

	SERVICIO NACIONAL DE CERTIFICACIÓN AMBIENTAL PARA LAS INVERSIONES SOSTENIBLES	Código	MANUAL N° 00003- 2021-SENACE/PE
		Versión	01
		Fecha	19.10.21
<h1>MANUAL</h1> <h2>ESTÁNDARES DE NOMBRES DE OBJETOS DE BASE DE DATOS PARA EL DESARROLLO DE SOFTWARE DEL SENACE</h2>			

1. OBJETO

Establecer lineamientos y estándares para la nomenclatura de los objetos de base de datos que almacenan información, permitiendo su rápida identificación, mejorando la legibilidad del código fuente y facilitando el mantenimiento de las bases de datos del Servicio Nacional de Certificación Ambiental para las Inversiones Sostenibles - Senace.

2. ALCANCE

Lo dispuesto en el presente manual es de obligatorio cumplimiento para los servidores de la Oficina de Tecnología de la Información, quienes deberán supervisar que los proveedores de tecnologías digitales cumplan lo dispuesto en el manual de estándares de nombres de objetos de base de datos para el desarrollo de software del Servicio Nacional de Certificación Ambiental para las Inversiones Sostenibles – Senace.

3. PROCESO VINCULADO

El presente manual está vinculado al macroproceso 11-Gestión de la Tecnología de la Información, dentro del cual se encuentran los procesos: 26 - Gestión de requerimientos de desarrollo de software y 27 - Desarrollo y control de calidad de software, en concordancia con el Manual de Procesos del Senace vigente.

4. RESPONSABILIDAD

La Oficina de Tecnologías de la Información es responsable de velar por el cumplimiento de lo dispuesto en el presente manual.

5. DEFINICIONES¹

- 5.1 **Ambiente o entorno de base de datos:** Es el conjunto de *hardware* y *software* donde se ejecutan las bases de datos utilizadas durante el desarrollo de los sistemas informáticos hasta su puesta en producción.
- 5.2 **Base de datos:** Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En las bases de datos, se puede almacenar información sobre personas, expedientes, estudios, mapas, entre otros tipos de datos.
- 5.3 **Base de datos de desarrollo:** Se refiere a la base de datos en la que los desarrolladores crean los objetos de base de datos y realizan las primeras pruebas. No contiene información real.
- 5.4 **Base de datos de calidad:** Se refiere a la base de datos en la que el personal designado como analista de calidad realiza pruebas del código fuente de los

¹ Definiciones elaboradas por la Oficina de Tecnología de la Información

objetos de base de datos, antes de que estos pasen a producción. Contiene información similar a la real.

- 5.5 **Base de datos de producción:** Es la base de datos donde se conectan los usuarios finales y sistemas informáticos en producción. Contiene información real de la entidad.
- 5.6 **Consulta de base de datos:** Son líneas de código escritas en lenguaje SQL para obtener información de una o más tablas de una base de datos. Es también conocida por el término en inglés “*Query*”.
- 5.7 **Database link (Dblink):** Un *dblink* es una conexión que se establece entre dos bases de datos. El objetivo de esta conexión es la obtención de datos o la realización de consultas sobre una base ajena a la que estamos conectados.
- 5.8 **Diccionario de datos:** Se refiere a la metadata (datos acerca de los datos). Es un conjunto de definiciones que contiene las características de los datos, tales como significado, relación con otros datos, origen, uso, formato, etc.
- 5.9 **Esquema (Schema):** Es una colección de estructuras lógicas de datos u objetos de base de datos relacionados. Agrupa los objetos de base de datos de un sistema informático específico.
- 5.10 **Integridad referencial:** Es un sistema de reglas que utilizan las bases de datos relacionales para asegurarse que los registros de tablas relacionadas son válidos y que no se borren o cambien datos relacionados de forma accidental produciendo errores de integridad.
- 5.11 **Lenguaje de consulta estructurado (SQL):** Denominación que proviene del inglés *Structured Query Language*. Es un lenguaje de programación diseñado para administrar y recuperar información de sistemas de gestión de bases de datos.
- 5.12 **Lenguaje de manipulación de datos (DML):** Término que proviene del inglés *Data Manipulation Language*. Es parte del lenguaje SQL y permite llevar a cabo las tareas de consulta o modificación de los datos contenidos en las bases de datos del sistema gestor de bases de datos.
- 5.13 **Modelo entidad relación (Modelo E/R):** Es una herramienta para el modelado de datos, la cual facilita la representación de entidades (objetos del mundo real) y sus relaciones (vínculos entre entidades) dentro de una base de datos. Se conforma por un diagrama entidad-relación, una lista de atributos y la descripción de otras relaciones que no pueden mostrarse en el diagrama.
- 5.14 **Objetos de base de datos:** Son estructuras lógicas que se almacenan dentro de la base de datos, cumplen determinadas funciones y son usadas por los sistemas informáticos.
- 5.15 **Usuario de base de datos:** Son usuarios que se crean dentro de la base de datos; son utilizados para que los sistemas informáticos se conecten a la base de datos y/o almacenen sus objetos de base de datos.

6. ABREVIATURAS

BD	:	Base de datos.
OTI	:	Oficina de Tecnologías de la Información.
Senace	:	Servicio Nacional de Certificación Ambiental para las Inversiones Sostenibles.

7. DISPOSICIONES O ASPECTOS GENERALES

- 7.1 La aplicación de estos estándares es de uso obligatorio en el desarrollo de nuevos sistemas de información y en el mantenimiento evolutivo de los mismos ya sea mediante recursos propios o a través de la contratación de proveedores de servicios.
- 7.2 La OTI es responsable de brindar a los proveedores de servicios de desarrollo de sistemas que entregan código fuente, los estándares de base de datos definidos para asegurar un eficiente mantenimiento evolutivo.
- 7.3 La OTI deberá revisar y validar la aplicación de este manual, exigiendo el modelo entidad relación y el diccionario de datos correspondiente.
- 7.4 Los estándares descritos en el presente manual deben aplicarse en la creación de nuevos objetos en las bases de datos de los ambientes de desarrollo, calidad, producción, y otros ambientes de base de datos utilizados durante el desarrollo de los sistemas informáticos.

8. DESCRIPCIÓN

8.1. USUARIOS DE BASE DE DATOS PARA LOS SISTEMAS INFORMÁTICOS

Para la clasificación de los sistemas de información dentro de la base de datos, se ha establecido que cada “sistema informático” a desarrollar contará con dos (02) usuarios de base de datos: uno de almacenamiento de sus objetos denominado “**USUARIO PROPIETARIO**” (conocido también como “**OWNER USER**” por su traducción al inglés) y el otro de acceso a la base de datos desde la aplicación denominado “**USUARIO DE ACCESO**”.

Con el “**USUARIO DE ACCESO**”, se realizarán las compilaciones de los sistemas informáticos y cambios de contraseña periódicos según las políticas de seguridad establecidas. Sobre este usuario se asignarán los privilegios sustentados y necesarios para la funcionalidad del aplicativo dentro de la base de datos, siguiendo lo establecido en lo referente a la seguridad de la información de la base de datos.

El estándar de nomenclatura para los usuarios de base de datos “**USUARIO PROPIETARIO**” y “**USUARIO DE ACCESO**” se define a continuación:

8.1.1. USUARIO PROPIETARIO

Usuario que almacenará los objetos de la aplicación.

Nomenclatura: IDO_<NombreAplicación(1)>

Ejemplo: IDO_VENTANILLAUNICA

Tipo de usuario	Nombre de aplicación(1)
IDO	VENTANILLA ÚNICA

Donde:

- (1) **Nombre Aplicación**, viene a ser el nombre abreviado de la aplicación, que debe de ser igual tanto para el “**USUARIO PROPIETARIO**” como para el “**USUARIO DE ACCESO**” y su tamaño dependerá de lo permitido por el gestor de la base de datos.

8.1.2. USUARIO DE ACCESO

Usuario que la aplicación utilizará para acceder a la base de datos.

Nomenclatura: IDU_<NombreAplicación(1)>

Ejemplo: IDU_VENTANILLAUNICA

Tipo de usuario	Nombre de aplicación(1)
IDU	VENTANILLA ÚNICA

Donde:

- (1) **Nombre de Aplicación**, viene a ser el nombre abreviado de la aplicación, que debe de ser igual tanto para el “**USUARIO PROPIETARIO**” como para el “**USUARIO DE ACCESO**” y su tamaño dependerá de lo permitido por el gestor de la base de datos.

8.2. ESTÁNDAR DE NOMENCLATURA DE OBJETOS DE BASE DE DATOS

Los objetos de base de datos son estructuras lógicas que se almacenan dentro de la base de datos y forman parte del código fuente de los sistemas informáticos que soportan los procesos de negocio, con el objetivo de mejorar la administración y mantenimiento de los sistemas se han definido los siguientes estándares:

Nomenclatura:

<IdSistema(1)><IdObjeto(2)>_<NombreObjeto(SegúnNomenclatura)(3)>

Donde:

- (1) **Identificador del sistema.** - Los sistemas informáticos serán identificados a través de un código de dos (02) caracteres, el cual formará parte de la

nomenclatura del objeto para identificar a qué sistema pertenece. A continuación, se indican algunos identificadores propuestos para sistemas existentes:

Tabla 1 - Identificador del sistema

Nombre del sistema	Identificador de sistema
TEMPUS	TP
SIGA	SG
CONTACTOS	CT
VENTANILLA ÚNICA	VU

(2) **Identificador de objeto.** - Corresponde al tipo de objeto (tabla, vista, procedimiento, paquete, secuencia, código java, disparadores, etc.) al que pertenece el objeto e información adicional del objeto según sea el caso, tal como se muestra en el siguiente cuadro:

Tabla 2 - Identificador de objeto

TIPO DE OBJETO	IDENTIFICADOR DE OBJETO
Tabla (Table)	T<TipoTabla>
Vista (View)	V<TipoVista>
Procedimiento almacenado (Stored procedure)	SP
Paquete (Package)	PKG
Función (Function)	FU
Disparador (Trigger)	G<TipoTrigger>
Restricción (Constraint)	<TipoConstraint>K
Índice (Index)	ID<TipoIndice>
Secuencia (Sequence)	SE
Enlace de BD (Database Link)	LK
Dimensión (Dimension)	DS
Directorio (Directory)	DI
Tipo (Type)	TY

TIPO DE OBJETO	IDENTIFICADOR DE OBJETO
Trabajo (Job)	J
Rol (Role)	R
Perfil (Profile)	P
Sinónimo (Synonym)	S
Código Java	C<TipoCódigoJava>

- (3) **Nombre de objeto.**- Es una palabra o conjunto de palabras que describen al objeto de acuerdo a su función, a los datos que almacenará, etc. Para nombres compuestos irá separado por el carácter subrayado ‘_’.

Ejemplos:

Tabla	: VUTM_PERSONA, VUTD_DIRECCIONES
Vista	: VUVM_PERSONA, VUVD_DIRECCIONES
Vista materializada	: VUVZ_PERSONA Procedimiento : VUSP_PERSONA, VUSP_DIRECCIONES
Función	: VUFU_PERSONA, VUFU_DIRECCIONES
Paquete	: VUPKG_PERSONA
Disparador	: VUGT_AF_UI_VUTM_PERSONA
Restricción	: PK_VUTM_PERSONA, CK_VUTM_PERSONA_01, FK_VUTM_PERSONA_01, UK_VUTM_PERSONA_01.
Índice	: IDX_VUTM_PERSONA, IDU_VUTM_PERSONA, IDB_VUTM_PERSONA, IDF_VUTM_PERSONA
Secuencia	: VUSE_PERSONA
Enlace de BD	: VULK_PPRI_CPRI
Directorio	: VUDI_TRABAJO
FuenteJava	: VUCS_PERSONA
Clase Java	: VUCC_PERSONA
Tipo	: VUTY_VUTM_PERSONA
Trabajo	: VUJJ_01_OTI_A_RMSESSION
Rol	: R_SEGURIDAD
Perfil	: P_APLICATIVO

Sinónimo : SP_VUTM_PERSONA

A continuación, se detalla la nomenclatura de objeto de base de datos según su tipo:

8.2.1. TABLA (TABLE)

Es la unidad básica de almacenamiento de una base de datos, que almacena los datos dentro de un arreglo de filas y columnas.

Se define por un conjunto de columnas (Ver “9.3.- Estándar de Columnas”), ocupa un espacio dentro de la base de datos y está identificado como tipo de objeto “TABLE”.

Con respecto a su nomenclatura, ha sido clasificada según el tipo de información que almacena, tal como se muestra en el siguiente cuadro:

Nomenclatura: <IdSistema (1)><IdObjeto (2)>_<NombreTabla(3)>

Id. Sistema(1)	Id. Objeto(2)	Nombre de tabla(3)
VU	TT	DIRECCIONES

Ejemplo: VUTT_DIRECCIONES

Donde:

(1) **Identificador de sistema:** Código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (Ver **Tabla 1 - Identificador del Sistema**).

(2) **Identificador de objeto,** está formado por<TipoObjeto><TipoTabla>.

<TipoObjeto>corresponde a la identificación del objeto tipo tabla (Ver **Tabla 2 - Identificador de Objeto**). Tiene la longitud de un (01) caracter.

Ejemplo:T<TipoTabla>

<TipoTabla> define el tipo de información que se almacena dentro de la misma. (Ver **Tabla 3 – Tipo de Tabla**). Tiene la longitud de un(01) caracter.

Tabla 3 – Tipo de tabla

Tipo contenido	Abreviatura (01 caracter)
Maestra	M
Movimiento	V
Cabecera	C
Detalle	D
Auxiliar	X
Referencia	R
Log	L
Histórica	H
Temporal	T
Tabla de Hechos (Fact Table – Inteligencia de Negocios)	F
Tabla de Dimensiones (Dimensions Table - Inteligencia de Negocios)	S
Tabla de Tránsito (Staging Table) - Inteligencia de Negocios)	G

- (3) **Nombre de tabla**, es una palabra o conjunto de palabras que describen a la tabla, tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.2.2. VISTA (VIEW)

Es la representación lógica de una o más tablas; y, en esencia, es una consulta o “**Query**” almacenado dentro de la base de datos y está definida con el nombre de “VIEW”.

Nomenclatura: <IdSistema (1)><IdObjeto (2)>_<NombreVista(3)>,

Ejemplo: VUVM_PERSONA

Id. Sistema(1)	Id. Objeto (2)	Nombre de vista(3)
VU	VM	PERSONA

Donde:

(1) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. **(Ver Tabla 1 - Identificador del Sistema Informático).**

(2) **Identificador de objeto**, corresponde a la identificación del objeto tipo vista. Está formado <TipoObjeto><TipoVista>.

<TipoObjeto> corresponde a la identificación del objeto tipo vista **(Ver Tabla 2 - Identificador de Objeto)**. Tiene una longitud de un (01) caracter.

Ejemplo: V<TipoVista>

<TipoVista> define el tipo de información que se almacena dentro de la misma. Tiene una longitud de un (01) caracter. **(Ver Tabla 4 - Tipo de Vista).**

Tabla 4 - Tipo de vista

Tipo de Vista	Abreviatura (1 caracter)
Maestra	M
Movimiento	V
Cabecera	C
Detalle	D
Auxiliar	X
Referencia	R
Log	L
Histórica	H
Temporal	T
Materializada	Z

(3) **Nombre de vista**, es una palabra o conjunto de palabras que describen a la vista, tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.2.3. PROCEDIMIENTO ALMACENADO (STORED PROCEDURE)

Un procedimiento almacenado es un programa formado por un conjunto de instrucciones SQL y estructuras de control que permiten dotar de cierta lógica al procedimiento. Está guardado físicamente en la base de datos en formato ejecutable como un tipo de objeto con el nombre de “STORED PROCEDURE”.

Nomenclatura:

<IdSistema(1)><IdObjeto(2)>_<NombreProcedimiento(3)>

Ejemplo: VUSP_PERSONA

Id. Sistema(1)	Id. Objeto(2)	Nombre de procedimiento(3)
VU	SP	PERSONA

Donde:

- (1) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. **(Ver Tabla 1 - Identificador del Sistema).**
- (2) **Identificador de objeto**, corresponde a la identificación del objeto tipo Procedimiento Almacenado **(Ver Tabla 2 - Identificador de Objeto)**. Tiene una longitud de dos (02) caracteres y está identificado por la abreviatura “SP”.
- (3) **Nombre de procedimiento**, es una palabra o conjunto de palabras que describen al procedimiento, tiene una longitud máxima según lo permitido por la versión del motor; para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.2.4. FUNCIÓN (FUNCTION)

Una función es una rutina almacenada que recibe unos parámetros escalares de entrada, posteriormente se procesa según el cuerpo definido de la función y por último retorna en un resultado de un tipo específico que permitirá cumplir un objetivo. Asimismo, forma parte de los procesos del sistema informático y está definida dentro de la base de datos con el nombre de “FUNCTION”.

Nomenclatura: <IdSistema(1)><IdObjeto(2)>_<NombreFuncion(3)>

Ejemplo: VUFU_PERSONA

Id. Sistema(1)	Id. Objeto (2)	Nombre de función (3)
VU	FU	PERSONA

Donde:

- (1) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. **(Ver Tabla 1 - Identificador del Sistema Informático)**.
- (2) **Identificador de objeto**, corresponde a la identificación del objeto tipo función **(Ver Tabla2 - Identificador de Objeto)**. Tiene una longitud de dos (02) caracteres y está identificado por la abreviatura "FU".
- (3) **Nombre de función**, es una palabra o conjunto de palabras que describen a la función, tiene una longitud máxima según lo permitido por la versión del motor; para nombres compuestos irá separado por el carácter de subrayado '_'.

8.2.5. PAQUETE (PACKAGE)

Un paquete es un objeto que agrupa y organiza varias funciones, tipos y/o procedimientos almacenados en una base de datos. Está definido dentro de la base de datos con el nombre de "PACKAGE".

Consta de dos (02) elementos:

1.Especificación o cabecera: contiene las declaraciones públicas de sus programas, tipos, constantes, variables, cursores, excepciones, las cuales son accesibles desde cualquier parte de la aplicación.

2.Cuerpo: contiene los detalles de implementación y declaraciones privadas, las cuales son accesibles solamente desde los objetos del paquete.

Nomenclatura: <IdSistema(1)><IdObjeto(2)>_<NombrePaquete(3)>

Ejemplo: VUPKG_PERSONA

Id. Sistema(1)	Id. Objeto (2)	Nombre de paquete(3)
VU	PKG	PERSONA

Donde:

- (1) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene

una longitud de dos (02) caracteres. (**Ver Tabla 1 -Identificador del Sistema**).

- (2) **Identificador de objeto**, corresponde a la identificación del objeto tipo paquete (**Ver Tabla 2 - Identificador de Objeto**). Tiene una longitud de tres (03) caracteres y está identificado por la abreviatura “**PKG**”.
- (3) **Nombre de paquete**, es una palabra o conjunto de palabras que describen al paquete, tiene una longitud máxima según lo permitido por la versión del motor; para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.2.6. DISPARADOR (TRIGGER)

Un disparador es un objeto construido por un bloque de código que se ejecuta automáticamente cuando ocurre un evento sobre una determinada tabla o vista. Los disparadores forman parte de la lógica de los procesos del sistema informático y están definidos dentro de la base de datos con el nombre de “TRIGGER”.

Nomenclatura:

<IdSistema(1)><IdObjeto(2)>_<Modalidad(3)>_<Acción(4)>_<Nombre Trigger(5)>

Ejemplo: VUGT_AF_UI_VUTM_PERSONA

Id. Sistema (1)	Id. Objeto (2)	Modalidad (3)	Evento (4)	Nombre de trigger (5)
VU	GT	AF	UI	VUTM_PERSONA

Donde:

- (1) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (**Ver Tabla 1 - Identificador del Sistema**).
- (2) **Identificador de objeto**, corresponde al conjunto de letras que identifican al objeto tipo trigger (**Ver Tabla 2- Identificador de Objeto**). Tiene una longitud de dos (02) caracteres y está conformado por la abreviatura “**G**”<TipoTrigger>, donde <TipoTrigger> puede tomar los valores indicados en la **Tabla 5 – Tipo de Trigger**

Tabla 5 – Tipo de trigger

Descripción	Abreviatura (1 caracter)
Disparador de tabla	T
Disparador de esquema	E
Disparador de base de datos	B

- (3) **Modalidad**, indica en qué momento se ejecuta el disparador, y sobre qué se ejecuta. Tiene una longitud de dos (02) caracteres:<TiempoTrigger><Efecto> (ver **Tabla 6– Tiempo de Disparador** y **Tabla 7 -Efecto**).

Tabla 6 – Tiempo de disparador

Descripción	Abreviatura (1 caracter)
Antes de ejecutar el evento	A
Después de ejecutado el evento	D

Tabla 7 – Efecto

Descripción	Abreviatura (1 caracter)
Fila	F
Sentencia	S

- (4) **Evento**, indica a qué evento responde el disparador. Tiene una longitud de hasta dos (02) caracteres (Ver **Tabla 8 – Eventos**).

Tabla 8 – Eventos

Descripción	Abreviatura (hasta 2 caracteres)
-------------	-------------------------------------

Actualización (Update)	U
Eliminación (Delete)	D
Inserción (Insert)	I
Conexión (Logon)	LI
Desconexión (Logoff)	LO

- (5) **Nombre del disparador**, es una palabra o conjunto de palabras que describen al disparador, y tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado ‘_’.

<IdSistema(1)><IdObjeto(2)>_<Nombreobjeto(3)>

(ver **Tabla 1- Identificador del Sistema** y **Tabla 2- Identificador de Objeto**).

Ejemplo: VUTM_PERSONA

Id. Sistema (1)	Id. Objeto (2)	Nombre del objeto (3)
VU	TM	PERSONA

8.2.7. RESTRICCIÓN (CONSTRAINT)

Es un tipo de objeto que tiene la particularidad de mantener la integridad de los datos almacenados dentro de la base de datos y se definen en el modelo E/R del sistema informático:

Tabla 9 – Tipo de restricción

Tipo de restricciones	Abreviatura (1 caracter)
Clave primaria (Primary Key)	P
Clave foránea (Foreign Key)	F
Clave única (Unique Key)	U
Verificación (Check)	C

8.2.7.1. CLAVE PRIMARIA (PRIMARY KEY)

La restricción de tipo clave o llave primaria hacen referencia a un campo o grupo de campos que identifican en forma única un registro dentro de una tabla. Ningún otro registro puede tener la misma clave primaria. La clave o llave primaria se utiliza para distinguir un registro con el fin de que se pueda tener acceso a él, organizarlo y manipularlo:

Nomenclatura:

<IdRestricción(1)>_<Sistema(2)><IdObjeto(3)>_<NombreObjeto(4)>

Ejemplo: PK_VUTM_PERSONA

Tipo restriccion(1)	Id. Sistema(2)	Id. Objeto(3)	Nombre del objeto(4)
PK	VU	TM	PERSONA

Donde:

(1) **El identificador de restricción** está formado por <TipoRestricción>K. Tiene dos (02) caracteres.

(Ver Tabla 9 -Tipo Restricción).

(2) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. *(Ver Tabla1 - Identificador del Sistema).*

(3) **Identificador de objeto**, está formado <TipoObjeto><TipoTablaTipoVista>

<TipoObjeto>corresponde a la identificación del objeto tipo tabla *(Ver Tabla 2 - Identificador de Objeto).* Tiene la longitud de un (01) caracter.

Ejemplo T<TipoTabla>

<TipoTablaTipoVista> define el tipo de información que se almacena dentro de la misma *(Ver Tabla 3 - Tipo de Tabla)* o *(ver Tabla 4 - Tipo de Vista).* Tiene la longitud de un (01) caracter.

(4) **Nombre de objeto**, viene a ser el nombre del objeto al cual se aplica la restricción, el cual puede ser una tabla, vista materializada, etc.

8.2.7.2. CLAVE FORÁNEA, CLAVE ÚNICA, Y RESTRICCIÓN DE VERIFICACIÓN

- **Clave foránea (FOREIGN KEY)**

La restricción de tipo clave foránea (o FOREIGN KEY por su nombre en inglés) identifica una columna o grupo de columnas en una tabla (tabla hija) que se refiere a una columna o grupo de columnas en otra tabla (tabla padre o referenciada). Las columnas en la tabla hija deben ser la clave primaria u otra clave candidata en la tabla referenciada. Permite mantener la integridad referencial de información almacenada en otra tabla.

- **Clave única (UNIQUE KEY)**

La restricción de tipo clave única (o UNIQUE KEY por su nombre en inglés) sirve para indicar que una columna tendrá valores únicos para cada fila de la tabla. Especifica que dos registros no puedan tener el mismo valor en un campo. Se pueden aplicar varias restricciones de este tipo a una misma tabla, y pueden aplicarse a uno o varios campos que no sean clave primaria.

- **Restricción de verificación (CHECK)**

La restricción de tipo verificación se utiliza para limitar el rango de valores que puede tener una columna.

Nomenclatura:

<IdRestricción(1)>_<Sistema(2)><IdObjeto(3)>_<NombreObjeto(4)>_<secuencia(5)>

Ejemplo: FK_VUTM_PERSONA_01
 UK_VUTM_PERSONA_01
 CK_VUTM_PERSONA_01

TipoConstraint (1)	Id. Sistema (2)	Id. Objeto (3)	Nombre del objeto (4)	Secuencia (5)
FK	VU	TM	PERSONA	01
UK	VU	TM	PERSONA	01
CK	VU	TM	PERSONA	01

Donde:

(1) **El identificador de restricción**, está formado por <TipoRestricción>K. Tiene dos (02) caracteres.

El <TipoRestricción> (**Ver Tabla 9 - Tipo Restricción**).

Esta es una copia auténtica imprimible de un documento electrónico archivado en el Senace, aplicando lo dispuesto por el Art. 25 de D.S. 070-2013-PCM y la Tercera Disposición Complementaria Final del D.S. 026- 2016-PCM. Su autenticidad e integridad pueden ser contrastadas a través de la siguiente dirección web: "<https://www.senace.gob.pe/verificacion> ingresando el código de verificación que aparece en la parte superior izquierda de este documento.

- (2) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (**Ver Tabla1 - Identificador del Sistema**).
- (3) **Identificador de objeto**, está formado <TipoObjeto><TipoTablaTipoVista>
 <TipoObjeto>corresponde a la identificación del objeto tipo tabla (**Ver Tabla 2 - Identificador de Objeto**). Tiene una longitud de un (01) caracter. **Ejemplo:**T<TipoTabla>
 <TipoTablaTipoVista> define el tipo de información que se almacena dentro de la misma (**Ver Tabla 3 - Tipo de Tabla**) o (**ver Tabla 4 - Tipo de Vista**). Tiene la longitud de un (01) caracter.
- (4) **Nombre de objeto**, viene a ser el nombre del objeto al cual se le aplica la restricción, el cual puede ser una tabla, vista materializada, etc.
- (5) **Secuencia**, está formada por dos (02) caracteres y corresponde a la numeración de la restricción del objeto.

8.2.8. INDICE (INDEX)

Es un tipo de objeto que tiene la particularidad de mejorar el tiempo de respuesta de las sentencias de manipulación de datos o DML (del inglés “Data Manipulation Language”). Almacena contenidos dentro de la base de datos, por lo que se debe considerar un espacio físico para este tipo de objeto. Este tipo de objeto está definido en la base de datos con el nombre “INDEX”.

Tabla 10 – Tipo índice

Tipo índice	Abreviatura (1 caracter)
Árbol balanceado (B-Tree)	X
Mapa de bits (Bitmap)	B
Único (Unique)	U
Basado en función	F

Nomenclatura:

ID<TipoIndice(1)>_<Sistema(2)><IdObjeto(3)>_<NombreObjeto(4)>

Ejemplo: IDX_VUTM_PERSONA_01

id. índice(1)	Id. Sistema (2)	Id. Objeto (3)	Nombre del objeto (4)
IDX	VU	TM	PERSONA

Donde:

(1) **El identificador de índice**, está formado por ID<TipoÍndice>. Tiene tres (03) caracteres.

<TipoÍndice> (**Ver Tabla 10 - Tipo Índice**).

(2) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (**Ver Tabla 1- Identificador del Sistema**).

(3) **Identificador de objeto**, está formado <TipoObjeto><TipoTablaTipoVista>

<TipoObjeto>corresponde a la identificación del objeto tipo tabla (**Ver Tabla 2- Identificador de Objeto**). Tiene una longitud de un (01) carácter. **Ejemplo:**T<TipoTabla>

<TipoTablaTipoVista> define el tipo de información que se almacena dentro de la misma (**Ver Tabla 3 - Tipo de Tabla**) o (**ver Tabla 4 - Tipo de Vista**). Tiene una longitud de un (01) carácter.

(4) **Nombre de objeto**, es una palabra o conjunto de palabras que describen el índice. Hace referencia al objeto que hará uso del índice, y puede ser el nombre de una tabla, de una vista materializada, etc.

8.2.9. SECUENCIA (SEQUENCE)

Una secuencia es un tipo de objeto que permite generar valores enteros secuenciales únicos y asignárselos a campos numéricos de una tabla. Se utilizan generalmente para las claves primarias de las tablas, garantizando que sus valores no se repitan. Está definida dentro de la base de datos con el nombre de "SEQUENCE".

Nomenclatura: <IdSistema(1)><IdObjeto(2)>_<NombreSecuencia(3)>

Ejemplo: VUSE_PERSONA

Id. Sistema (1)	Id. Objeto (2)	Nombre de secuencia (3)
VU	SE	PERSONA

Donde:

- (1) **Identificador de sistema**, código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (**Ver Tabla 1 - Identificador del Sistema**).
- (2) **Identificador de objeto**, corresponde a la identificación del objeto secuencia (**Ver Tabla2 - Identificador de Objeto**) y está indicado por la abreviatura “SE”.
- (3) **Nombre de secuencia**, es una palabra o conjunto de palabras que describen la secuencia, tiene una longitud máxima según lo permitido por la versión del motor; para nombres compuestos irá separado por el carácter de subrayado ‘_’. Hace referencia al nombre de la tabla que hará uso de la secuencia.

8.2.10. ENLACE DE BASE DE DATOS (DATABASE LINK)

Un enlace de base de datos es una conexión que se establece entre dos bases de datos (BD origen y BD destino). El objetivo de esta conexión es que la BD origen pueda obtener o consultar datos de una BD destino. Un enlace de base de datos está definido dentro de la base de datos origen con el nombre de “DATABASE LINK”.

El enlace de base de datos es creado dentro del esquema del usuario propietario (Owner user) de la aplicación y para conectarse remotamente deberá de crearse en la base de datos destino un Usuario de Acceso con el mismo nombre.

A continuación, como ejemplo se muestra una tabla con los nombres de las bases de datos:

Tabla 11 – Nombres de base de datos

Nombre de BD	Entorno BD	Motor de BD
ORCL	Producción	Oracle 11G
COMUN	Producción	Oracle 11G
COREDB	Producción	Oracle 11G
VUCA	Producción	Oracle 11G

Nomenclatura:

<IdSistema(1)><IdObjeto(2)>_<BDOrigen(3)>_<BDDestino(4)>

Esta es una copia auténtica imprimible de un documento electrónico archivado en el Senace, aplicando lo dispuesto por el Art. 25 de D.S. 070-2013-PCM y la Tercera Disposición Complementaria Final del D.S. 026- 2016-PCM. Su autenticidad e integridad pueden ser contrastadas a través de la siguiente dirección web: “<https://www.senace.gob.pe/verificacion> ingresando el código de verificación que aparece en la parte superior izquierda de este documento.

Ejemplo: VULK_ ORCL _VUCA

Id. Sistema (1)	Id. Objeto (2)	Nombre BD origen (3)	Nombre BD destino (4)
VU	LK	ORCL	VUCA

Donde:

- (1) **Identificador de sistema**, es el código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. **(Ver Tabla 1-Identificador del Sistema).**
- (2) **Identificador de Objeto**, corresponde a la identificación del objeto enlace de BD (DatabaseLink) **(Ver Tabla 2 - Identificador de Objeto)** y está indicado por la abreviatura “LK”.
- (3) **NombreBDOrigen**, es el nombre de la base de datos origen. **(Ver Tabla 11 – Nombres de Base Datos).**
- (4) **NombreBDDestino**, es el nombre de la base de datos destino. **(Ver Tabla 11 – Nombres de Base Datos).**

8.2.11. DIRECTORIO (DIRECTORY)

Un directorio es un objeto de base de datos que apunta a un directorio del sistema operativo en la máquina del servidor de bases de datos para leer y escribir archivos. Está definido dentro de la base de datos con el nombre de “DIRECTORY”.

Nomenclatura: <IdSistema(1)><IdObjeto(2)>_<NombreDirectorio(3)>

Ejemplo: VUDI_TRABAJO

Id. Sistema (1)	Id. Objeto (2)	Nombre directorio (3)
VU	DI	TRABAJO

Donde:

- (1) **Identificador de sistema**, es el código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. **(Ver Tabla 1 - Identificador del Sistema).**

- (2) **Identificador de objeto**, corresponde a la identificación del objeto directorio (*Ver Tabla2 - Identificador de Objeto*) y está indicado por la abreviatura “DI”.
- (3) **Nombre directorio**, es la palabra o conjuntos de palabras que describen al directorio de acuerdo con su función dentro del sistema informático. Tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.2.12. CÓDIGO JAVA

Java es un lenguaje de programación que puede ser entendido y utilizado por la base de datos. Existen dos tipos de código JAVA como objetos de base de datos: fuente java (source java) y clase java (java class):

Tabla 12 – Tipo de código java

Tipo de código Java	Abreviatura (2 caracteres)
Fuente Java (Java Source)	CS
Clase Java (Java Class)	CC

Nomenclatura:

<Sistema(1)><IdObjeto(2)>_<NombreObjeto(3)>

Ejemplos:

VUCS_UTIL

VUCC_FUNCIONES

Tener en cuenta que el nombre completo del objeto de tipo Java deberá crearse en mayúscula.

Id. Sistema (1)	Id. Objeto (2)	Nombre del objeto (3)
VU	CS	UTIL
VU	CC	FUNCIONES

Donde:

- (1) **Identificador de sistema**, es el código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (**Ver Tabla 1 - Identificador del Sistema**).
- (2) **Identificador de objeto**, corresponde a la identificación del objeto tipo Java. Tiene una longitud de dos (02) caracteres, con la siguiente nomenclatura: C<TipoCodigoJava> (**Ver Tabla 12 - Tipo de Código Java**).
- (3) **Nombre objeto**, es una palabra o conjunto de palabras que describen al objeto dentro del sistema informático, tiene una longitud máxima según lo permitido por la versión del motor; para nombres compuestos irá separado por el carácter de subrayado '_'.

8.2.13. TIPO (TYPE)

Es un tipo de objeto que describe un tipo de dato definido por el usuario, el cual encapsula una estructura de datos junto con las funciones y procedimientos, que hace posible modelar entidades del mundo real.

Nomenclatura:<Sistema(1)><IdObjeto(2)>_<NombreObjeto(3)>

Ejemplo: VUTY_PERSONA

Id. Sistema (1)	Id. Objeto (2)	Nombre del objeto (3)
VU	TY	PERSONA

Donde:

- (1) **Identificador de sistema**, es el código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (**Ver Tabla 1 - Identificador del Sistema**).
- (2) **Identificador de objeto**, corresponde a la identificación del objeto tipo (Type). Tiene una longitud de dos (02) caracteres (**Ver Tabla 2 - Identificador de Objeto**).
- (3) **Nombre objeto**, es la palabra o conjunto de palabras que describen al objeto de base de datos dentro del sistema informático. Tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado '_'.

8.2.14. TRABAJO (JOB)

Es aquel proceso, programa o archivo que puede ejecutarse en el tiempo o fecha que se desee.

Nomenclatura:

<Sistema(1)><IdObjeto(2)><TipoTrabajo(3)>_<Sec(4)>_<AreaSolicitante(5)>_<Accion(6)>_<NombreObjeto(7)>

Ejemplo: VUJJ_01_OTI_A_RMSESSION

Id. Sistema (1)	Id. Objeto (2)	Tipo trabajo (3)	Sec (4)	Área solicitante (5)	Acción (6)	Nombre del objeto (7)
VU	J	J	01	OTI	A	RMSESSION

Donde:

- (1) **Identificador de sistema**, es el código que identifica al sistema informático al cual pertenecerá el objeto de base de datos. Tiene una longitud de dos (02) caracteres. (Ver **Tabla 1 - Identificador del Sistema**).
- (2) **Identificador de objeto**, corresponde a la identificación del objeto trabajo (Job). Tiene la longitud de un (01) carácter. (Ver **Tabla 2 - Identificador de Objeto**).
- (3) **Tipo trabajo**, corresponde a la identificación del tipo de trabajo (Job). Tiene la longitud de un (01) carácter. (Ver **Tabla 13 -Tipo de Trabajo**).

Tabla 13 – Tipo de trabajo

Tipo de Trabajo	Abreviatura (1 carácter)
PROGRAMS	P
SCHEDULES	S
JOBS	J
JOB_CLASS	C
WINDOW	W
WINDOW_GROUP	G

- (4) **Secuencia**, es una numeración secuencial de dos (02) dígitos.
- (5) **Área solicitante**, corresponde a la identificación del área que solicita el trabajo. En la **Tabla 14- Área solicitante**, se muestran abreviaturas para algunas áreas solicitantes.

Tabla14 – Área solicitante

Área Solicitante	Abreviatura (Máx.3caracteres)
Consejo Directivo	CD
Consejo Técnico Consultivo	CTC
Presidencia Ejecutiva	PE
Gerencia General	GG
Dirección de Evaluación Ambiental para Proyectos de Recursos Naturales y Productivos	DEAR
Dirección de Evaluación Ambiental para Proyectos de Infraestructura	DEIN
Dirección de la Dirección de Gestión Estratégica en Evaluación Ambiental	DGE
Subdirección de Proyección Estratégica y Normatividad	NOR
Subdirección de Registros Ambientales	REG
Subdirección de Seguimiento y Articulación	SEA
Oficina de Administración	OA
Unidad de Contabilidad y Control Previo	CCP
Unidad de Logística	LOG
Unidad de Recursos Humanos	RH
Unidad de Tesorería.	TES

Área Solicitante	Abreviatura (Máx.3caracteres)
Oficina de Asesoría Jurídica	OAJ
Oficina de Atención a la Ciudadanía y Gestión Documental	OAC
Órgano de Control Institucional	OCI
Oficina de Planeamiento y Presupuesto	OPP
Unidad de Planeamiento, Desarrollo Institucional y Cooperación Técnica	PLA
Unidad de Programación y Presupuesto	PRE
Oficina de Tecnologías de la Información	OTI
Unidad de Comunicaciones e Imagen Institucional	COM

- (6) **Acción del trabajo**, corresponde a la identificación de la acción que va realizar el trabajo (*Ver Tabla 15 - Acción del Trabajo*).

Tabla 15 – Acción del trabajo

Acción del trabajo	Abreviatura (1 carácter)
Alerta	A
Proceso	P

- (7) **Nombre objeto**, es una palabra o conjunto de palabras que describen al objeto de base de datos. Tiene una longitud máxima según lo permitido por la versión del motor; para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.2.15. ROL (ROLE)

Un rol es la agrupación de privilegios y/u otros roles; se utiliza para crear diferentes niveles de acceso a la base de datos. Este tipo de objeto está definido dentro de la base de datos con el nombre de “ROLE”.

Nomenclatura: R_<Tipo de Rol(1)>

Ejemplo: R_SEGURIDAD

Donde:

(1) **Tipo de rol**, es la descripción del rol:

Tabla 16 – Tipo de Rol

Tipo Rol(1)
Seguridad
Analista_EVA
DBA_Senior
DBA_Junior

8.2.16. PERFIL (PROFILE)

Un perfil de usuario es un conjunto de límites de recursos a los que se le asigna un nombre. Los perfiles pueden ser asignados a diferentes usuarios para controlar y limitar el uso de recursos del sistema. Este tipo de objeto está definido dentro de la base de datos con el nombre de "PROFILE"

Nomenclatura: P_<TipoPerfil(1)>

Ejemplo: P_SEGURIDAD

Donde:

(1) **Tipo perfil**, es la descripción del perfil:

Tabla 17 – Tipo de Perfil

Tipo Perfil (1)
Seguridad
Analista_EVA
DBA_Senior
DBA_Junior

8.2.17. SINÓNIMO (SYNONYM)

Un sinónimo es un objeto de base de datos que permite darle un nombre alternativo a otros objetos. Está identificado en la base de datos con el nombre "SYNONYM".

Nomenclatura: S<TipoSinónimo(1)>_<Objeto(2)>

Ejemplo: SP_VUTM_PERSONA, SR_VUTM_PERSONA

Donde:

(1) **Tipo sinónimo**, define el ámbito del sinónimo.

Tabla 18 – Tipo de Sinónimo

Tipo sinónimo	Abreviatura (1 carácter)
Público	P
Privado	R

(2) **Objeto**, es el nombre del cualquier objeto de la base de datos que puedan tener un sinónimo. Estos pueden ser: tablas, vistas, secuencias, procedimientos, funciones, paquetes, vistas materializadas, clase java, tipos, y otros..

8.3. ESTÁNDAR DE COLUMNAS

Es el patrón de nombres de columnas de las tablas en la base de datos.

Nomenclatura: <Prefijo (1)>_<Nombre (2)>. Tener en cuenta que el nombre completo deberá de estar en mayúscula.

Ejemplo: CO_ADMINISTRADO, FE_INGRESO_EXPEDIENTE

Prefijo (1)	Nombre (2)
CO	ADMINISTRADO
FE	INGRESO_EXPEDIENTE

Donde:

- (1) **Prefijo**, define el contenido de la columna. No confundir con el tipo de dato de la columna. Tiene una longitud de dos (02) caracteres. **(Ver Tabla 19- Prefijo de Columna).**

Tabla 19 – Prefijo de Columna

Contenido	Prefijo	Tipo Dato
Código	CO	Char/Numeric
Descripción	DE	Varchar2
Número	NU	Numeric
Secuencia	SE	Numeric
Indicador	IN	Char
Tipo	TI	Char
Estado	ES	Char
Fecha/Hora	FE	Date
Identificador (puede ser el identificador de sesión, carga, etc.)	ID	Char
Monto	MO	Decimal
Porcentaje	PC	Decimal
Cantidad	CA	Numeric / Decimal/ Integer
Nombre	NO	Varchar2
Apellido	AP	Varchar2
Medida	ME	Numeric / Decimal
Usuario	US	Varchar2/Numérico
Dirección	DI	Varchar2

Contenido	Prefijo	Tipo Dato
Dirección IP	IP	Varchar2/ Char
Contraseña	PW	Varchar2

- (2) **Nombre**, es la palabra o conjunto de palabras que describen la columna, tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado ‘_’.

8.4. VARIABLES Y PARÁMETROS DE PROCEDIMIENTOS ALMACENADOS

Es el patrón de nombres de variables y parámetros de procedimientos almacenados en la base de datos.

Tabla 20 – Abreviatura de tipo de dato para variables y parámetros

Tipo Contenido	Abreviatura
Char	c
Date	d
Float	f
Long	l
Number	n
Varchar2	v
Cursor	cr
Bfile	bf
Blob	bl
Clob	cl
Nchar	nc
NClob	ncl
Nvarchar2	nv

Tipo Contenido	Abreviatura
Ordimage	oi
Raw	ra
Rowid	ri
Long raw	lr
utl_smtp.connection	smtpc

8.5. PARÁMETRO

El parámetro es el dato que se considera como imprescindible y orientativo para lograr evaluar o valorar una determinada situación. Son los datos que se usan para intercambiar datos entre funciones y procedimientos almacenados.

Los parámetros pueden ser de entrada y salida y su declaración estaría definida de la siguiente manera:

Nomenclatura: p_<TipoDato(1)><NombreParametro(2)>

Ejemplo:

```
PROCEDURE VUSP_PERSONA
    p_crQuery IN OUT CUR_REF_MAT_ID,
    p_nCond NUMBER,
    p_vTI_Pago VARCHAR2)
```

Donde:

- (1) **Tipo dato**, define el tipo de dato que almacenará el parámetro (ver **Tabla 20 Abreviatura de tipo de dato para variables y parámetros**) y deberá ser registrado en **minúscula**.
- (2) **Nombre parámetro**, es una palabra o conjunto de palabras que describen al parámetro, donde mínimo la primera letra irá en mayúscula. Tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado '_'.

8.6. VARIABLE

La variable es una palabra que representa aquello que varía o que está sujeto a algún tipo de cambio. Las variables son espacios reservados en la memoria, como su nombre indica, pueden cambiar de contenido a lo largo de la ejecución de un programa. Una variable corresponde a un área reservada en la memoria principal del ordenador.

Su declaración estaría definida de la siguiente manera:

Nomenclatura: v_<TipoDato(1)><NombreVariable(2)>

Ejemplo:

```
v_nReturn    NUMBER;  
v_cEsTram_Act CHAR (1);  
v_vTi_Captu  VARCHAR2(2);
```

Donde:

- (1) **Tipo dato**, define el tipo de dato que almacenará la variable (**ver Tabla20 - Abreviatura de tipo de dato para variables y parámetros**) y deberá ser registrado en **minúscula**.
- (2) **Nombre variable**, es la palabra o conjunto de palabras que describen a la variable, donde mínimo la primera letra irá en mayúscula. Tiene una longitud máxima según lo permitido por la versión del motor. Para nombres compuestos irá separado por el carácter de subrayado '_'.