

# Curso 1º SMR

## Módulo: AO

### Actividades 3 (2) Bases de datos: Creamos Foreign Keys en BASE

#### ¿Qué pretendemos conseguir?

Ser capaces de crear Foreign Keys mediante la opción Relaciones.

Ser capaces de crear Foreign Keys mediante SQL.

#### ¿Qué necesitamos?

Openoffice Base.

Haber realizado las Sesiones 1,2 y 3 de Bases de Datos.

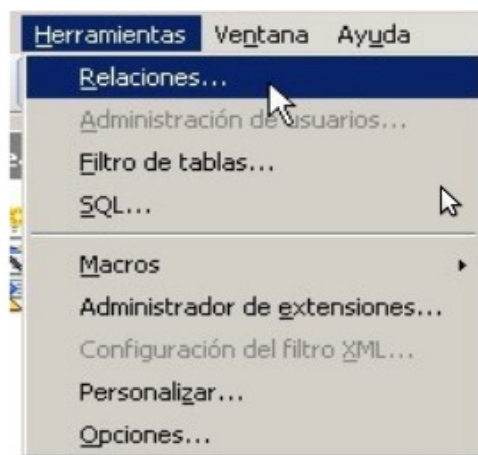
#### ¿Qué es una Foreign Key?

Una Foreign Key (FK a partir de ahora) es un campo cuyo valor es un Primary Key en otra tabla, es, por tanto, necesario que el valor se haya introducido previamente en la tabla de referencia.

El campo Primary Key de una tabla y el campo Foreign Key de la otra deben ser del mismo tipo ¿Podrías explicar por qué?

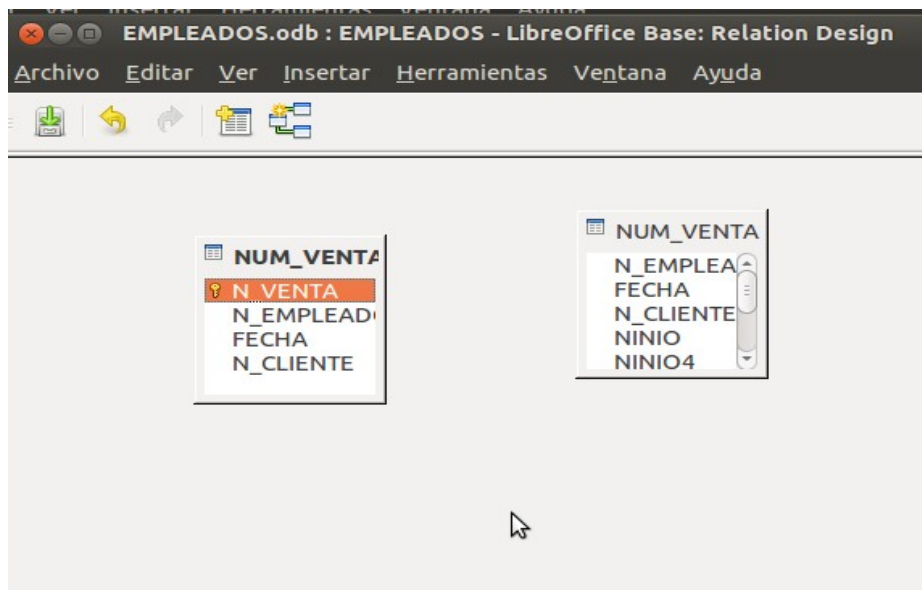
#### ¿Cómo podemos crear una Foreign Key?

A.-Mediante el campo Relaciones del menú Herramientas.



Una vez pulsada esta opción el programa nos pedirá que añadamos las tablas con las que vamos a trabajar.

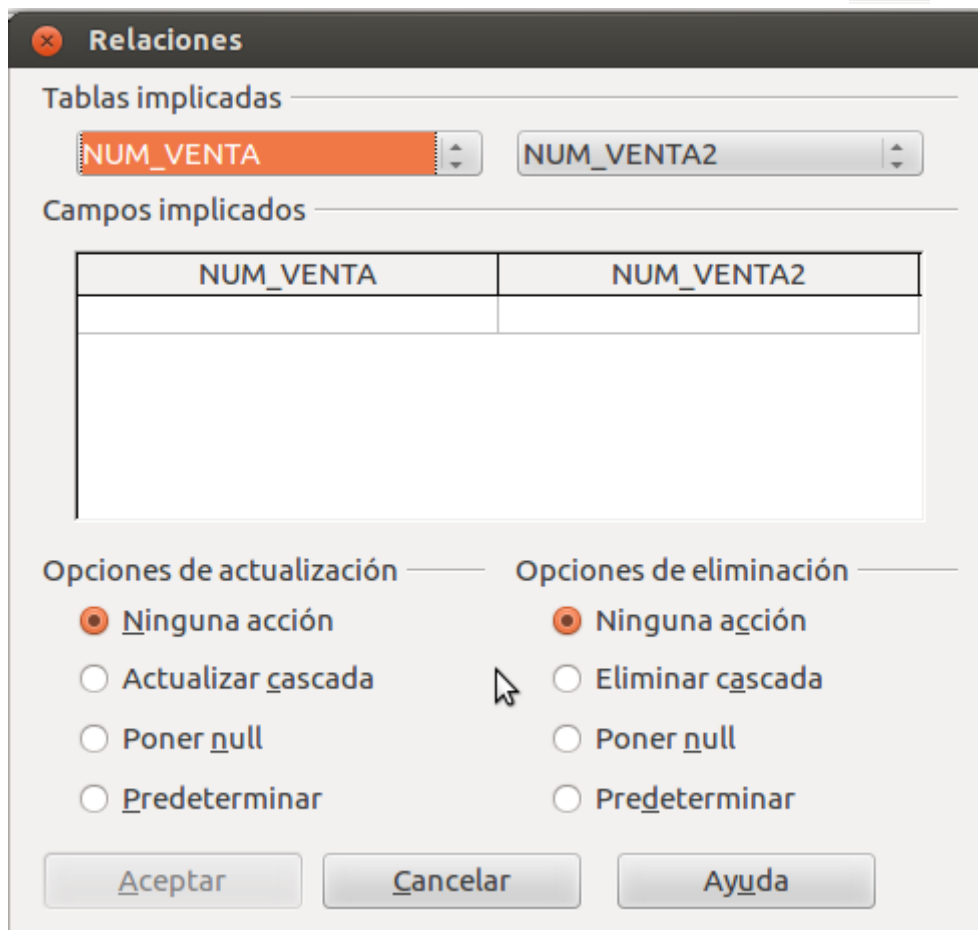
Una vez hecho esto se nos mostrará una pantalla similar a la siguiente:



Desde aquí podremos añadir más tablas:



Y podremos, ya, crear relaciones entre tablas:



Actividad 1: Crea la Tabla1 con los siguientes campos: ID1 Integer Valor Automático, CAMPO1 VARCHAR y CAMPO2 VARCHAR. Crea la Tabla2 con los siguientes campos ID2 Integer Valor Automático, CAMPO21 VARCHAR y CAMPO22 VARCHAR.

Accede a Relaciones y Añade las dos tablas.

Ahora crea la siguiente relación: Tabla1.ID1 con Tabla2.ID2. ¿Qué observas en el gráfico?

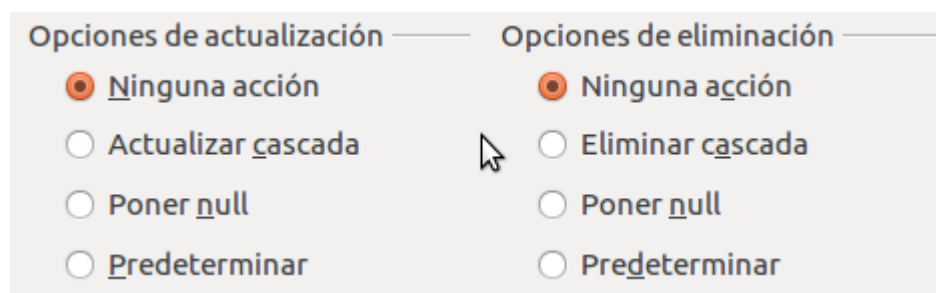
### ¿Cómo podemos eliminar una relación?

Bastará con pinchar en la flecha que indica la relación y seleccionar Eliminar.

Actividad 2: Elimina la relación creada en la Actividad 1.

Actividad 3: Añade, en Tabla2, el campo ID1 Integer pero no Automático. Vuelve a entrar en relaciones y ahora enlaza el campo ID1 de la Tabla1 con el ID1 de la Tabla2. Debe aparecer un 1 en la flecha justo al lado de Tabla1 y una "n" en la flecha junto a Tabla2. Explica por qué.

### ¿Para qué sirven las opciones que aparecen debajo de la ventana de Relación?



La columna Opciones de actualización nos indicará qué hará el motor de base de datos cuando se actualice el campo Primary Key en la tabla "padre".

- Ninguna acción: No permitirá actualizar el campo si existen valores en la tabla "HIJA".
- Actualizar cascada: cambiará el valor del Primary Key y todos los valores FK en la tabla "HIJA"
- Poner null: Al actualizar el campo PK de la Tabla "PADRE" dejará en nulo los campos FK de la tabla "HIJA".

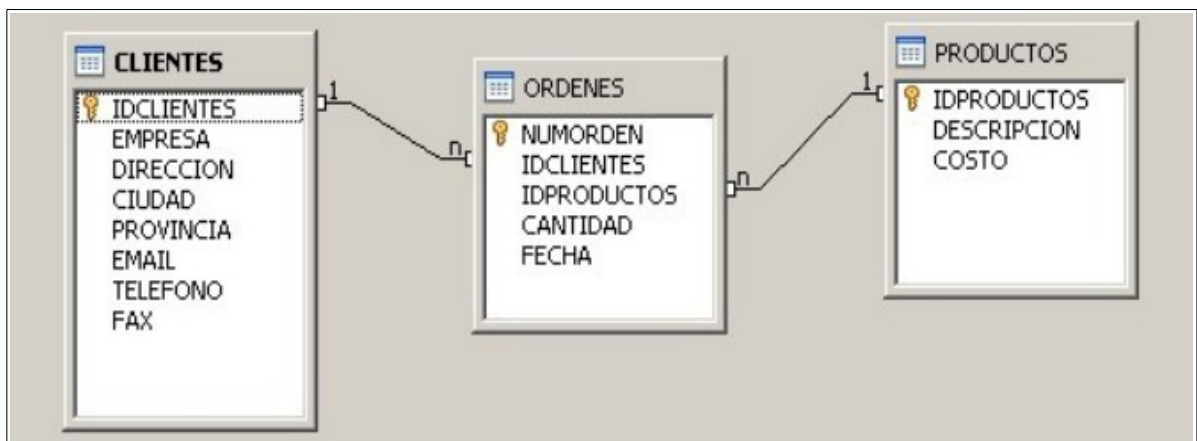
La columna Opciones de eliminación qué hará el motor de base

de datos cuando se elimine el campo Primary Key en la tabla "padre".

- Ninguna acción: No permitirá eliminar el campo si existen valores en la tabla "HIJA".
- Eliminar cascada: eliminará todas las filas de la tabla con Foreign Key que coincidan con el Primary Key de la tabla "Padre"
- Poner null: Al eliminar la fila PK de la Tabla "PADRE" dejará en nulo los campos FK de la tabla "HIJA".

Actividad 4: Con las Tabla1 y Tabla2 creadas anteriormente, y con la relación 1-n entre los campos Tabla1.ID1 y Tabla2.ID2 realiza todas las posibles combinaciones de eliminación, borrado, actualización. Crea una hoja de cálculo en la que indiques los resultados.

Actividad 5: Crea las siguientes tablas y sus relaciones:



El tipo de los campos a tu elección.

Actividad 6: Con el esquema de la Actividad 5 crea los siguientes TRIGGERS:

- Al borrar un cliente se eliminarán todas sus órdenes y los productos de éstas.
- Al modificar el IDCLIENTES se modificarán todos los IDCLIENTES que dependan de éste en la tabla ORDENES

Actividad 7: Con el esquema anterior y las modificaciones que creas oportunas, crea relaciones 1:1 entre las tablas CLIENTES-ORDENES y ORDENES-PRODUCTOS.

## B.-Mediante sentencias SQL.

Y la sentencia es esta:

```
ALTER TABLE <TABLA QUE CONTENDRÁ EL FOREIGN KEY>  
    ADD FOREIGN KEY (<NOMBRE DEL CAMPO FK>  
    REFERENCES <TABLA CON LA PK> (<CAMPO PK>)
```

Ejemplo:

Supongamos la Tabla1 con los campos:

```
ID1 integer pk  
CAMPO1 VARCHAR(10)
```

Y la Tabla2 con los campos:

```
ID2 integer pk  
ID1 integer  
CAMPO2 VARCHAR(20)
```

Si queremos que Tabla2.ID1 sea FK respecto a Tabla1.ID1 tendremos que hacer lo siguiente:

```
ALTER TABLE TABLA2  
    ADD FOREIGN KEY (ID1)  
    REFERECES TABLA1 (ID1)
```

Para que la relación se vea desde la opción Relaciones del menú principal de BASE es necesario actualizar las tablas antes.

El campo Foreign Key debe existir previamente en la tabla.

Actividad 8: Elimina las relaciones de la Actividad 5 y vuelve a crearlas mediante SQL.

Actividad 9: Realiza una violación de FK (intenta introducir un valor que no se encuentra en la PK de la tabla referenciada) y apunta el mensaje que aparece.

Actividad 10: ¿Cómo crees que se borrarían Relaciones mediante SQL? Inténtalo. Busca información sobre eliminación de FK. ¿Para qué crees que nos puede servir el mensaje "rarito" que nos ha mostrado BASE en la Actividad 9?