestamp	TS
Uniqueidentifier	UN
Xml	XM

9. Tablas.

Aplica a:

- Tablas
- Tablas temporales

Nomenclatura:

El nombre de la tabla debe ser descriptivo, en singular y en MAYÚSCULAS.

Las Tablas identifican una entidad del Sistema con un nombre completo.

Las palabras deben ser separadas por un caracter underscore “_”.

Ejemplo:

CUENTA
CUENTA_DETALLE

Los nombres de las **columnas** ser descriptivos y en mayúsculas. El nombre de la columna debe incluir el tipo de dato del campo. Por ejemplo, para el campo MONEDA de tipo varchar el nombre de la columna deberá ser MONEDA_VC

Ejemplo:

```
CREATE TABLE [api].[CUENTA](
    [CUENTA_ID_IN] [int] IDENTITY(1,1) NOT NULL,
    [NUMERO_VC] [varchar](20) NULL,
    [SALDO_DC] [decimal](16, 2) NULL,
    [MONEDA_VC] [varchar](3) NULL,
    [BLOQUEO_VC] [varchar](1) NULL,
    [HABILITADO_BT] [bit] NULL,
    [USUARIO_CREACION_VC] [varchar](20) NULL,
    [FECHA_CREACION_DT] [datetime] NULL,
    [USUARIO_MODIFICACION_VC] [varchar](20) NULL,
    [FECHA_MODIFICACION_DT] [datetime] NULL,
    [ESTADO_BT] [bit] NOT NULL)
```

Nota: Cuando se creen tablas temporales añadir el prefijo TEMP_XXXX para reconocerlas. Por ejemplo [api].[TEMP_CUENTA]

TIPO DE DATO	IDENTIFICADOR
Bigint	BI
Int	IN
Smallint	SI
Tinyint	TI
Bit	BT
Decimal	DC
Numeric	NU
Money	MO
Float	FL
Real	RE
Datetime	DT
Smalldatetime	SD
Date	DA
Time	TM
Char	CH
Varchar	VC
Text	TX
Nchar	NC
Nvarchar	NV
Ntext	NT

Binary	BY
Varbinary	VY
Image	IM
Timestamp	TS
Uniqueidentifier	UN
Xml	XM

10. Constraints.

- **Nomenclatura Primary Key:** PK_NombreTabla
- **Nomenclatura Foreign Key:** FK_NombreTablaOrigen_NombreTablaReferenciada
- **Unique:** UQ_NemónicoTabla_NombreUnique
- **Default:** DF_NemónicoTabla_NombreColumna
- **Check:** CK_NemónicoTabla_NombreCheck

Ejemplo:

PK_CLIENTE
FK_FACTURA_CLIENTE
UQ_CLIENTE_CODIGO_CLIENTE
DF_CLIENTE_FECHA
CK_CLIENTE_CODIGO_CLIENTE

11. Índices.

Nomenclatura:

I_XXX_YYYY....

XXX : Tipo de índice (Unique(AK), Clustered(CL), NonClustered(NCL))
YYY : Nombre del Índice

Ejemplo:

I_AK_CNTA_NUMERO_VC (índice Unique)
I_CL_CCUSTODIA (índice Clustered)
I_NCL_CODDEPOSITANTE (índice NonClustered)

12. Triggers.

Nomenclatura:

TR_XXXX_YYYY.....

XXXX: Nombre de la tabla
YYYY: Acción realizada

Ejemplo:

TR_NOMBRETABLA_Delete (Cuando se realiza una eliminación en la tabla).

TR_NOMBRETABLA_Update (Cuando se realiza una actualización en la tabla).

TR_NOMBRETABLA_Insert (Cuando se realiza una inserción en la tabla).

13. Stored Procedures.

Nomenclatura:

NOMBRETABLA_ACCION

El nombre del store procedure debe estar compuesto por el nombre de la tabla underscore “_” y la acción a realizar.

Por ejemplo:

Insert:

NOMBRETABLA_Add (Inserta un registro a la tabla)

Update:

NOMBRETABLA_Update (Actualiza un registro a la tabla)

Delete:

NOMBRETABLA_Delete (Elimina un registro de la tabla, física o cambia al estado)

Select:

NOMBRETABLA_GetByID (Retorna un sólo registro por PK o AK).

NOMBRETABLA_ListByXXXX (Retornan más de un registro)

NOMBRETABLA_ListXXXX

Nota: No todos los stored procedures siguen esta convención, dado que algunos abstraen un mayor nivel de funcionalidad como ejecutar un proceso o algún query complejo.

Ejemplo:

EMPLEADO_Add

EMPLEADO_Insert

EMPLEADO_Update

EMPLEADO_GetByID

EMPLEADO_ListByEmpresa

EMPLEADO_ProcesaSueldo

Nota: Los nombres de los Stored Procedures **NO** deben comenzar con “sp”, esto porque generalmente el SQL piensa que son system procedures y los busca primero en la Base de Datos master.

Por otro lado, SET NOCOUNT ON (elimina la notificación del nro. de registros afectados por cada sentencia SQL lo cual incrementa el performance).

Estructura del Stored Procedure:

- Identificador, nombre de stored procedure., y parámetros
- Comentarios:
 - Descripción: “funcionalidad del stored procedure”
 - Fecha de creación
 - Fecha de modificación
 - Parámetros
 - Autor:
 - Versión:
 - Cambios Importantes
- Declaración Variables locales
 - Mayúsculas
- Sentencias SQL
 - Palabras del lenguaje SQL, y funciones de sistema en MAYUSCULAS, columnas y otras variables en Mayúsculas.
 - Sentencias legibles e indentadas (cada clausula SQL en una línea nueva)

Ejemplo:

```
CREATE PROCEDURE [bill].[DISPOSITIVO_Insert] @DSPT_ID_VC
VARCHAR(150)
,@DSPT_TIPO_VC VARCHAR(10)
AS
/*****
*Descripcion:      Procedimiento para ingresar nuevos dispositivos
*Fecha Crea:      17/06/2019
*Fecha Mod:
*Parametros:

                @DSPT_ID_VC -> Id del dispositivo
                @DSPT_TIPO_VC -> Tipo del dispositivo

*Autor:                        Braian Espejo
*Versión:                      BETA
*Cambios Importantes:

*****/
BEGIN

<Declaración de variables>
<Sentencias SQL>

END
```

14. Vistas.

Nomenclatura:

VW_YYY....

VW: Identificador para las Vistas

YYY: Nombre de la Vista en Mayúsculas

Ejemplo:

VW_CONSULTAR_PRODUCTO

Para el nombre de las vistas utilizar verbos en Infinitivo

Nota: La estructura de la vista debe ser similar a la del stored procedure.

15. Programación.

Utilizar **MAYÚSCULAS** para las sentencias propias del SQL

Ejemplo:

```

CREATE PROCEDURE [bill].[DISPOSITIVO_Insert] @DSPT_ID_VC VARCHAR(150)
, @DSPT_TIPO_VC VARCHAR(10)
AS
/*****
*Descripcion:      Procedimiento para ingresar nuevos dispositivos
*Fecha Crea:       17/06/2019
*Fecha Mod:
*Parametros:

                @DSPT_ID_VC -> Id del dispositivo
                @DSPT_TIPO_VC -> Tipo del dispositivo

*Autor:            Braian Espejo
*Versión:          BETA
*Cambios Importantes:
*****/
BEGIN
    UPDATE bill.DISPOSITIVO
    SET DSPT_ESTADO_BT = 0
    WHERE DSPT_ID_VC = @DSPT_ID_VC
        AND DSPT_TIPO_VC = @DSPT_TIPO_VC

    INSERT INTO bill.DISPOSITIVO (
        DSPT_ID_VC
        , DSPT_TIPO_VC
        , DSPT_PUSH_ID_VC
        , DSPT_HABILITADO_BT
        , DSPT_HABILITADO_PUSH_BT
        , DSPT_USUARIO_CREACION_VC
        , DSPT_FECHA_CREACION_DT
        , DSPT_USUARIO_MODIFICACION_VC
        , DSPT_FECHA_MODIFICACION_DT
        , DSPT_ESTADO_BT
    )
    VALUES (
        @DSPT_ID_VC
        , @DSPT_TIPO_VC
        , ''
        , 1
        , 0
        , 'UsrBD_BILETERA'
        , GETDATE()
        , 'UsrBD_BILETERA'
        , GETDATE()
        , 1
    )

    SELECT @@IDENTITY AS [@@IDENTITY]
END

```

Nota: Para indentar se deberá utilizar <https://poorsql.com/>

16. BackUp Tablas Transaccionales.

Para aplicativos transaccionales, se debe manejar tablas históricas por meses (Hasta 6 meses), bajo la siguiente estructura.

Nomenclatura:

NOMBRE_TABLA_AAAAMM

AAAA: Año

MM: Mes

Ejemplo:

TBL_LOG_TRANSACCION_201905 (MES)

TBL_LOG_TRANSACCION (7 DIAS)

Se adjunta un ejemplo para la generación de tablas históricas:



GeneraBackUpDiario.sql

17. Creación de Usuarios.

Para la creación de usuarios, usar la siguiente nomenclatura:

Nomenclatura:

UsrBD_NOMBRE_APLICACION

Donde:

NOMBRE_APLICACION: Describe el nombre de la aplicación a la cual el usuario tiene acceso.

Ejemplo

UsrBD_APIBCP

18. Restricciones y Notas.

No es permitido el uso de cursores

No es permitido el uso de LinkedServer

No es permitido el uso de scripts anidados *Mediante Transact SQL utilizar una base de datos del mismo servidor.

No es permitido dar permisos de AUTHORIZATION, OWNER, CONTROL a los esquemas