

	<b>Nomenclatura de Bases de Datos</b>	<b>Código:</b>	
		<b>Revisión:</b>	00
		<b>Fecha:</b>	13/02/2017

## 1. OBJETIVO

Establecer un estándar en la nomenclatura de objetos utilizados en base de datos. Comprende la nomenclatura de tablas, campos, esquemas, sinónimos, procedimientos, funciones, triggers, constraints, índices y vistas.

## 2. ALCANCE

La implementación de los estándares en este manual es de seguimiento obligatorio para todas las Bases de Datos creadas para MS SQL Server, en la Gerencia de Desarrollo y Programación de Avior Airlines.

## 3. RESPONSABLES

**Administrador de Base de Datos:** Definir el contenido de este documento.

**Director de Informática:** Revisar los cambios a nivel técnico y procedimental de este documento.

**Vicepresidente de Informática:** Aprobar los cambios a nivel técnico y procedimental de este documento.

**Programador:** cumplir con la estandarización presentada en el presente documento

## 4. REFERENCIAS

## 5. DEFINICIONES

**Tabla:** Las tablas son objetos de base de datos que contienen todos sus datos. En las tablas, los datos se organizan en filas y columnas. Cada fila representa un registro único y cada columna un campo dentro de un registro.

**Columnas:** una columna de una tabla representa algún atributo del objeto representado por la tabla.

<b>Código:</b>		<b>Nomenclatura de Bases de Datos</b>	
<b>Revisión:</b>	00		
<b>Fecha:</b>	13/02/2017		

**Filas:** una fila representa una única repetición del objeto representado por la tabla.

**Primary keys:** Una clave Primaria (PK) es una columna o una combinación de columnas cuyos valores identifican de forma única cada fila de la tabla.

**Foreign keys:** Una clave externa (FK) es una columna o combinación de columnas que se utiliza para establecer y exigir un vínculo entre los datos de dos tablas.

**Índices:** Un índice contiene claves generadas a partir de una o varias columnas de la tabla o la vista y punteros que asignan la ubicación de almacenamiento de los datos especificados. Se puede mejorar notablemente el rendimiento de las aplicaciones y consultas de bases de datos creando índices correctamente diseñados para que sean compatibles con las consultas.

**Esquema:** Un esquema es un contenedor con nombre para objetos de base de datos, que permite agrupar objetos en espacios de nombres independientes.

**Sinónimos:** Un sinónimo es un objeto de base de datos que sirve para los siguientes objetivos: Proporciona un nombre alternativo para otro objeto de base de datos, que puede existir en un servidor local o remoto y proporciona una capa de abstracción que protege una aplicación cliente de cambios realizados en el nombre o la ubicación del objeto base.

**Procedimientos:** Un Procedimiento Almacenado es un conjunto de sentencias SQL que han sido creadas anteriormente y almacenadas en un servidor de bases de datos.

**Funciones:** Una función definida por el usuario es una rutina de Transact-SQL que acepta parámetros, realiza una acción y devuelve el resultado de esa acción como un valor.

**Triggers:** Los triggers o disparadores de base de datos son bloques de código almacenados asociados a una tabla que se ejecutan o disparan automáticamente cuando se producen ciertos eventos o sucesos que afectan a la tabla (inserción, borrado o modificación de filas).

**Vistas:** una vista es una consulta que se presenta como una tabla (virtual) a partir de un conjunto de tablas en una base de datos relacional. Las vistas tienen la misma estructura que una tabla: filas y columnas. La única diferencia es que sólo se almacena de ellas la definición.

**Pascal Case:** o Notación Pascal. Una palabra con la primera letra en mayúsculas, y la primera letra de cada palabra subsecuente también en mayúsculas. Ejemplo: PrimerNombre.

	<b>Nomenclatura de Bases de Datos</b>	<b>Código:</b>	
		<b>Revisión:</b>	00
		<b>Fecha:</b>	13/02/2017

**Integridad referencial:** indica que las relaciones entre las tablas se mantienen adecuadamente. Los datos de una tabla sólo deben apuntar a filas existentes de otra tabla y no pueden apuntar a filas inexistentes.

## 6. DESARROLLO/INSTRUCCIONES/LINEAMIENTOS

### GUÍAS GENÉRICAS Y BUENAS PRÁCTICAS

- Utilizar nombres en castellano para todos los elementos de la base de datos, tablas, vistas, campos, etc.
- Los nombres de objetos deben ser lo más cortos posible, fáciles de leer, y lo más descriptivos posible, evitando términos ambiguos o que se presten a distintas interpretaciones.
- Los nombres deben incluir sólo caracteres del alfabeto español excepto vocales con acento, eñes, y diéresis, y no deben utilizarse caracteres especiales ('#', '/', ';', '%', '+', '-', etc.) ni espacios. El primer carácter del nombre debe ser una letra.
- No nombrar campos que representan lo mismo de forma distinta. La forma en que se nombran iguales propiedades debe ser consistente en todo un esquema. Ejemplo: Nombrar al campo clave de la tabla Empleado como Id, y después referenciarlo en otras tablas como Empleadold es inconsistente, el campo debe ser nombrado Empleadold en todos los casos que se quiera almacenar una clave de Empleado.
- Evitar tener demasiadas columnas NULLABLES en una tabla. Esto es indicio de un esquema poco o nada normalizado. Falta de normalización puede conllevar problemas de consistencia en los datos en la medida que un mismo campo se puede terminar almacenando en varias tablas. Excesiva normalización puede tener asociada una pérdida de performance en ciertas operaciones sobre la base de datos. Es necesario encontrar el equilibrio correspondiente a los requerimientos de cada

<b>Código:</b>		<b>Nomenclatura de Bases de Datos</b>	
<b>Revisión:</b>	00		
<b>Fecha:</b>	13/02/2017		

proyecto en este punto. Como regla general la tercera forma normal es un buen punto intermedio.

- No tener tablas sin definición de claves primarias.
- Evitar tener tablas innecesarias en el sistema. Un buen diseño es uno simple.
- Documentar significado de los nombres de los campos, si son campos que se usen para clasificar colocar en la descripción del campo y significado de cada combinación.

### Comentarios en Funciones y Procedimientos

```
-- =====
-- Author: Carolina Coronado
-- Create date: 15-10-2017
-- Description: Selecciona y retorna los datos de clientes clasificados por tipos
-- =====
```

Los comentarios en el código siempre seguirán el siguiente estándar:

- Al inicio del código se debe indicar:
- Nombre de quien lo escribió
- Fecha de escritura
- Breve descripción de lo que realiza ese código y que retorna
- Breve descripción de cada uno de los parámetros de entrada, si los hay

En caso de haber modificaciones:

- Nombre de la persona que lo modificó
- Fecha de modificación

	<b>Nomenclatura de Bases de Datos</b>	<b>Código:</b>	
		<b>Revisión:</b>	00
		<b>Fecha:</b>	13/02/2017

## **ESTÁNDAR DE NOMBRES EN OBJETOS DE BASES DE DATOS**

Se utilizarán las siguientes convenciones de nombres para los diferentes objetos de bases de datos

### **Bases de Datos**

Las Bases de datos deben representar el propósito de la misma y no a los usuarios, departamentos o gerencias. No deben ser necesariamente iguales a los nombres de las aplicaciones informáticas.

Las Bases de datos deben nombrarse

- En español
- Sin utilizar espacios en blanco ni underscore
- Pueden tener nombres compuestos
- Notación pascal

### **Esquemas**

Se recomienda el uso de esquemas dentro de la base de datos, por defecto al crear una base de datos se cuenta con el esquema DBO, sin embargo es deseable crear un esquema con otro nombre por motivos de seguridad aunque no es obligatorio crear otro.

En una base de datos de uso por múltiples aplicaciones se deben agrupar los objetos (tablas, funciones, procedimientos) en múltiples esquemas.

Los esquemas deben ser nombrados

- En español
- Sin utilizar espacios en blanco ni underscore
- Su longitud debe ser de 3 a 6 letras
- Una sola palabra
- Todo en minúsculas

<b>Código:</b>		<b>Nomenclatura de Bases de Datos</b>	
<b>Revisión:</b>	00		
<b>Fecha:</b>	13/02/2017		

## Tablas

Las tablas deben nombrarse

- En español
- Sin utilizar espacios en blanco ni underscore
- Si el nombre es compuesto solo la última palabra debe ir en plural
- Notación pascal
- Longitud máxima 30 caracteres

**NOTA** en algunos casos especiales en que una base de datos ya existente los objetos estén nombrados en inglés, las tablas para dicha BD deben nombrarse en inglés para conservar la uniformidad.

## Campos o columnas

Las columnas deben ser nombradas según los lineamientos a continuación:

- Los nombres deben ser simples, representativos e intuitivos.
- Los nombres de las columnas de una tabla deben estar expresados en singular.
- Campos que representen la misma entidad del mundo real, deben estar nombrados de la misma manera en todas las tablas de un esquema.
- Evitar en uso de los tipos nchar, nvarchar y ntext.
- El nombre de las columnas debe tener una longitud máxima 30 caracteres
- Todos los campos deben llevar una breve descripción de forma obligatoria.

## Clave primarias: Primary keys

En todos los casos posibles el nombre de la clave primaria debe ser el nombre de la tabla más las letras ID al final. Ejemplo PaisID.

	<b>Nomenclatura de Bases de Datos</b>	<b>Código:</b>	
		<b>Revisión:</b>	00
		<b>Fecha:</b>	13/02/2017

Dicha clave puede ser una clave artificial (identity, unique identifier) o una clave natural en caso de la clave natural ser de tipo varchar se preferirá colocar adicionalmente un identity que pasara a ser el PK de la tabla y a la clave natural agregar una restricción unique.

Se pueden usar claves combinadas pero se evitará su uso siempre que sea posible. Utilizarla solo en caso de que sea más práctico utilizar clave combinada que crear un campo adicional identity.

### Foreign keys

Para el nombramiento de FK se utilizará la notación que usa por defecto MS SQL Server

FK\_<TABLA\_QUE\_REFERENCIA\_<TABLA\_REFERENCIADA

### Indexes

La definición de índices tiene un impacto positivo en los tiempos de consulta de registro y uno negativo en los de inserción y actualización de los campos del índice.

Se usará la notación por defecto de MS SQL Server

[IDX\_]<TABLA>\_<CAMPO>[\_AUX]

### Procedimientos Almacenados

Los stored procedures deben ser nombrados según la siguiente nomenclatura:

- Usar notación Pascal
- Longitud máxima del nombre 40 caracteres
- El nombre debe estar compuesto por la entidad y la acción y opcionalmente de un complemento, ejemplos: DependientesEliminar, EmpleadoConsultaxID

<Entidad> <Acción> <Complemento>

- El código SQL dentro de un procedure debe ir en mayúsculas
- Devolver siempre las columnas necesarias, no use sentencias del tipo “SELECT \* FROM”, en su lugar coloque explícitamente los nombres de los campos
- Utilice cursores solo cuando es necesario.

<b>Código:</b>		<b>Nomenclatura de Bases de Datos</b>	
<b>Revisión:</b>	00		
<b>Fecha:</b>	13/02/2017		

- Utilice tablas temporales solamente cuando es estrictamente necesario
- Agregue comentarios en la lógica compleja.

### Funciones definidas por el usuario

Las funciones definidas por usuario deben seguir la siguiente nomenclatura:

- Usar notación Pascal
- Longitud máxima del nombre 30 caracteres
- El nombre de la función debe estar basado en el valor devuelto por la misma, compuesto opcionalmente por una acción o verbo, opcionalmente seguido de un sustantivo y complemento de ser necesario: Ejemplos: BuscarEmpleadoID, EstaActivaAgencia, TipoDeVuelo

< Acción/Verbo > < Sustantivo > <Complemento>

- El código SQL dentro de una función debe ir en mayúsculas
- Evite de ser posible el uso de tablas temporales
- No use cursores
- Agregue comentarios en la lógica compleja.

### Triggers

Los triggers deben ser nombrados según la siguiente nomenclatura:

- Usar notación Pascal
- Longitud máxima del nombre será de 50 caracteres
- El nombre del trigger debe indicar la tabla cuyo evento lo dispara, el tipo de trigger (AF para AFTER, IO para INSTEAD OF) y opcionalmente un complemento. Ejemplos: IataAFInsertar, EmpleadoAFAActualizarCargo

Tabla><AF/IO><Evento><Complemento>

- Documente en el encabezado si el trigger ejecuta operaciones DML en otras tablas.



	<b>Nomenclatura de Bases de Datos</b>	<b>Código:</b>	
		<b>Revisión:</b>	00
		<b>Fecha:</b>	13/02/2017

- El código SQL dentro de un trigger debe ir en mayúsculas
- Agregue comentarios en la lógica compleja.

### **Tipos de datos definidos por el usuario**

#### **Tipo de Dato Escalar**

- Aplican las mismas convenciones que para nombrar una columna

#### **Tipo de Dato Tabla**

- Aplican las mismas convenciones que para nombrar una tabla

<b>Código:</b>		<b>Nomenclatura de Bases de Datos</b>	
<b>Revisión:</b>	00		
<b>Fecha:</b>	13/02/2017		

## 7. DOCUMENTOS RELACIONADOS INTERNOS

Documentos Internos	Código

## 8. DOCUMENTOS RELACIONADOS EXTERNOS

Documentos Externos	Código
N/A	N/A

## 9. FORMATOS RELACIONADOS

Formato	Código
Solicitud de Documentación	FO-GOM-001/16
Carta de Respuesta de Solicitud	FO-GOM-002/16

## 10. MEDICION DE INDICADOR DE GESTIÓN

KPI	Fórmula	Frecuencia
N/A	N/A	N/A

## 11. MATRIZ DE RIESGO DEL PROCEDIMIENTO

Objetivo	Actividad	Riesgo del proceso	Criticidad	Documento de Referencia
N/A	N/A	N/A	N/A	N/A