

LOGALI

TEORÍA

Tablas de base de datos

SAP ABAP Programación





Contenido

1. Tabla de Base de Datos	3
1.1. Creación de tablas transparentes	4
1.2. Campos de tipo importe y cantidad.....	9
2. Instrucciones SQL de ABAP/4.....	11
1.1. INSERT.....	11
1.2. SELECT	12



1. Tabla de Base de Datos

Las tablas de base de datos son objetos basados en estructuras de datos, capaces de almacenar gran cantidad de información en múltiples líneas de registros.

Existen los siguientes tipos de tablas:

- **Tabla Transparente** - Tabla habitual de base de datos relacional (SQL).
- **Tabla Cluster** – Tabla que guarda registros de varias tablas SAP con la misma clave cluster, en el mismo cluster físico de la base de datos.
- **Tabla Pool** - Tabla donde varias tablas SAP se guardan en una única tabla física de Base de Datos. Con las tablas Pool se mejora el acceso a los registros.

Las tablas más comunes son las tablas transparentes, que es el nombre que les da SAP a las tablas de Base de Datos relacionales. De hecho, los dos términos más comunes utilizados en el día a día de un Consultor ABAP son “tablas transparentes” y “tablas internas”; entendemos que las primeras son las tablas del Diccionario de Datos, y las segundas son espacios de memoria que se utilizan dentro de un programa, cuya existencia es únicamente en tiempo de ejecución del programa que tiene declarada la tabla interna.

Una tabla está formada por columnas (campos) y filas (entradas). Un campo tiene un nombre unívoco y atributos; por ejemplo, un campo clave. La clave primaria es una combinación de columnas o campos únicos que identifica los datos de un registro.

Para indicar el tipo de datos de las columnas utilizamos los elementos de datos que utilizan dominios.

Las maneras en que los programas ABAP pueden utilizar una tabla transparente son:



- Mediante sentencias OPEN SQL (o SQL nativo).
- Como tipo de estructura al definir variables (o tipos de datos más complejos).

Seguiremos con un ejemplo práctico y crearemos una tabla de base de datos (transparente) utilizando el siguiente modelo de datos:

Campo	Elemento de datos	Clave
MANDT	MANDT	X
FACTURA	VBELN_VF	X
CLS_FACT	FKART	
TP_FACT	FKTYP	
TP_DOC	VBTYP	
EJERCICIO	GJAHR	
PER_CNTB	POPER	

1.1. Creación de tablas transparentes

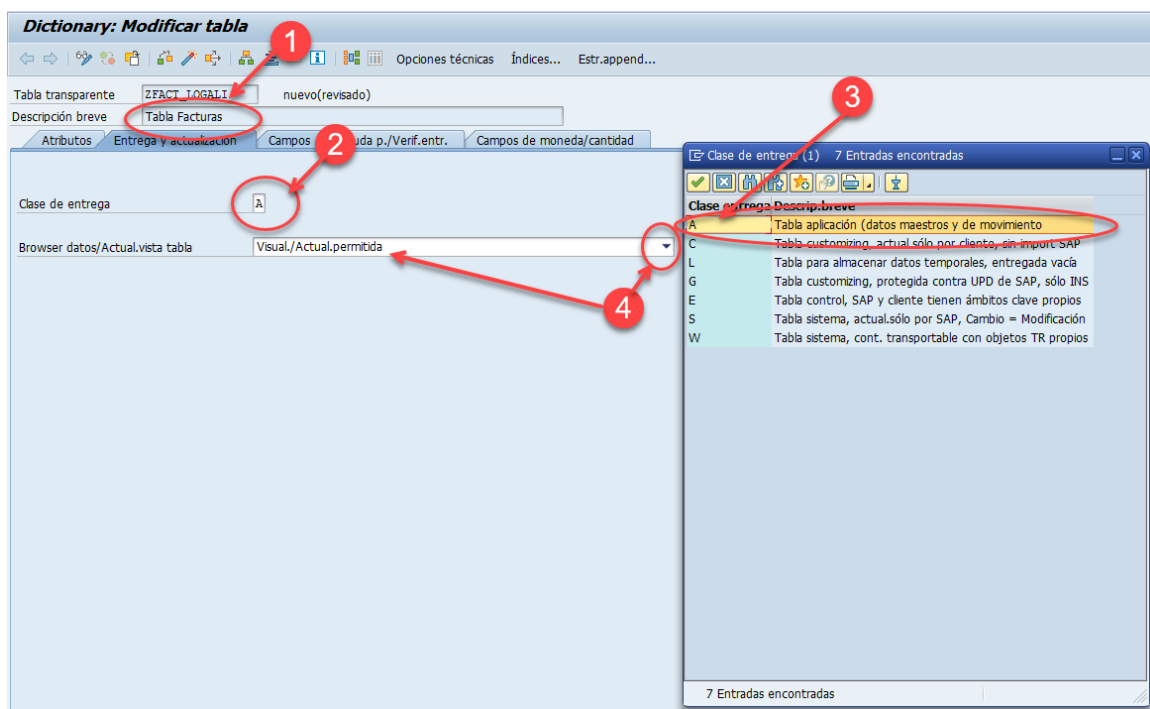
Pasos para crear la tabla:

- Accedemos a la transacción del diccionario de datos SE11 y añadimos el nombre de la tabla en campo **Tabla base datos**.

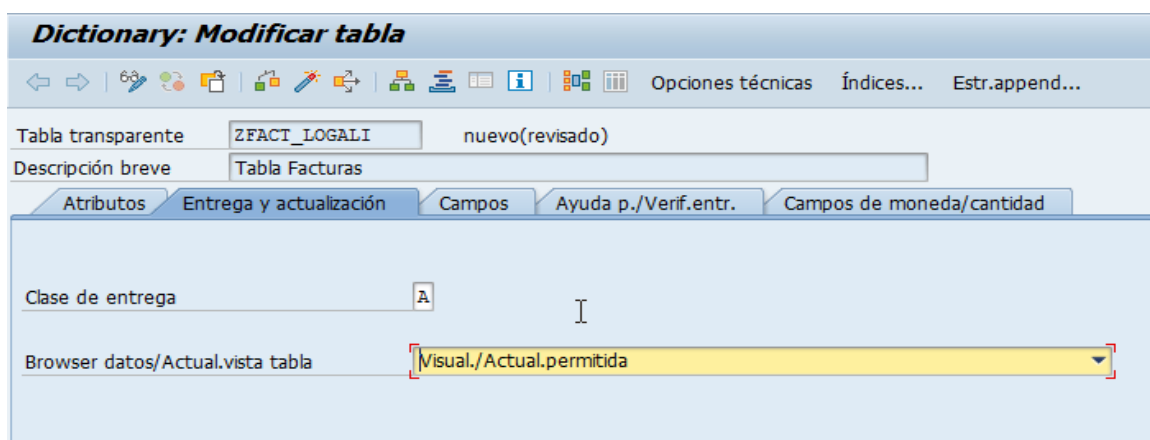


- B. Completamos el campo **Descripción** con un texto (en el ejemplo se utiliza *Tabla Facturas*) y en la pestaña **Entrega y actualización** seleccionamos la Clase de entrega **A** y el tipo de actualización **Visualización/Actualización permitida**.

La clase de entrega se selecciona en función de los tipos de datos que va a contener la tabla. En este caso se crea una tabla que va a contener información y se selecciona la primera opción **Tabla aplicación A**, opción que permite tener datos maestros y datos de movimientos.



El resultado quedaría así:





C. En la pestaña **Campos** añadimos los campos de la tabla.

Dictionary: Modificar tabla

Tabla transparente: ZFACT_LOGALI nuevo(revisado)

Descripción breve: Tabla Facturas

Atributos Entrega y actualización Campos Ayuda p./Verif.entr. Campos de moneda/cantidad

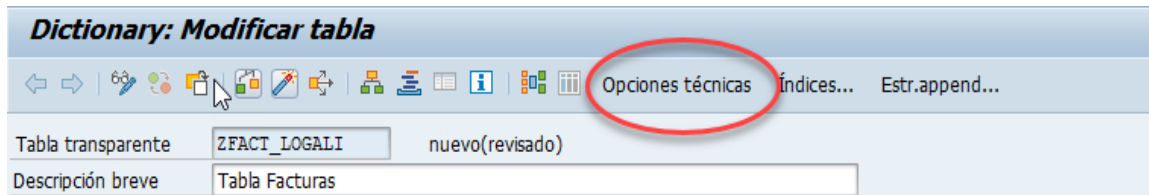
Campo	Clv	Val...	Elem.datos	Tipo de d...	Long.	Deci...	Descripción breve
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Mandante
FACTURA	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	VBELN_VF	CHAR	10	0	Factura
CLS_FACT	<input type="checkbox"/>	<input type="checkbox"/>	FKART	CHAR	4	0	Clase de factura
TP_FACT	<input type="checkbox"/>	<input type="checkbox"/>	FKTYP	CHAR	1	0	Tipo de factura
TP_DOC	<input type="checkbox"/>	<input type="checkbox"/>	VBTP	CHAR	1	0	Tipo de documento comercial
EJERCICIO	<input type="checkbox"/>	<input type="checkbox"/>	GJAHR	NUMC	4	0	Ejercicio
PER_CNTB	<input type="checkbox"/>	<input type="checkbox"/>	POPER	NUMC	3	0	Período contable
	<input type="checkbox"/>	<input type="checkbox"/>					
	<input type="checkbox"/>	<input type="checkbox"/>					
	<input type="checkbox"/>	<input type="checkbox"/>					
	<input type="checkbox"/>	<input type="checkbox"/>					

D. Antes de activar la tabla en base de datos debemos configurar las opciones técnicas. Las opciones técnicas optimizan los requisitos de almacenamiento y el comportamiento de acceso de las tablas de base de datos. Cuando se activa la tabla en el Diccionario de datos ABAP se crea la tabla en el sistema de base de datos. Las opciones técnicas definen la forma en que se maneja la tabla al ser creada en el sistema de base de datos. Las opciones para la clase de datos y la categoría de tamaño determinan el área de almacenamiento a seleccionar (**TABLESPACE**), así como el tamaño de tabla previsto. Las opciones de la memoria intermedia definen cómo se debe guardar la tabla en la memoria intermedia. Por ejemplo, si sabemos que los datos de las tablas son datos que son utilizados muchas veces, podemos indicarle al sistema que grabe los datos en la memoria intermedia. De este modo por cada acceso de lectura a la tabla cuando hacemos un **SELECT** la respuesta de la lectura es muy rápida, el sistema no tiene que acceder a la base de datos para obtener el registro, sino que directamente lo devuelve de la memoria intermedia. El gestor de grabación en memoria intermedia es muy complejo guardando referencias de los registros más utilizados y establece sincronización con los datos que hay en base de datos. También podemos definir si se deben registrar en log las modificaciones en las entradas de tabla. De esta forma, cada vez que se modifica un dato o un

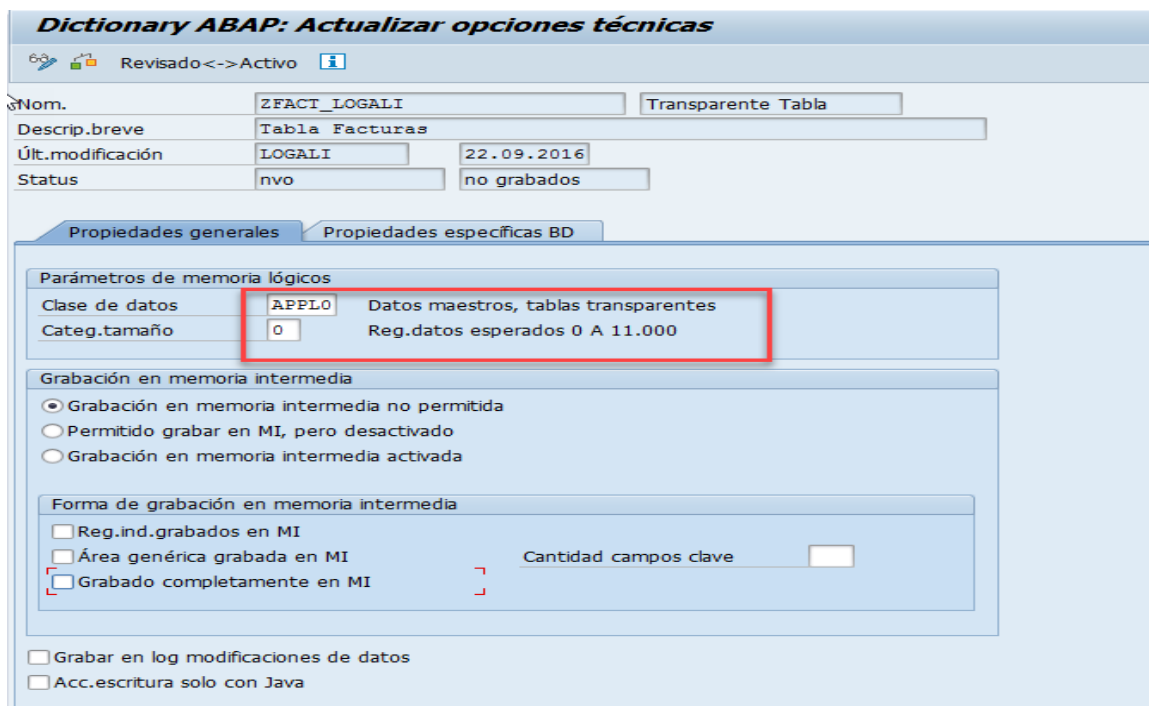


registro de la tabla se grabará un log que permitirá visualizar el historial de las modificaciones.

Para configurar las opciones técnicas utilizamos la opción de la barra de herramientas, grabando las modificaciones y asignando el paquete de desarrollo y la orden de transporte.



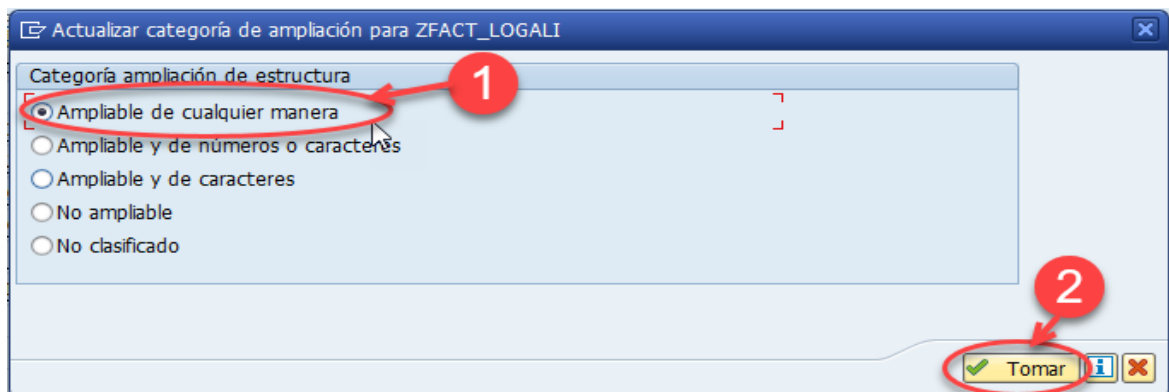
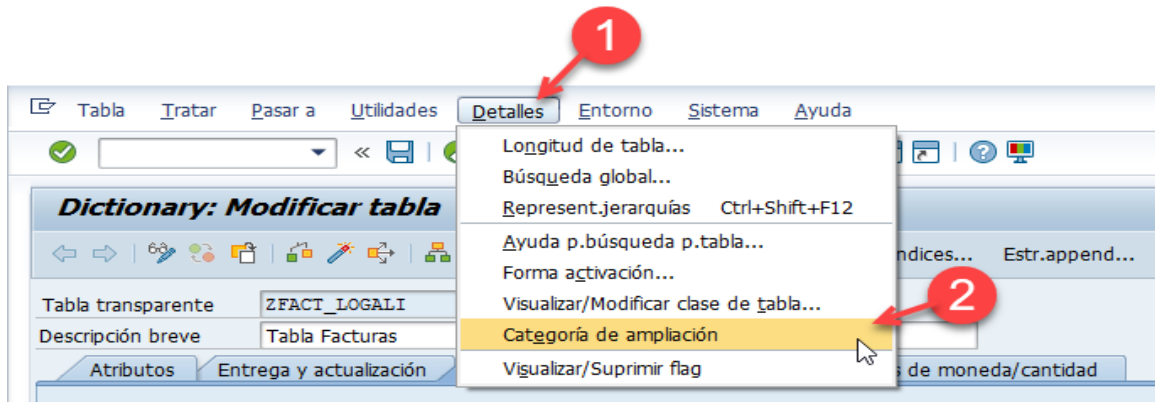
Como opciones técnicas seleccionamos la Clase de datos **APPL0 - Datos maestros, tablas transparentes** y la Categoría de tamaño **0 – Registros de datos esperados 0 a 11000**.



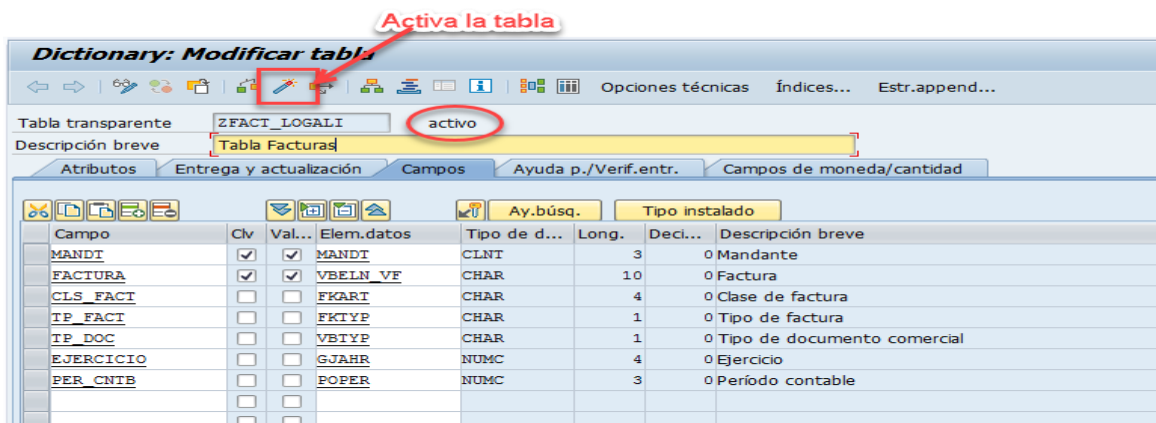
Grabamos las modificaciones y volvemos con la flecha atrás o con la tecla F3.



- E. En este momento se puede activar la tabla en base de datos. Para no tener advertencias en la activación configuramos las categorías de ampliación que se pueden encontrar en el menú en **Detalles**.



- F. Por último, activamos la tabla utilizando la opción de la barra de herramientas y se genera el objeto Tabla transparente (tabla de base de datos).





1.2. Campos de tipo importe y cantidad

En SAP los campos de tipo importe y cantidad tienen un tratamiento a parte porque el sistema es multilenguaje, preparado para ejecutar la misma aplicación en distintos idiomas y distintas monedas. Para entender cómo se crean los campos importe y cantidad utilizamos la tabla de base de datos *ZFACT_LOGALI* que hemos creado en la sección anterior realizando los siguientes pasos:

- A. Primero añadimos los campos *IMPORTE* y *CANTIDAD* en la pestaña **Campos**. También añadimos los campos correspondientes que pueden medir los valores del importe y de la cantidad, es decir un campo *MONEDA* para el importe y un campo *UNIDAD* para la cantidad. No es necesario crear los campos *MONEDA* y *CANTIDAD* en la misma tabla, lo que es imprescindible es indicar un campo *MONEDA* para el importe que puede estar creado en otra tabla, lo mismo ocurre con la *UNIDAD*. Para los campos indicados utilizaremos elementos de datos estándar.

Dictionary: Modificar tabla

Tabla transparente: ZFACT_LOGALI Inactivo(revisado)
Descripción breve: Tabla Facturas

Campos

Campo	Civ	Val...	Elem.datos	Tipo de d...	Long.	Deci...	Descripción breve
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Mandante
FACTURA	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	VBELN_VF	CHAR	10	0	Factura
CLS_FACT	<input type="checkbox"/>	<input type="checkbox"/>	FKART	CHAR	4	0	Clase de factura
TP_FACT	<input type="checkbox"/>	<input type="checkbox"/>	FKIYP	CHAR	1	0	Tipo de factura
TP_DOC	<input type="checkbox"/>	<input type="checkbox"/>	VBIYP	CHAR	1	0	Tipo de documento comercial
EJERCICIO	<input type="checkbox"/>	<input type="checkbox"/>	GJAHR	NUMC	4	0	Ejercicio
PER_CNTB	<input type="checkbox"/>	<input type="checkbox"/>	POPER	NUMC	3	0	Período contable
IMPORTE	<input type="checkbox"/>	<input type="checkbox"/>	WTGXXX	CURR	15	2	Valor total en moneda de transacción
MONEDA	<input type="checkbox"/>	<input type="checkbox"/>	WAERS	CUKY	5	0	Clave de moneda
CANTIDAD	<input type="checkbox"/>	<input type="checkbox"/>	MEGXXX	QUAN	15	3	Cantidad total
UNIDAD	<input type="checkbox"/>	<input type="checkbox"/>	MEINB	UNIT	3	0	Unidad de cantidad contabilizada

- B. Ahora tenemos que relacionar en la pestaña **Campos de moneda/cantidad** el campo moneda para el campo importe y el campo unidad para el campo



cantidad. En este caso los dos campos: *MONEDA* y *UNIDAD* se encuentran en la misma tabla, como se mencionaba no es obligatorio que estén en la misma tabla. En la pestaña de moneda/cantidad el sistema ha habilitado las columnas **Tabla Referencia** y **Campo Referencia** para los campos **IMPORTE** y **CANTIDAD**. Relacionamos los campos de referencia según en la imagen.

Dictionary: Modificar tabla

Tabla transparente: ZFACT_LOGALI activo

Descripción breve: Tabla Facturas

Atributos Entrega y actualización Campos Ayuda p./Verif.entr. Campos de moneda/cantidad

Ay.búsq. 1 / 11

Campo	Elem.datos	Tipo de datos	Tab. referencia	Campo ref.	Descripción breve
MANDT	MANDT	CLNT			Mandante
FACTURA	VBELN_VF	CHAR			Factura
CLS_FACT	FKART	CHAR			Clase de factura
TP_FACT	FKTYP	CHAR			Tipo de factura
TP_DOC	VBTP	CHAR			Tipo de documento comercial
EJERCICIO	GJAHR	NUMC			Ejercicio
PER_CNTB	EOPER	NUMC			Período contable
IMPORTE	WIGXXX	CURR	ZFACT_LOGALI	MONEDA	Valor total en moneda de transacción
MONEDA	WAERS	CUKY			Clave de moneda
CANTIDAD	MEGXXX	QUAN	ZFACT_LOGALI	UNIDAD	Cantidad total
UNIDAD	MEINB	UNIT			Unidad de cantidad contabilizada

Con esto hemos relacionado correctamente los campos *IMPORTE* y *CANTIDAD* indicando que el campo *IMPORTE* tiene su moneda en la misma tabla *ZFACT_LOGALI* en el campo *MONEDA* y que el campo *CANTIDAD* tiene la unidad igual en la misma tabla en el campo *UNIDAD*.



2. Instrucciones SQL de ABAP/4.

1.1. INSERT.

La sentencia **INSERT** permite introducir registros sencillos o el contenido de una tabla interna en una base de datos SAP.

INSERT <tab>.

Grabará en la BDD el registro de cabecera. Por tanto, previamente a esta instrucción moveremos los valores que queremos introducir sobre el área de trabajo de la tabla.

Si **SY-SUBRC = 0** Registro insertado.

Si **SY-SUBRC > 0** La clave del registro que queríamos insertar ya existía en la tabla.

También es posible introducir datos desde una tabla interna.

INSERT <tab> FROM TABLE <intab>

Si **SY-SUBRC = 0** Registros insertados.

Si existe algún registro en la base de datos con clave igual a algún registro de la tabla interna, se producirá un error de ejecución del programa.

La tabla interna podrá tener la misma estructura que la tabla de base de datos utilizando **INCLUDE STRUCTURE** en su declaración.



1.2. SELECT

La sentencia SELECT será la instrucción fundamental para leer información de la base de datos.

SELECT * FROM <tab> INTO TABLE <intab> (WHERE <cond>)

Almacena los datos en la tabla interna <intab>, sobrescribiendo los registros que pudiera tener previamente. Si queremos que respete los registros que tenía la tabla **interna** antes de realizar el SELECT tendremos que utilizar:

**SELECT * FROM <tab> APPENDING TABLE <intab>
(WHERE <cond>).**

Podemos indicar un orden en el proceso de selección de registros.

SELECT * ... ORDER BY <campo1> <campo2> ...

Si queremos seleccionar un registro para bloquearlo de posibles modificaciones.

SELECT SINGLE FOR UPDATE * FROM <tab>.