



We are ready



Formación

Consultas básicas con DataCycle Reporting

APESOFI

Formación Consultas Básicas con DataCycle Reporting

© APE Software Components
08290 Parc Tecnològic del Vallès
Tel: 902 789 981 www.apesoft.com

Índice

| | |
|--|----|
| El modelo relacional | 3 |
| Componentes del modelo relacional..... | 3 |
| Bases de Datos..... | 3 |
| Entidades..... | 3 |
| Registros..... | 3 |
| Campos..... | 4 |
| Claves (primaria y foránea)..... | 4 |
| Relaciones..... | 4 |
| Integridad referencial..... | 5 |
| Vistas..... | 5 |
| Teoría de conjuntos..... | 5 |
| Lenguaje SQL..... | 6 |
| Traducir las necesidades del usuario..... | 6 |
| Situación de partida: Empresa Neptuno | 7 |
| Creación de un proyecto..... | 8 |
| Conexión con una base de datos..... | 11 |
| Conclusiones..... | 15 |
| La primera consulta | 16 |
| Todos los empleados..... | 16 |
| Editor de Consultas Modo Visual..... | 20 |
| Ejercicios..... | 28 |
| Consultas simples..... | 28 |
| Filtrando los datos..... | 31 |
| Consultas con varias tablas | 33 |
| Detalle de ventas..... | 33 |
| Añadir un parámetro..... | 41 |
| Añadir un parámetro tipo Lista de Valores..... | 46 |
| Categorías y productos sin existencias..... | 51 |
| Importar catalogo..... | 52 |
| Añadir Parámetro tipo Consulta de Base de Datos..... | 63 |
| Gestión de carpetas..... | 69 |
| Ejercicios..... | 71 |
| Consultando múltiples tablas..... | 71 |
| Sumarizando y agrupando los datos..... | 72 |
| Filtrar datos agrupados..... | 73 |
| Union..... | 74 |
| Subconsultas..... | 75 |
| Prácticas | 77 |
| Anexos | 79 |
| Anexo: Consultas con asistente (Tradicional)..... | 79 |

El modelo relacional

Anatomía de una Base de Datos Relacional

Componentes del modelo relacional

Bases de Datos

Una base de datos es un almacén en donde se encuentra la información corporativa, ya sea datos maestros como productos, clientes, proveedores ... como información sobre las operaciones realizadas en su empresa: altas, compras, facturas, ventas, albaranes, órdenes de fabricación ...

DataCycle Reporting definirá la conectividad hacia las Bases de Datos corporativas, que engloba la estructura de campos de sus tablas (Entidades) y las relaciones entre esas tablas (joins).

Entidades

Las entidades contienen la información de su corporación formando un conjunto de registros. Distinguimos entre entidades con datos maestros como productos, clientes o proveedores y entidades con las operaciones de su negocio como compras, ventas o órdenes de trabajo.

En DataCycle, una entidad es equivalente a un fichero de AS/400 o a una tabla en Oracle. DataCycle sólo guarda la estructura de campos, sin importar los datos.

Registros

Las entidades están formadas por registros que recogen la información corporativa de forma individual. En las tablas de facturas tendremos un registro con cada factura, en las de productos cada producto estará en un registro y así sucesivamente.

Campos

Cada entidad tiene definida una estructura de campos que determina la información que se almacena y de que tipo será.

En la entidad de productos habrá campos donde almacenar el nombre del producto, su coste, precio y proveedor.

Claves (primaria y foránea)

Cada registro debe ser direccionable de forma única. Es por ello que en toda entidad habrá una serie de campos que determinen de forma unívoca los registros que lo componen.

El dni para los clientes, un código de producto o el número de factura son tres ejemplos de claves primarias.

La clave foránea es un campo que relaciona los registros de una entidad con los de otra entidad. Por ejemplo en una Base de Datos de una biblioteca tendremos entidades de libros y de autores y las claves foráneas relacionarán unos con otros.

Relaciones

Una relación es una unión entre dos entidades tal y como se ha mostrado en el ejemplo de los libros y autores o entre una factura y sus líneas.

La clave primaria de una entidad debe aparecer como clave foránea en la otra entidad.

Se distinguen las siguientes formas de relacionar entidades:

1:1

Por cada registro de una entidad siempre encontraremos un solo registro en la otra entidad. Este caso nos lo encontramos cuando la información sobre una entidad es muy elevada y se trocea, por ejemplo podemos tener una entidad con información administrativa de los clientes y otra con la información comercial de los clientes. Por tanto por cada registro en una entidad habrá un único registro en la otra.

1:N

Es el caso más habitual. Por cada registro en una entidad encontraremos muchos registros relacionados en otra entidad. Por ejemplo un cliente tiene muchos pedidos, un proveedor muchos productos, una provincia muchas provincias ...

N:M

Esta relación se establece entre aquellas entidades en el que registros de ambas pueden estar relacionados con múltiples registro de la otra entidad. Un ejemplo sería tener la entidad de Alumnos y Asignaturas en donde un alumno puede estar matriculado a varias asignaturas y una asignatura tendrá varios alumnos.

Este caso no es soportado por las Bases de Datos relacionales tradicionales, debiendo ser substituidas por una nueva relación con tres tablas de tipo 1:N M:1.

Un caso particular sería el 0:M por ejemplo cuando un vendedor aún no ha hecho ventas o un alumno se ha dado de alta pero aún no se ha matriculado en ningún curso.

Integridad referencial

La coherencia de la Base de Datos se consigue si todas las relaciones entre las entidades están correctamente informadas. Por ejemplo no puede existir un pedido a un cliente que aún no haya sido introducido en el sistema.

Para asegurar la Integridad Referencial se debe adoptar determinadas políticas al realizar operaciones de modificación y borrado de campos clave y registros. Las posibilidades son las siguientes:

Borrado en cascada: En caso de borrar un registro deben borrarse todos los registros relacionados. Por ejemplo si se elimina una asignatura se deben eliminar la información sobre la matriculación a la misma.

Modificación en cascada: En caso de modificar la clave primaria se deberán modificar la clave foránea relacionada en todos aquellos registros en los que aparezca.

Se impide el borrado y modificación: mientras exista registros relacionados no se permite el borrado o eliminación de los registros principales.

Vistas

Una Base de Datos y sus entidades pueden contener información sensible y privilegiada. Una vista es una forma de enmascarar la información, mostrando tan solo un subconjunto de ella a las personas que desee.

Teoría de conjuntos

Todas las operaciones de consulta realizadas sobre las entidades se basan en la teoría de conjuntos.

En toda consulta se determina los campos que componen la respuesta. En conjunto de campos devueltos corresponderá a la respuesta buscada o podrá ser manipulado mediante uniones, intersecciones o diferencias para la obtención de nuevos resultados.

Lenguaje SQL

El lenguaje SQL es el que permite interrogar las Bases de Datos y resolver las consultas.

Aunque SQL es un lenguaje con vocación universal y estándar, han aparecido numerosas versiones y cada Base de Datos dispone de la suya propia. Las diferencias entre ellas son mínimas pero deben ser tenidas en cuenta al diseñar sus consultas. Posteriormente se muestra algunas de las diferencias existentes entre ellas.

Los ejemplos que se muestran a continuación están escritos en SQL de MS Access.

Traducir las necesidades del usuario

Una de las partes más complejas consiste en comprender el lenguaje en el que se expresa las necesidades de consulta de las Bases de Datos. Normalmente son expresadas por personal que no conoce las características informáticas ni la forma de interna de guardar los datos y que normalmente solo expresan el enunciado del problema. Por ejemplo:

Me han pedido que haga un estudio sobre los productos deficitarios

Que categorías son más rentables para la empresa

En que delegación se prevé problemas en el suministro de los pedidos

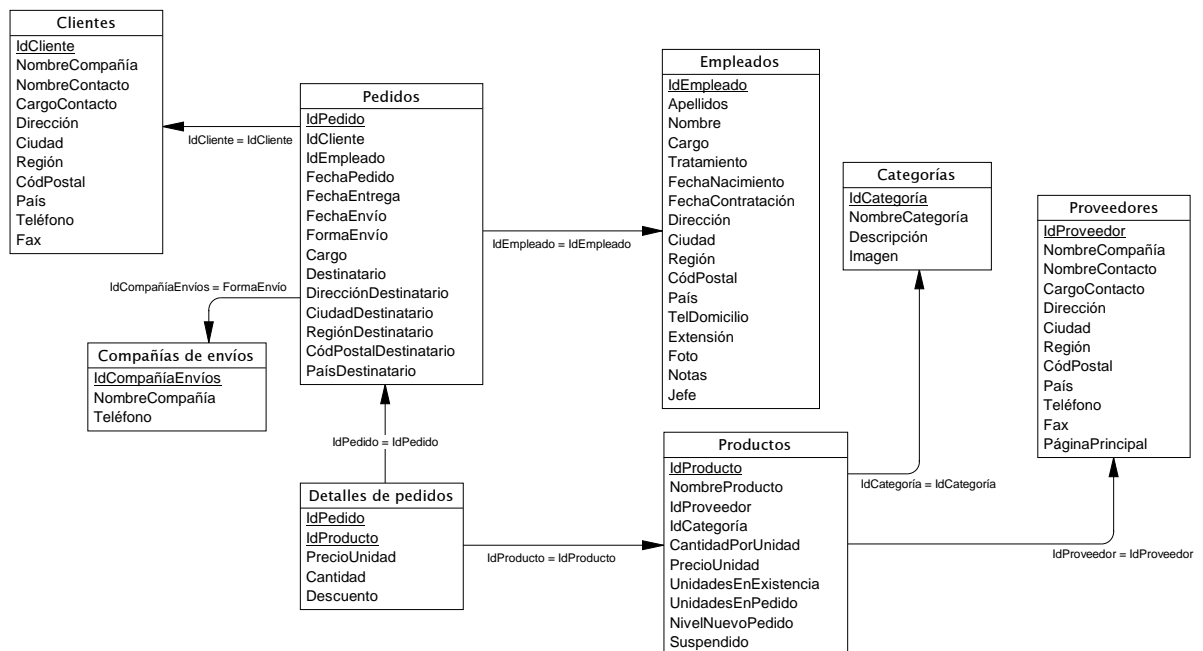
Sobre este aspecto poca ayuda podemos dar. Solo la experiencia y el conocimiento particular de usuarios y de la empresa pueden servir de ayuda.

Situación de partida: Empresa Neptuno

Descripción del ejemplo que vamos a seguir, creación de proyectos y definición de Bases de datos

Este tutorial se basará en los datos de una base de datos en formato Microsoft Access llamada neptuno.mdb. Esta base de datos, instalada de forma automática durante el proceso de instalación de DataCycle Reporting, contiene la información de una empresa que compra y vende productos alimentarios a todo el mundo. Básicamente se controlan los pedidos que la empresa efectúa.

El modelo relacional no forma parte del temario de esta presentación. Su estudio forma parte de un curso especializado de SQL.

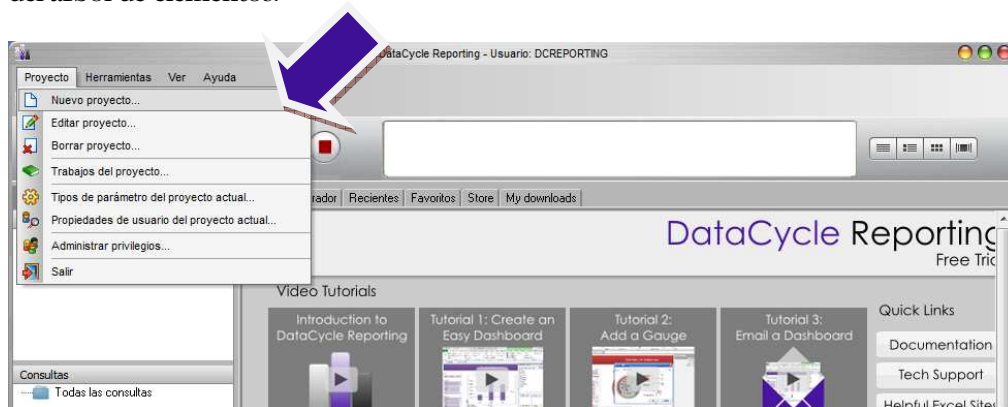


Creación de un proyecto

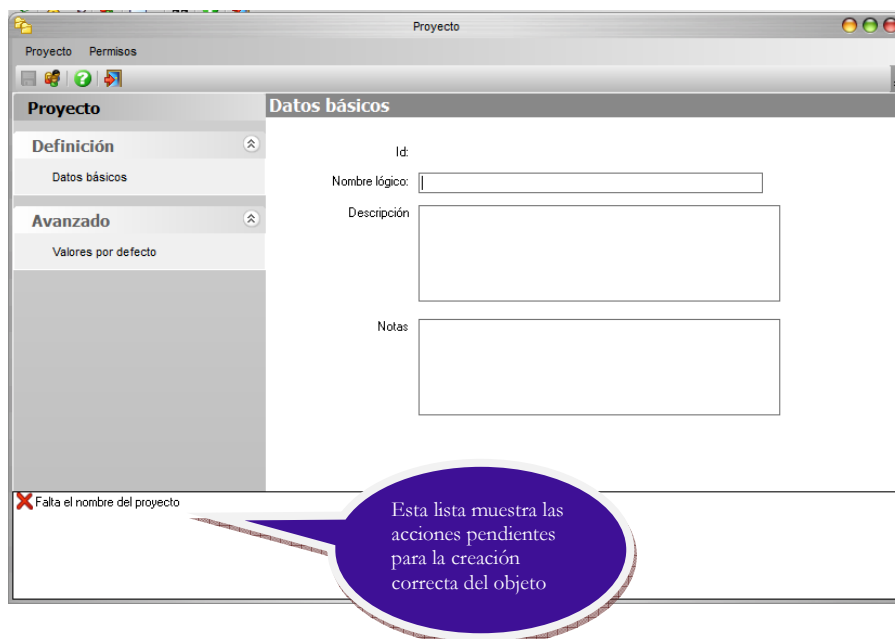
Para empezar, necesitamos un proyecto donde crear nuestros informes de ejemplo. Como hemos definido en el capítulo anterior, un proyecto es un conjunto de: Entorno de datos, informes y planificaciones.

Existe un proyecto en su repositorio instalado por defecto, denominado “**Tutorial**”, donde se encuentran completamente desarrollados todos los ejemplos de esta guía de evaluación.

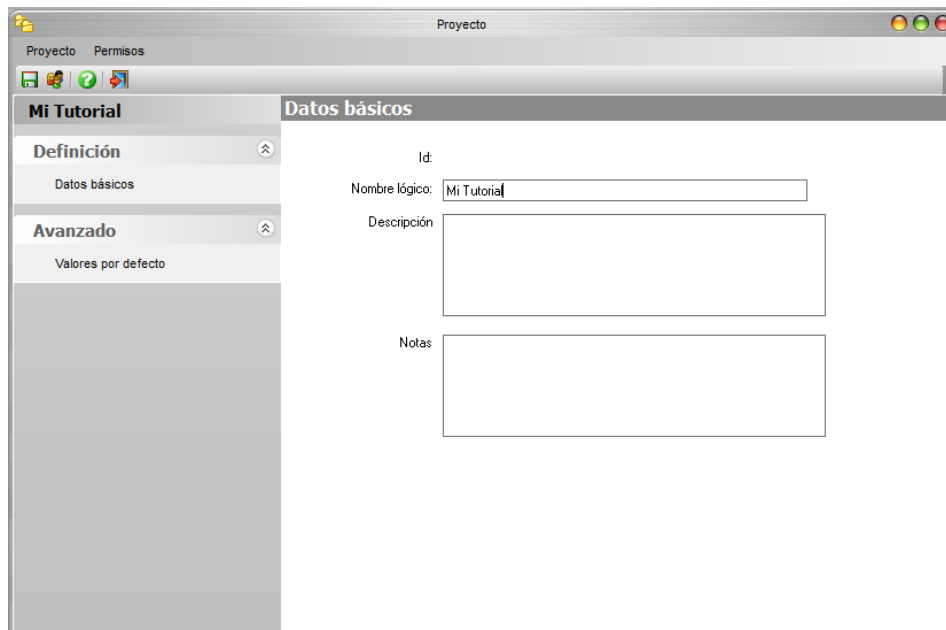
Vamos a crear un proyecto llamado **Mi Tutorial**. Esta operación se realiza de forma muy sencilla, seleccionando la opción **Nuevo proyecto** del menú Proyecto. Este menú aparece efectuando un clic con el botón derecho del ratón sobre el elemento del árbol de elementos:



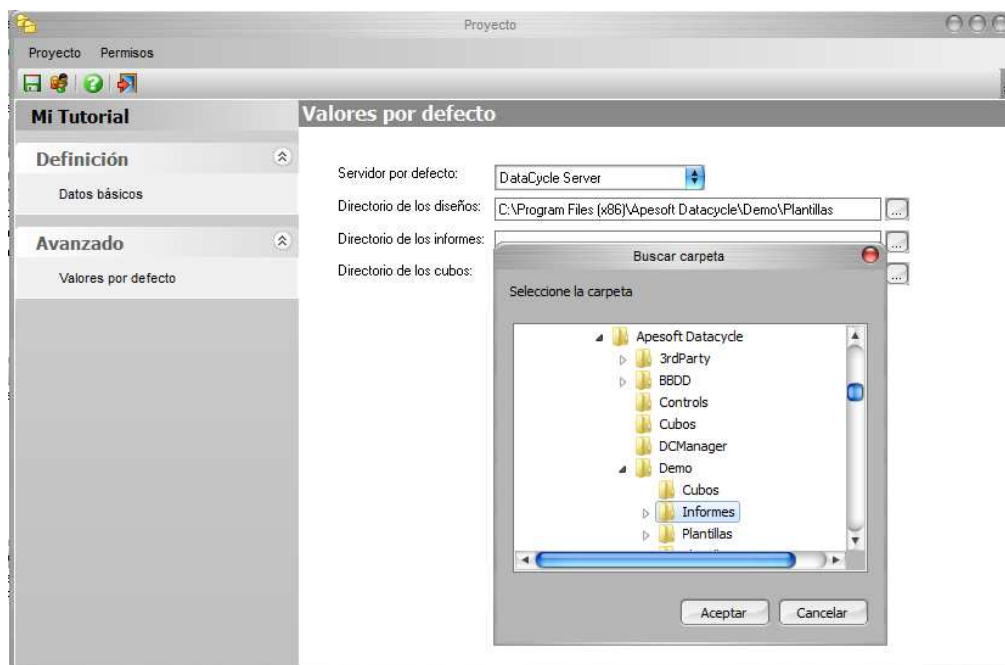
Seguidamente aparecerá la ventana de edición de un proyecto nuevo:





En los datos básicos de configuración podemos entrar el nombre del proyecto: **Mi Tutorial**.



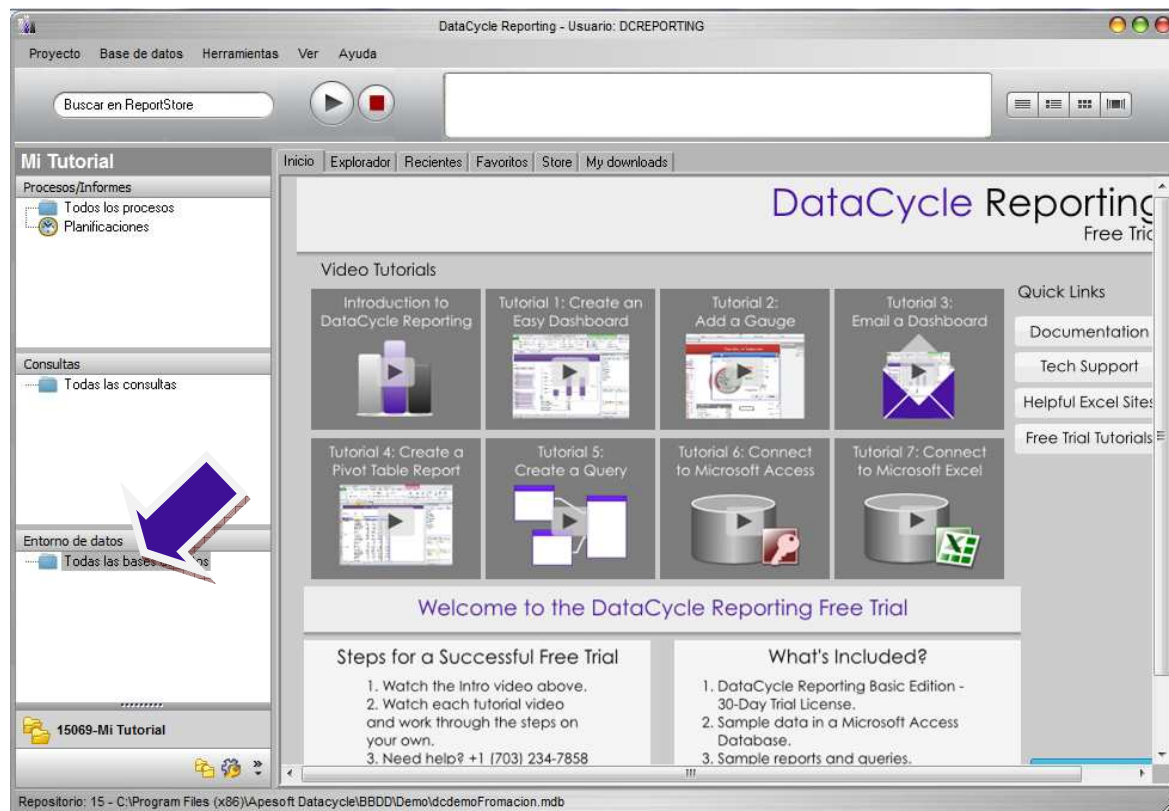
En los valores por defecto de la opción **Avanzado** informaremos del servidor, y los directorios de las plantillas e informes del proyecto, tal como se indica a continuación:



Estos valores serán los valores que se cogerán por defecto al crear los procesos.

Pulsamos el icono de guardar  para dar de alta el nuevo proyecto. Finalmente, para cerrar la ventana pulsaremos el icono  (salir).

El árbol reflejará el nuevo proyecto:

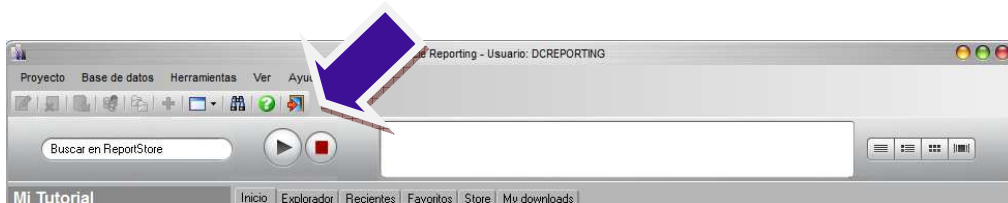


Vamos a detallar el contenido del proyecto **Mi Tutorial**.

Como podemos observar, dentro de cada proyecto hay carpetas especiales que contendrán:

- Los **Procesos** que generarán los informes en un formato enriquecido.
- Las **Planificaciones** de los procesos para que se generen los informes de forma automática a ciertas horas y ciertos días.
- Las **Consultas** que obtendrán la información de las bases de datos.
- Los **Entornos de datos**, base de datos de donde se generarán informes.

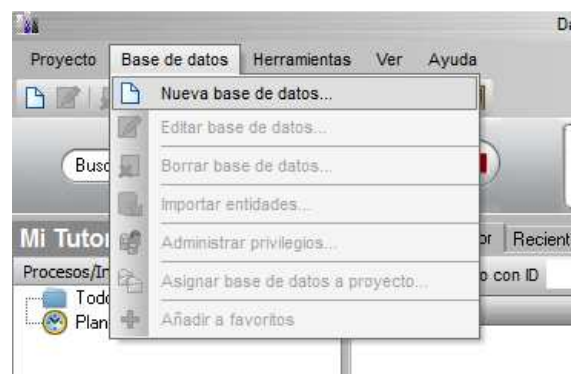
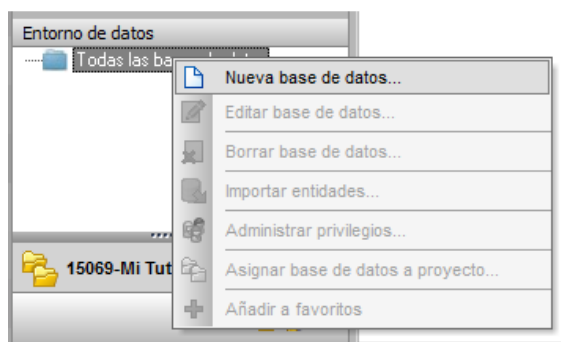
Por defecto la barra de herramientas está oculta, para activar esta barra se hace clic con el botón derecho del ratón en el menú.



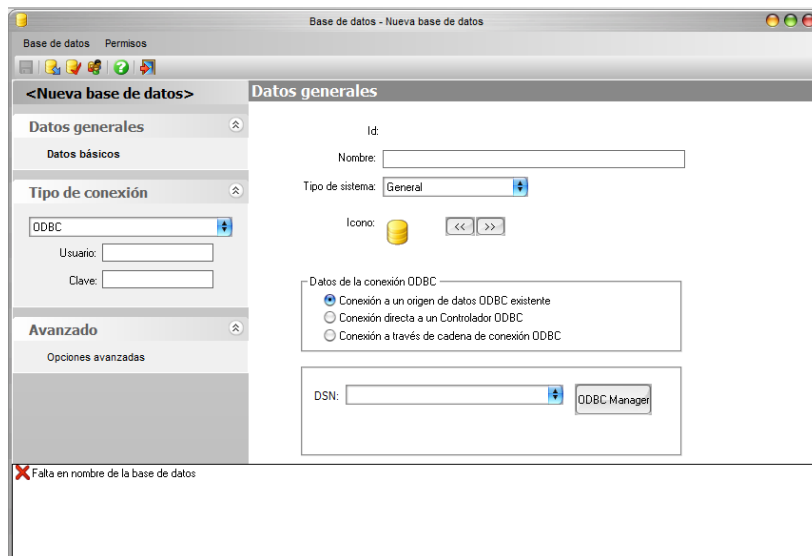
Conexión con una base de datos

Vamos a definir la base de datos de donde se van a generar informes. En nuestro tutorial será una base de datos en formato Microsoft Access que el programa de instalación ya se ha encargado de instalar. Esta base de datos se llama **neptuno.mdb** y puede que la reconozca como la base de datos que viene de ejemplo en toda instalación de Microsoft Office. En una situación real esta base de datos sería su AS400, su Oracle, su SQL Server,... (Incluso pueden ser varias bases de datos a la vez).

Para definir la base de datos Neptuno, pulsaremos la opción **Nueva** del menú Bases de Datos. Para ello realizamos una operación similar a la de creación de un proyecto que hemos efectuado anteriormente: realizamos un clic con el botón derecho del ratón sobre el elemento del árbol **Todas las bases de datos** y seleccionamos la opción **Nueva base de datos...** del menú que aparece, también podemos utilizar la primera opción del menú **Base de datos**.

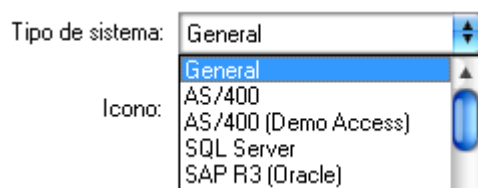


Aparecerá la ventana de edición de base de datos vacía:

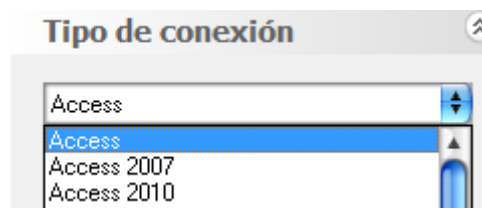


Vamos a informar los siguientes campos:

- En el nombre introduciremos el nombre de la base de datos: **BD Neptuno**
- Dejamos el tipo de sistema a **General**



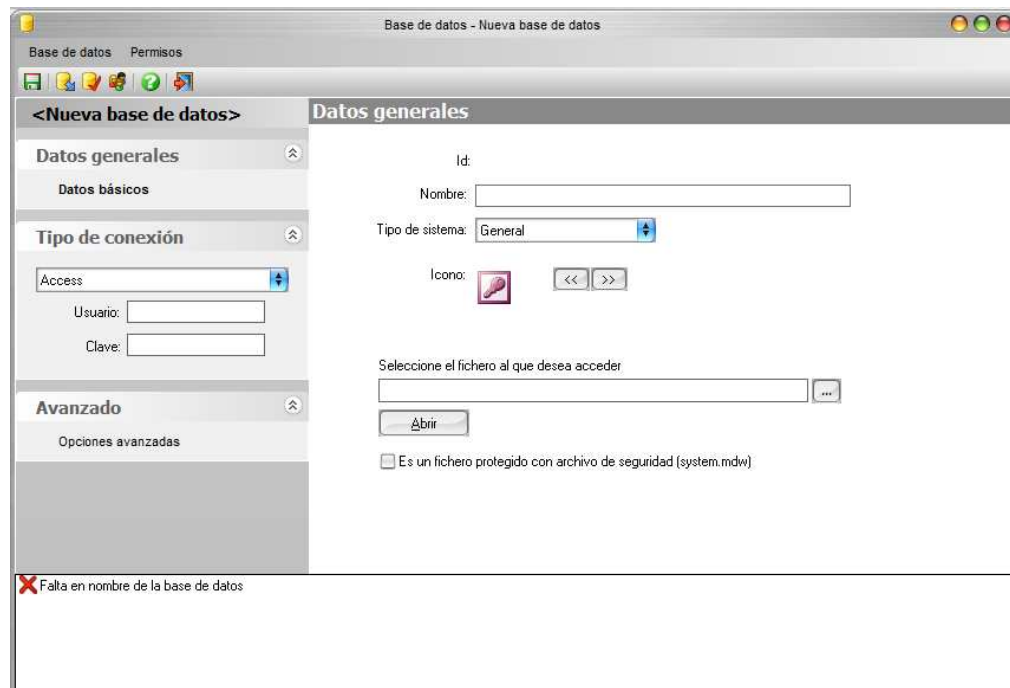
- Cambiamos el tipo de conexión a **MS Access**



- Para facilitar las futuras tareas, podemos asignar un icono identificativo a la base de datos pulsando los botones << y >>, hasta encontrar el icono que más se adecúe a la base de datos.



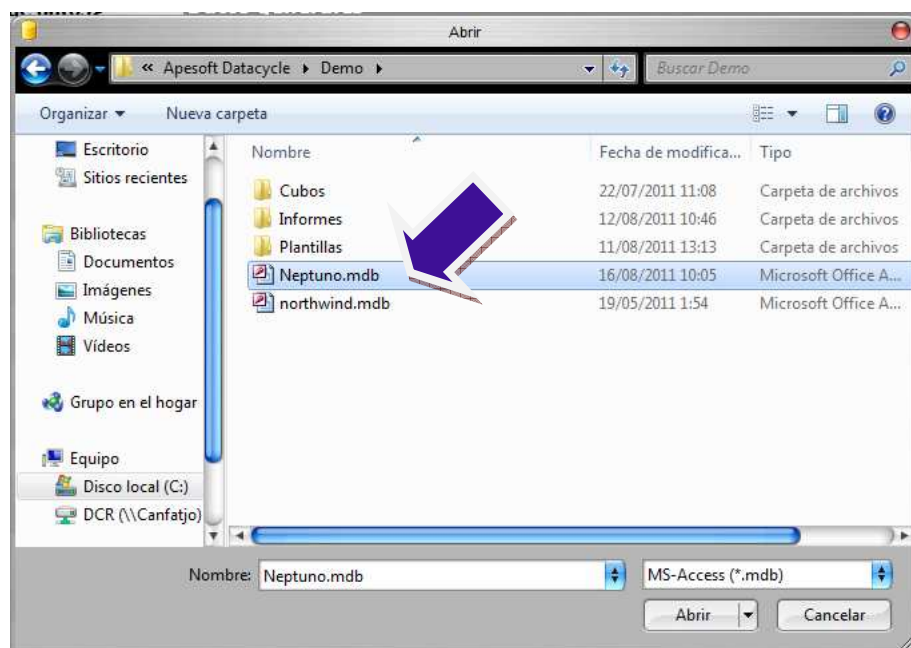
Observe que la ventana ha cambiado al seleccionar base de datos Access:



Asignamos “Neptuno” como nombre de la nueva Base de Datos.

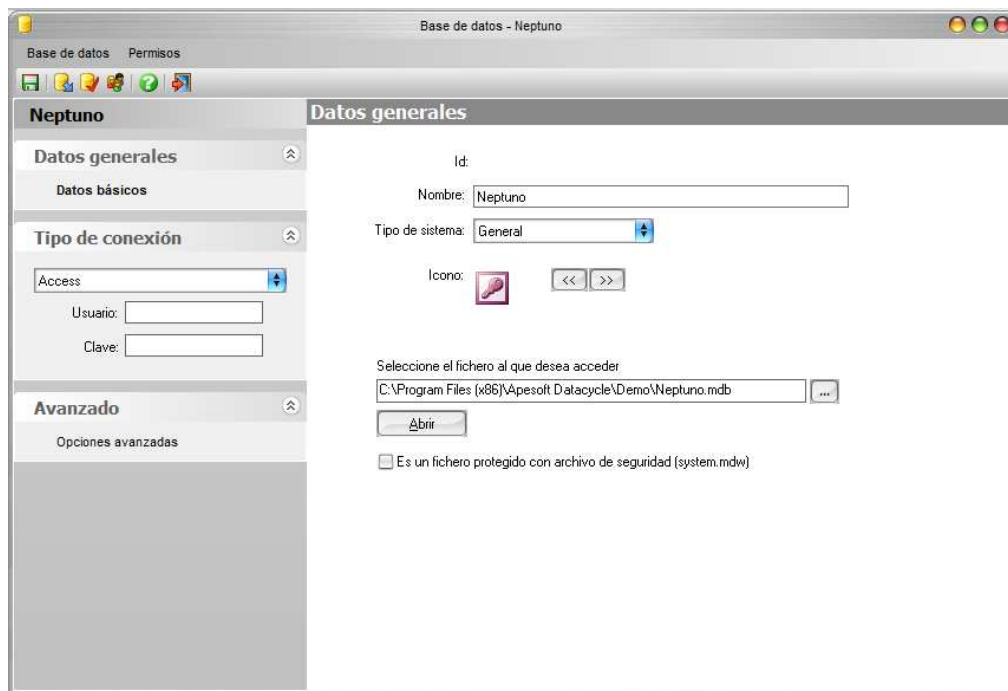
Seleccionaremos el fichero de base de datos **neptuno.mdb** pulsando el botón de selección de fichero: ...

Buscaremos el fichero **neptuno.mdb** dentro del directorio **Demo** que se encuentra a su vez en el directorio **Apesoft DataCycle** donde se haya instalado el producto durante la fase de instalación.

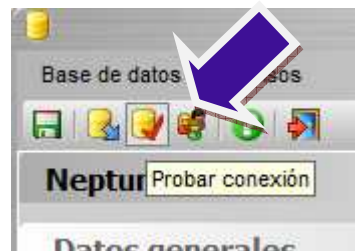
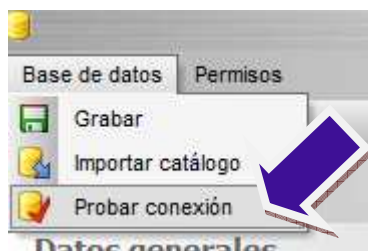


Pulsamos el botón **Abrir** para seleccionarlo.

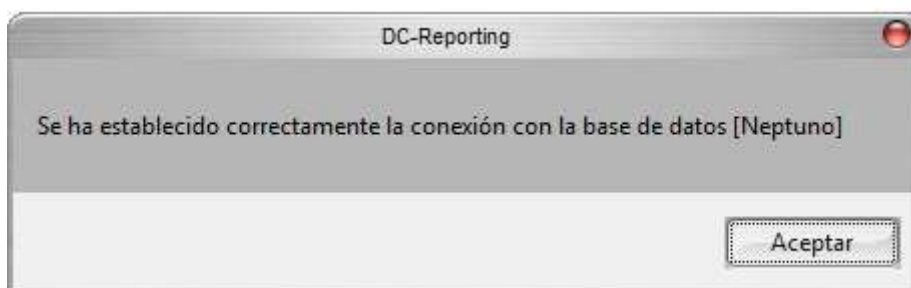
La ventana quedará así:



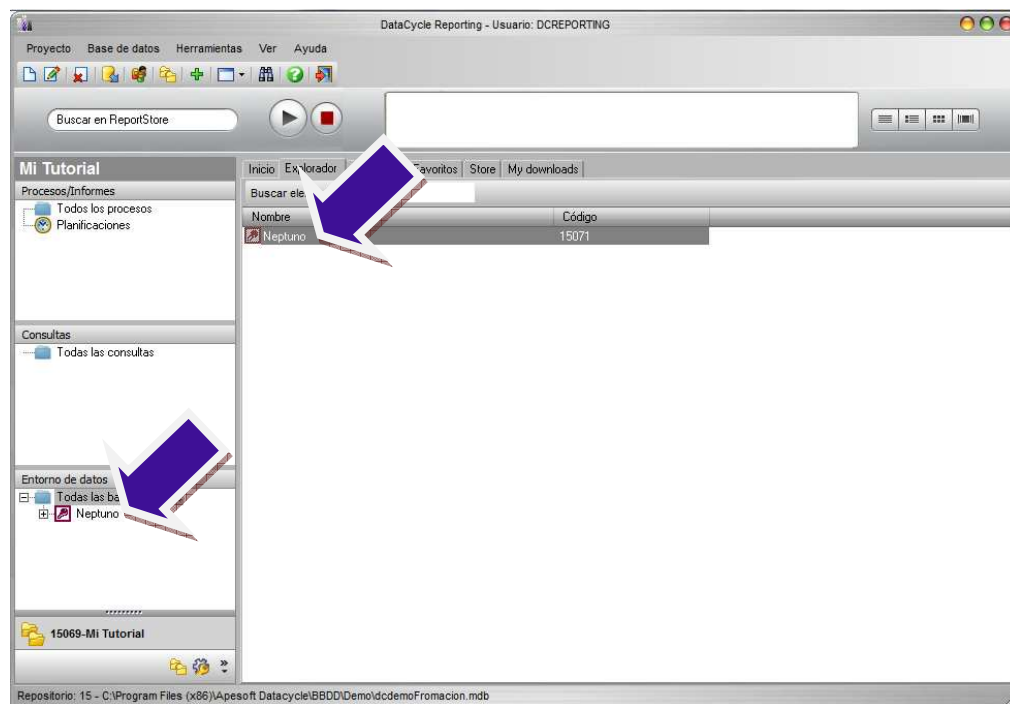
Ahora podemos pulsar la opción **Probar Conexión** desde el menú principal Base de Datos para probar si DataCycle puede efectivamente conectarse a la base de datos.



Debería aparecer el siguiente mensaje confirmándolo:



Cerramos esta ventana de confirmación, y finalmente grabamos y cerramos la ventana de creación de la base de datos. Podemos observar ahora que la nueva base de datos la hemos dado de alta en nuestro proyecto.



Conclusiones

En este capítulo hemos aprendido a crear un proyecto y a definir las bases de datos que hay dentro de él.

La creación de proyectos es una tarea que realizaremos una sola vez, al principio. Normalmente en una empresa existirá un solo proyecto de Reporting.

La definición de bases de datos es un proceso que se realiza pocas veces, aunque al principio es imprescindible.

La primera consulta

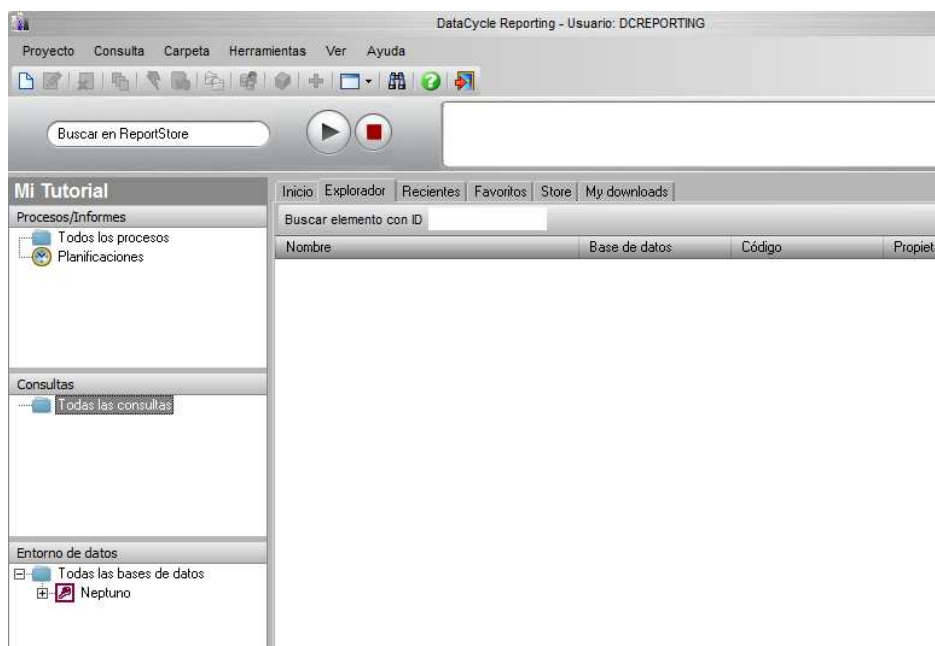
Vamos a desarrollar una consulta sencilla que recuperará la lista de empleados de la base de datos Neptuno.

EL objetivo para este capítulo va a ser crear nuestra primera consulta para obtener una lista de empleados de la base de datos Neptuno. Aunque esta consulta tendrá utilidad por sí misma, será la base para la creación de nuestro primer informe en el capítulo siguiente.

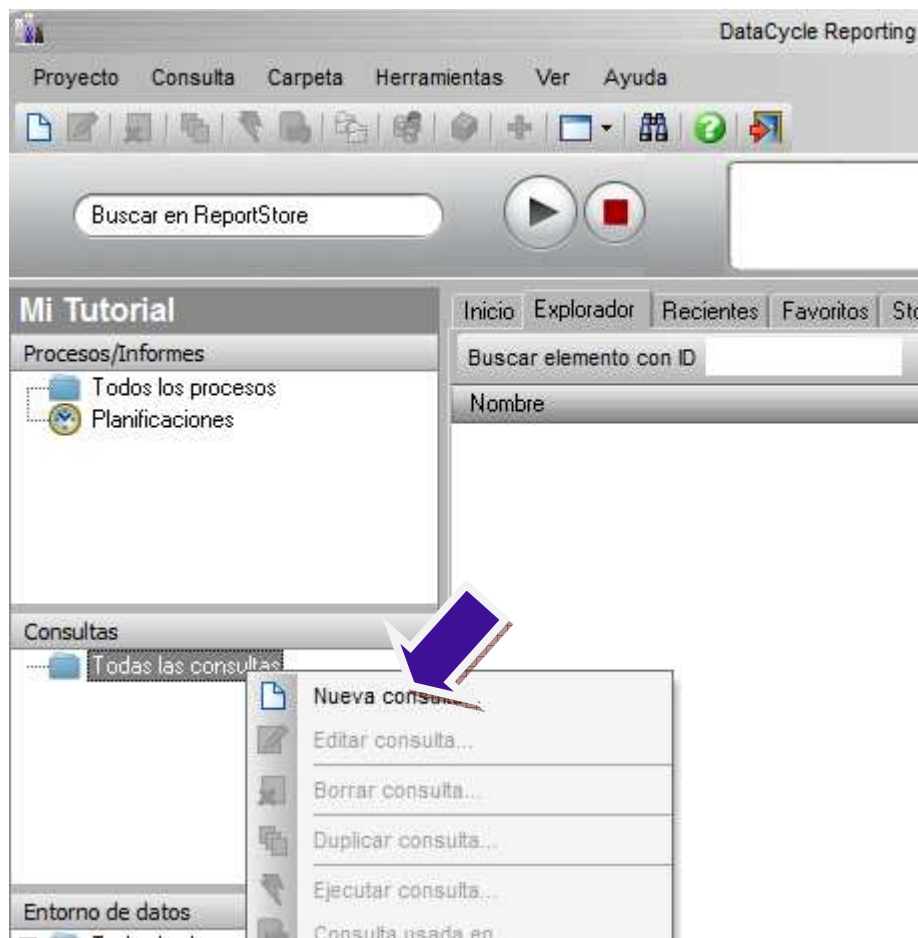
Todos los empleados

En DataCycle Reporting, las consultas son objetos reutilizables que pueden ser usados en diferentes procesos de generación de informes. Podríamos decir que las consultas crean una capa de abstracción que esconde la complejidad que existe en las bases de datos.

Localizamos el proyecto “Mi tutorial” y seleccionamos el panel de las consultas.



Activamos el menú de contexto de Consultas pulsando el botón derecho del ratón sobre el elemento **Todas las consultas** y seleccionamos la opción “Nueva Consulta”:



Al tratarse de una consulta nueva, se muestra la siguiente pantalla en donde debe indicar contra qué Base de Datos se realiza la consulta, en nuestro caso solo aparece Neptuno que es la única Base de datos que hemos dado de alta.



Seleccionamos la base de datos Neptuno. Al pulsar **Aceptar**, aparecerá la pantalla de edición de consultas:

Consulta - Nueva consulta

Consulta Permisos

Consulta

Definición

Datos básicos

Base de datos origen

Neptuno

Datos de la consulta

Habilitar el análisis (parsing) de la sentencia SQL para identificar los campos a retornar

Editar consulta (Modo visual)

Editar consulta con el asistente...

Avanzado

Opciones avanzadas

Otros parámetros

Datos básicos

Id: 0

Nombre:

Descripción técnica

Descripción para el usuario

Campos de la consulta

| Nombre lógico | Nombre físico | Expresión SQL |
|-----------------------------|---------------|---------------|
| La consulta no tiene campos | | |

Identificar campos Documentar campos

✘ Debe especificar el nombre de la consulta

✘ Debe crear la consulta SQL

En la ventana de edición de la consulta configuraremos todos los campos que determinaran el comportamiento de la misma. Es decir:

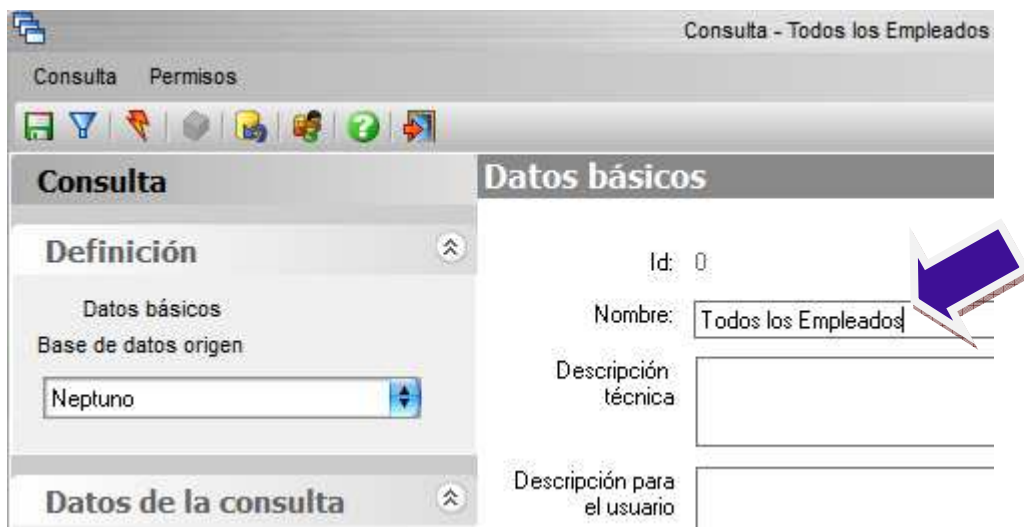
- **La base de datos** sobre la que se realizará la consulta. Podremos escoger cualquier base de datos que hayamos definido anteriormente (ver capítulos anteriores).
- **Diseñar consulta** para especificar los criterios de selección de la información deseada.
 - **Editar Consulta (Modo Visual)**
 - **Editar Consulta con el asistente**
- **Habilitar el análisis (parsing) de la sentencia SQL** para identificar automáticamente los campos que retorna la consulta. Puede suceder que para ciertas consultas complejas (unions, subselects, etc.) el análisis no sea capaz de identificar los campos de salida. En esos casos habilitaremos el **parser avanzado** de las opciones avanzadas
- **Los campos (columnas) a retornar** por la consulta. Cada vez que se modifica la sentencia SQL, DataCycle automáticamente identifica los campos que devolverá y asigna las descripciones que hayan sido documentadas en cada una de las entidades (tablas) en que se basa la consulta. Las descripciones de los campos a retornar podrán ser documentadas manualmente.



Los diferentes dialectos de SQL y Jet. Aunque SQL es el lenguaje estándar para la interrogación de bases de datos relacionales, en la práctica, las particularidades de los diferentes productos de base de datos del mercado ha provocado que cada de ellas haya ampliado la sintaxis SQL común con sus propias extensiones. Así por ejemplo, para convertir un valor de tipo cadena a tipo numérico, en ORACLE se utiliza la función TO_NUMBER mientras que en MS Access usaremos la función INT.

Como respuesta a la multitud de dialectos y para facilitar el desarrollo de aplicaciones multi-base de datos, Microsoft desarrolló la librería Jet, que implementa un dialecto SQL propio (el mismo que utiliza MS Access) y que permite acceder a múltiples tipos de base de datos sin preocuparse por la sintaxis particular de cada una de ellas. No es difícil intuir que la contrapartida a la independencia de la base de datos se encuentra en un menor rendimiento de las consultas (según los casos, muy importante) y en la imposibilidad de aprovechar recursos y prestaciones particulares de cada base de datos (funciones particulares, procedimientos almacenados, etc.)

Empezaremos asignando un título descriptivo a la consulta. La llamaremos **Todos los empleados**.

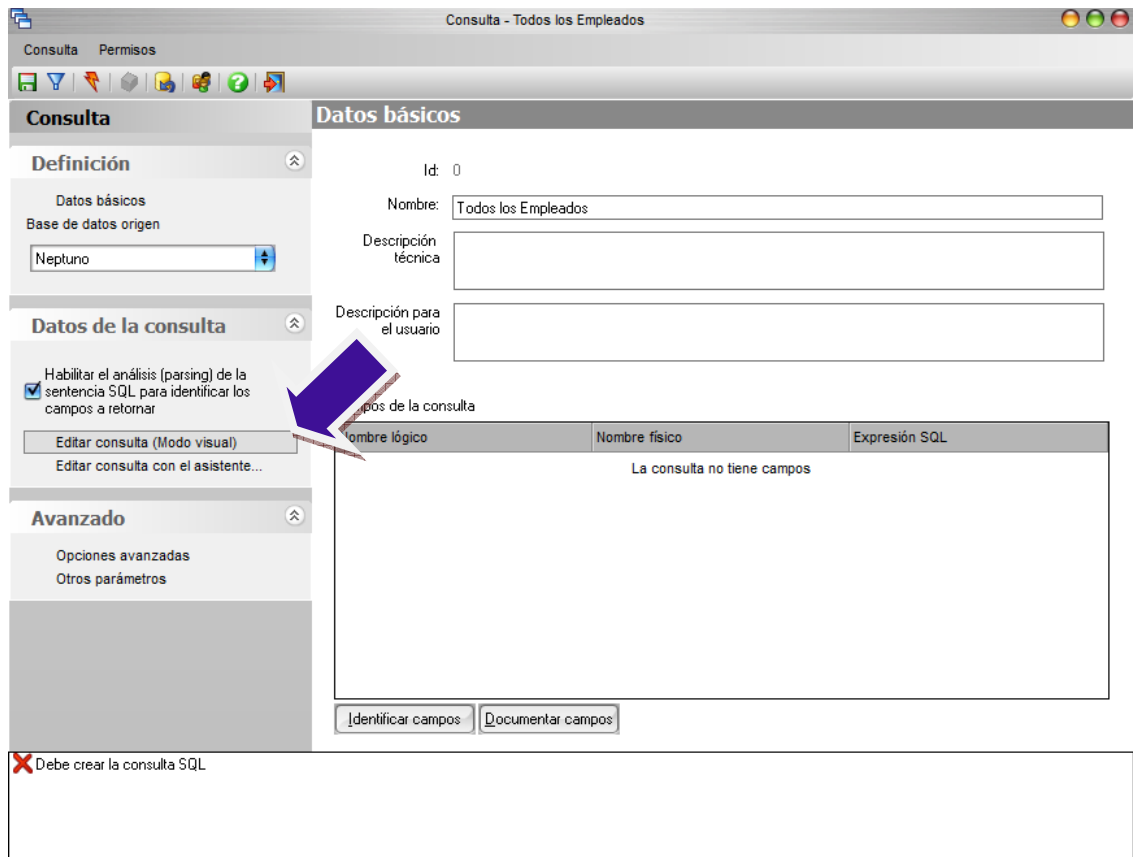


Para diseñar la consulta SQL de recuperación de datos de la base de datos utilizaremos el asistente de consultas de DataCycle.

DataCycle Reporting incluye dos asistentes que facilitan el diseño y prueba de las consultas SQL:

- Modo visual (recomendado): Potente interfaz gráfico donde sin tener grandes conocimientos de SQL se pueden diseñar fácilmente consultas.
- Editor consultas: Editor clásico de Datacycle Reporting para creación de consultas.

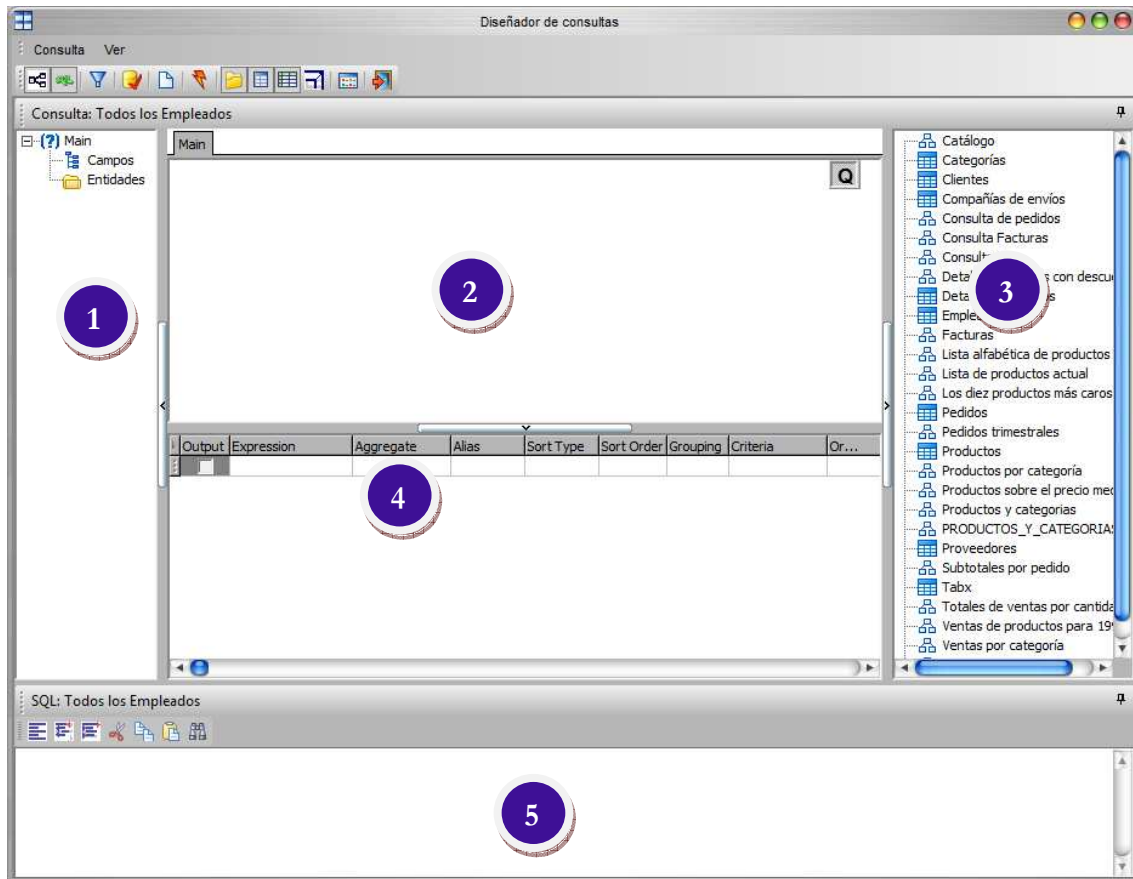
Para nuestra primera consulta accedemos al editor de consultas modo visual:



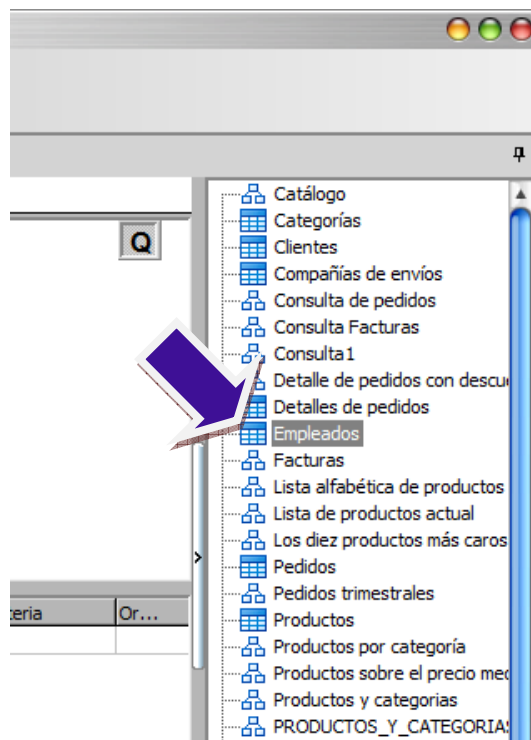
Editor de Consultas Modo Visual

El editor de consultas etiquetado como visual además de una sección de edición libre del texto SQL (5), muestra una sección con cuatro paneles gráficos que agilizan la construcción de la sentencia SQL de forma considerable:

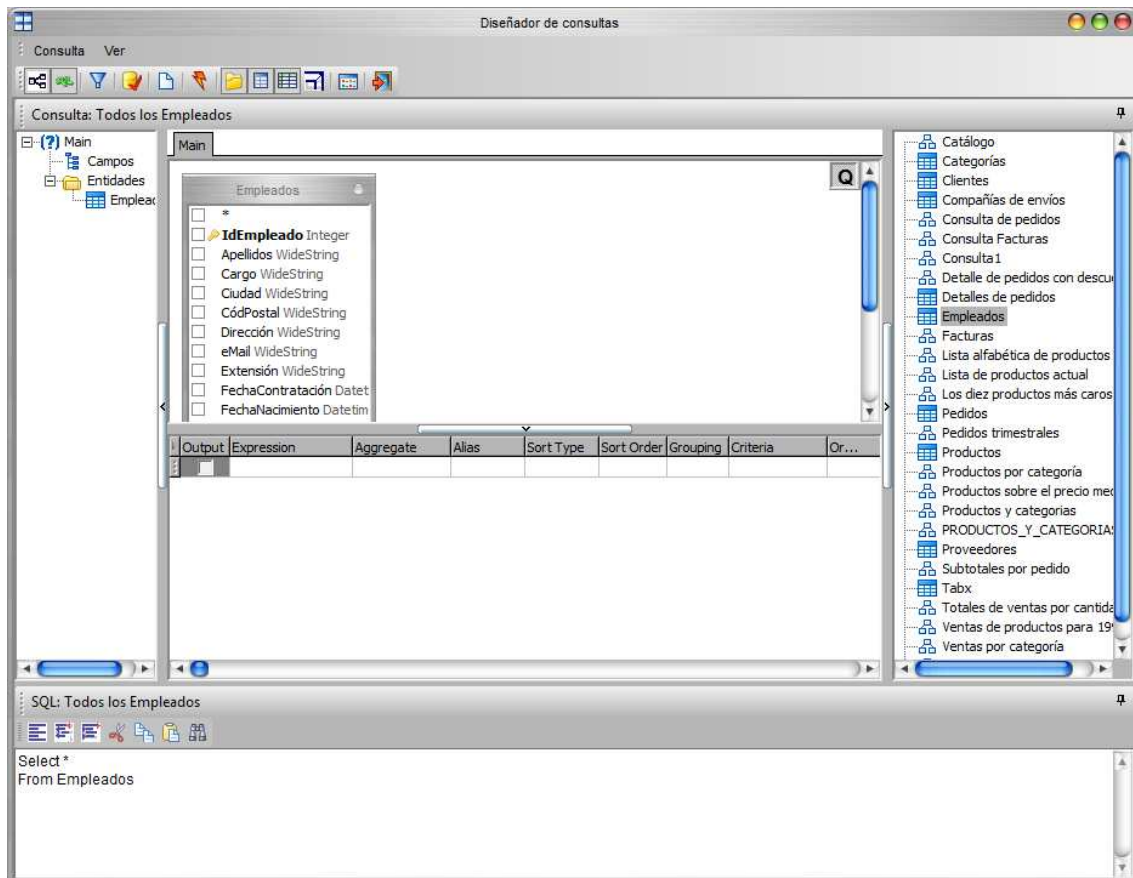
- Panel objetos de la consulta (1)
- Panel diagrama entidad-relación (2)
- Panel esquema base de datos (3)
- Panel cuadrícula de columnas (4)



Usaremos el panel de esquema base de datos para escoger aquellas tablas que deseamos consultar. En nuestro ejemplo seleccionaremos la tabla **Empleados**, haciendo doble clic con el ratón.

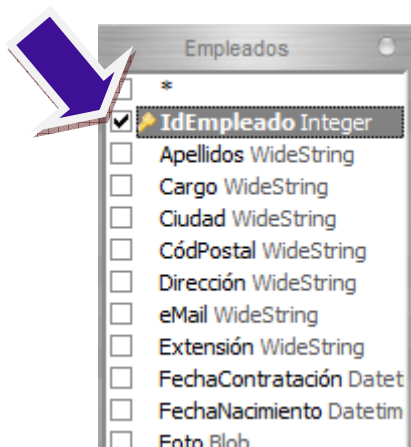


El asistente mostrará el siguiente estado:

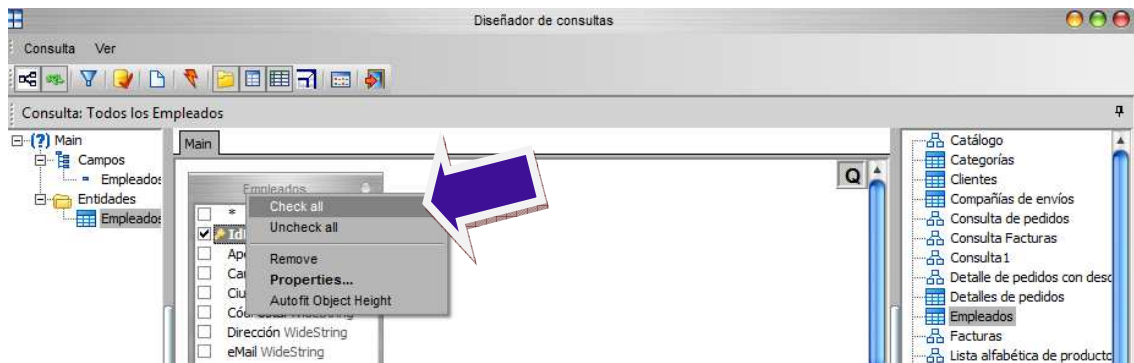


Podemos observar que en los paneles diagrama entidad-relación y objetos de la consulta se ha añadido la tabla **Empleados**. Asimismo en la sección edición libre del texto SQL se ve reflejado la acción que acabamos de realizar.

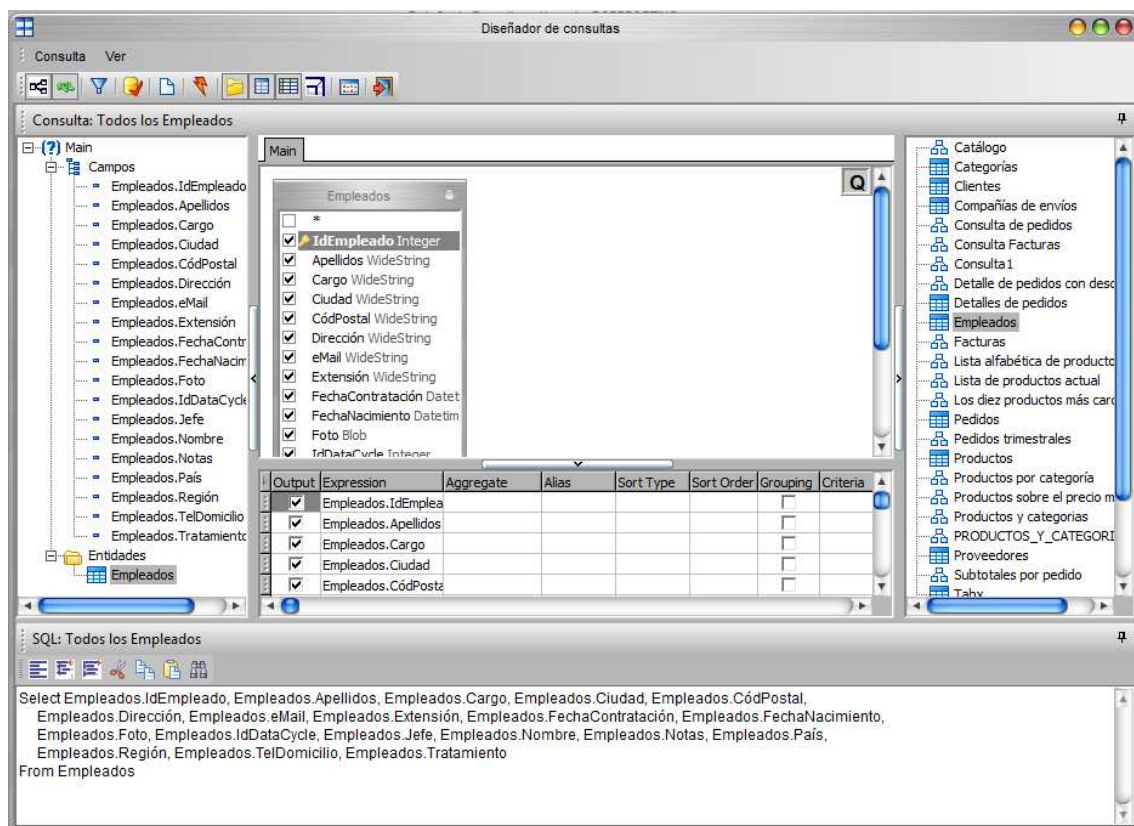
Para seleccionar los datos que queremos obtener en la consulta, haremos un clic del ratón, sobre las casillas de los campos de la tabla en el panel de diagrama.




Haciendo clic con el botón derecho del ratón, sobre la representación de la tabla en el panel de diagramas podemos seleccionar todos los campos a la vez.



El asistente mostrará el siguiente estado:



Vamos a Ejecutar la consulta para verificar que no hemos cometido ninguna equivocación. Para ello, pulsamos el botón  **Ejecutar**, y aparecerá el “Visor de Datos” de DataCycle mostrando los registros obtenidos por la consulta.

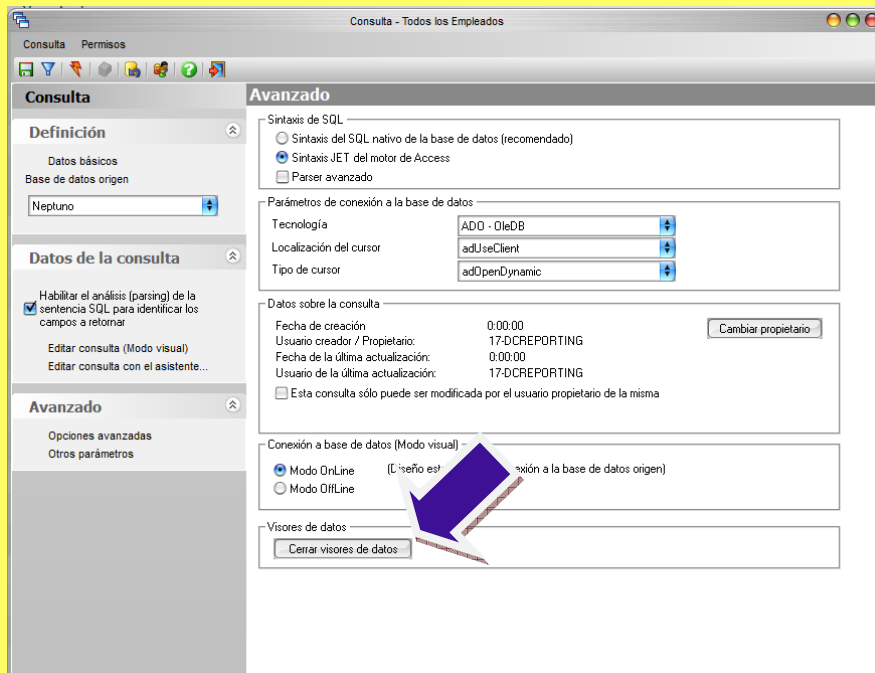
| IdEmpleado | Apellidos | Cargo | Ciudad | CódPostal | Dirección | eMail | Exter |
|------------|-----------|----------------------------|----------|-----------|-----------------------|-----------------------|-------|
| 1 | Davolio | Representante de ventas | Seattle | 98122 | 507 - 20th Ave. E. | Davolio@neptuno.com | 5467 |
| 2 | Fuller | Vicepresidente comercial | Tacoma | 98401 | 908 W. Capital Way | Fuller@neptuno.com | 3457 |
| 3 | Leverling | Representante de ventas | Kirkland | 98033 | 722 Moss Bay Blvd. | Leverling@neptuno.com | 3355 |
| 4 | Peacock | Representante de ventas | Redmond | 98052 | 4110 Old Redmond Rd. | Peacock@neptuno.com | 5176 |
| 5 | Buchanan | Gerente de ventas | Londres | SW1 8JR | 14 Garrett Hill | Buchanan@neptuno.com | 3453 |
| 6 | Suyama | Representante de ventas | Londres | EC2 7JR | Coventry House | Suyama@neptuno.com | 428 |
| 7 | King | Representante de ventas | Londres | RG1 9SP | Edgeham Hollow | King@neptuno.com | 465 |
| 8 | Callahan | Coordinador ventas interno | Seattle | 98105 | 4726 - 11th Ave. N.E. | Callahan@neptuno.com | 2344 |
| 9 | Dodsworth | Representante de ventas | Londres | WG2 7LT | 7 Houndstooth Rd. | Dodsworth@neptuno.com | 452 |
| 10 | Stevens | Jefe de Ventas | | | | Stevens@Neptuno.com | |



Registros cargados: 10 Nº registros leídos: 10 Registro actual: 0 T. conexión: 00:00:00 ImageManager Property P:

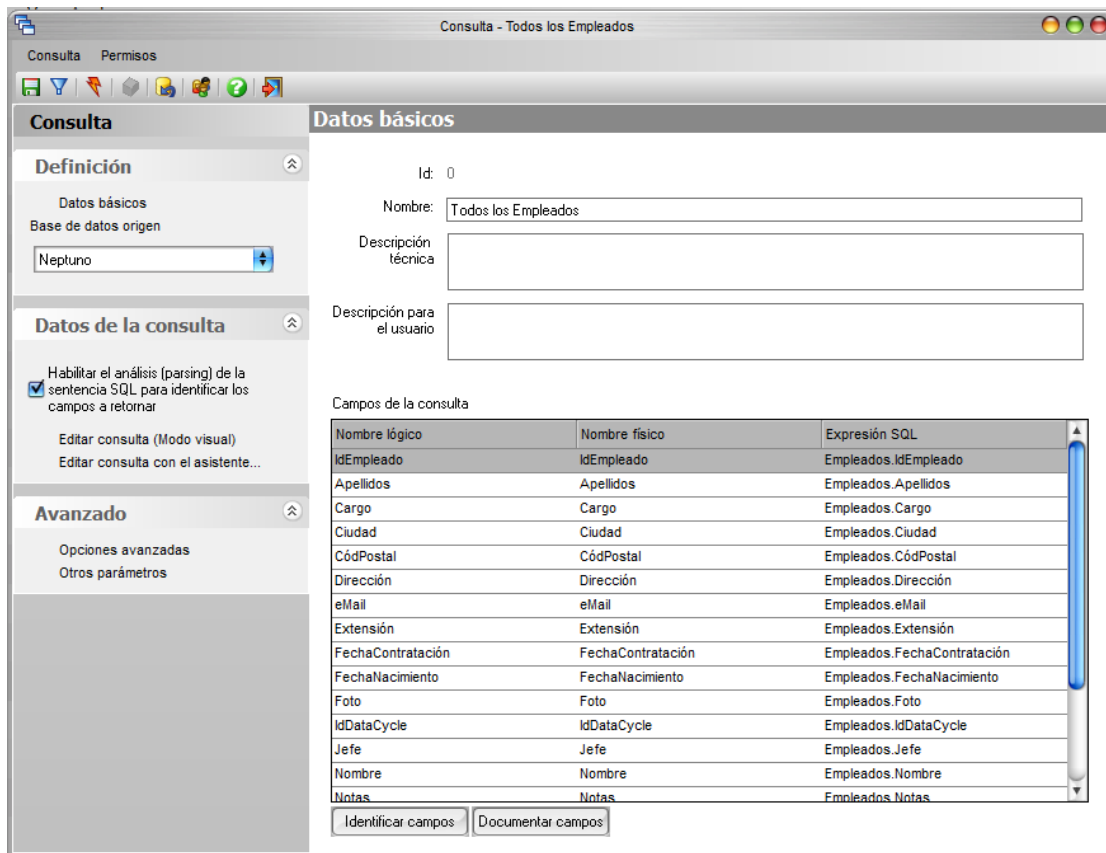
Cuando una consulta contiene muchos registros y tarda varios minutos en cargarse totalmente, aparece minimizada inicialmente y cuando ha finalizado toda la carga se maximiza automáticamente. De esta manera permite realizar otras tareas hasta que finalice la carga.



Hay consultas que tardan un tiempo elevado en retornar datos, esta ventana de visor permanecerá minimizada hasta que el gestor de BBDD resuelva la consulta. Si se desea cancelar la ejecución de cualquier consulta, se ha de ir a **Opciones avanzadas** y utilizar el botón cerrar visor de datos.






Para cerrar el Visor de Datos pulsamos sobre  (**salir**). Seguidamente regresamos a la ventana de edición de la consulta, y pulsamos .

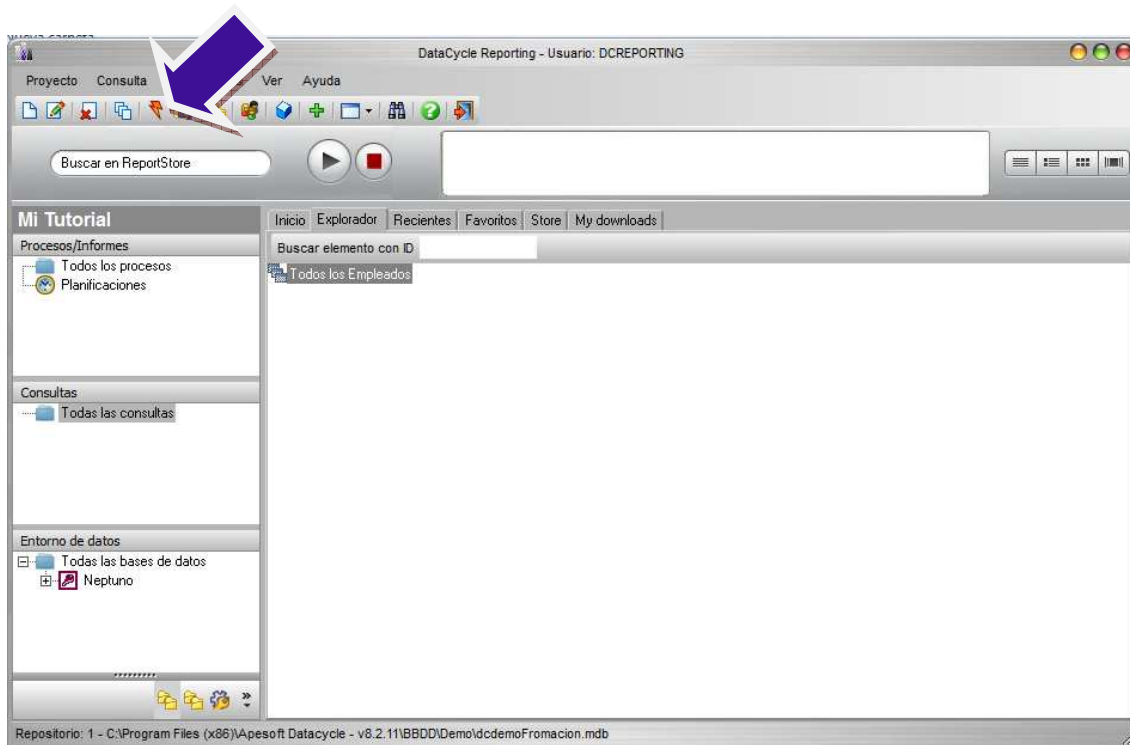



Como se puede apreciar, automáticamente se han documentado los campos a retornar por la consulta. Si lo deseamos podemos situarnos con el ratón sobre el campo, en la columna **[Nombre lógico]** para modificar la descripción correspondiente a cada columna. Esta descripción puede ser un nombre de campo más intuitivo que el que físicamente está documentado en la base de datos y que ayudará al diseñador de informes a entender mejor qué datos devuelve la consulta y en qué campos.

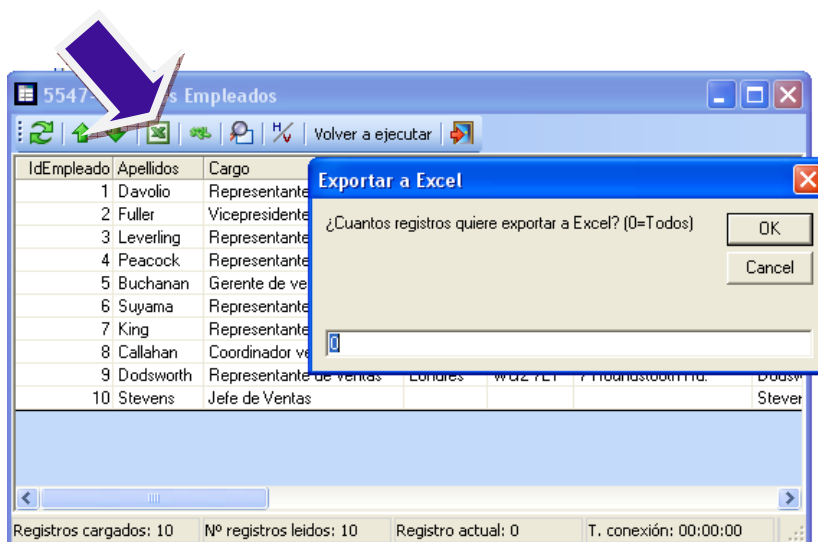
Campos de la consulta

| Nombre lógico | Nombre físico | Expresión SQL |
|-------------------|-------------------|-----------------------------|
| IdEmpleado | IdEmpleado | Empleados.IdEmpleado |
| Apellidos | Apellidos | Empleados.Apellidos |
| Cargo | Cargo | Empleados.Cargo |
| Ciudad | Ciudad | Empleados.Ciudad |
| CódPostal | CódPostal | Empleados.CódPostal |
| DirecciónEmpresa | Dirección | Empleados.Dirección |
| eMail | eMail | Empleados.eMail |
| Extensión | Extensión | Empleados.Extensión |
| FechaContratación | FechaContratación | Empleados.FechaContratación |

Finalmente pulsar sobre el botón  para salvar la consulta y sobre  para salir. En el explorador de DataCycle aparecerá nuestra primera consulta. A partir de ahora podremos ejecutarla tantas veces como deseemos sin necesidad de entrar a la ventana de edición de consultas. Simplemente hay que pulsar sobre el botón  de la barra de herramientas principal.



Con el **Visor de Datos** abierto (al ejecutar una consulta) podemos exportar los datos a Excel pulsando sobre el botón . Una ventana nos solicitará el número de registros que deseamos exportar, dejaremos el valor 0 para indicar que deseamos exportar todos los registros que devuelva la consulta.



El resultado final será la aparición de Excel con los datos ya exportados aunque sin formatear.

Libro1 - Microsoft Excel

Inicio Insertar Diseño de página Fórmulas Datos Revisar Vista Programador

Normal Diseño de página Mostrar u ocultar Zoom 100% Ampliar selección Nueva ventana Organizar todo Inmovilizar paneles Guardar área de trabajo Cambiar ventanas Macros

A1 IdEmpleado

| | A | B | C | D | E | F | G |
|----|------------|-----------|----------------------------|----------|-----------|-----------------------------------|-----------------------|
| 1 | IdEmpleado | Apellidos | Cargo | Ciudad | CódPostal | Dirección | eMail |
| 2 | 1 | Davolio | Representante de ventas | Seattle | 98122 | 507 - 20th Ave. E. Apt. 2A | Davolio@neptuno.com |
| 3 | 2 | Fuller | Vicepresidente comercial | Tacoma | 98401 | 908 W. Capital Way | Fuller@neptuno.com |
| 4 | 3 | Leverling | Representante de ventas | Kirkland | 98033 | 722 Moss Bay Blvd. | Leverling@neptuno.com |
| 5 | 4 | Peacock | Representante de ventas | Redmond | 98052 | 4110 Old Redmond Rd. | Peacock@neptuno.com |
| 6 | 5 | Buchanan | Gerente de ventas | Londres | SW1 8JR | 14 Garrett Hill Coventry House | Buchanan@neptuno.com |
| 7 | 6 | Suyama | Representante de ventas | Londres | EC2 7JR | Miner Rd. Edgeham Hollow | Suyama@neptuno.com |
| 8 | 7 | King | Representante de ventas | Londres | RG1 9SP | Winchester Way | King@neptuno.com |
| 9 | 8 | Callahan | Coordinador ventas interno | Seattle | 98105 | 4726 - 11th Ave. N.E. | Callahan@neptuno.com |
| 10 | 9 | Dodsworth | Representante de ventas | Londres | WG2 7LT | 7 Houndstooth Rd. | Dodsworth@neptuno.com |
| 11 | 10 | Stevens | Jefe de Ventas | | | | Stevens@Neptuno.com |

Hoja1 Hoja2 Hoja3

Listo Promedio: 18782,76364 Recuento: 178 Suma: 1033052 100%



Esta consulta desarrollada con el editor en modo visual, está explicada con el editor tradicional al final del documento en el Anexo **“Diseño de consultas con el editor tradicional”**.

Ejercicios

Una vez que hemos realizado la primera consulta con una sola tabla, se propone los siguientes ejercicios sobre la Base de Datos Neptuno.

Consultas simples

Consultas simples sobre una sola tabla

Las siguientes consultas trabajan seleccionando campos de una sola tabla de la Base de Datos Neptuno.

Ejemplo 01-01

En que ciudad viven mis clientes

```
SELECT Ciudad
FROM [Clientes]
```

Nota 1: También es válido el siguiente formato:

```
SELECT Clientes.Ciudad
FROM [Clientes]
```

Nota 2: Si se desea tener todos los campos disponibles puede utilizar la siguiente consulta aunque posteriormente no podrá seleccionar un campo concreto para su informe:

```
SELECT *
FROM [Clientes]
```

Ejemplo 01-02

Quiero conocer el nombre y dirección de todos mis empleados

```
SELECT Empleados.Nombre, Empleados.Apellidos, Empleados.Dirección, Empleados.Ciudad,
Empleados.Región, Empleados.CódPostal, Empleados.País
FROM [Empleados]
```

Ejemplo 01-03

Que categorías ofrecemos actualmente

```
SELECT Categorías.Descripción
FROM [Categorías]
```

Ejemplo 01-04

Cuáles son los productos y precios y a que categorías pertenecen

```
SELECT Productos.NombreProducto, Productos.PrecioUnidad, Productos.IdCategoría
FROM [Productos]
```

Nota: tan solo disponemos del identificador de categoría y no de su nombre.

Eliminar duplicados

Ahora veremos como evitar la aparición de información repetida.

Ejemplo 01-05

En que ciudades tengo algún cliente

```
SELECT DISTINCT Clientes.Ciudad  
FROM [Clientes]
```

Nota: En este caso, y para evitar confusiones entre ciudades diferentes pero con el mismo nombre, tiene más sentido la siguiente consulta:

```
SELECT DISTINCT Clientes.País, Clientes.Región, Clientes.Ciudad, Clientes.CódPostal  
FROM [Clientes]
```

Ordenando la información

Veremos ahora como solicitar información comercial ordenada por diferentes campos.

Ejemplo 01-06

Listado de nombres de nuestros empleados, incluyendo su número de teléfono e identificación, y mostrados ordenados por apellido y nombre

```
SELECT Empleados.Apellidos, Empleados.Nombre, Empleados.TelDomicilio, Empleados.IdEmpleado  
FROM [Empleados]  
Order By Empleados.Apellidos, Empleados.Nombre
```

Ejemplo 01-07

Listado de precios de nuestros productos mostrados de más caro a más barato

```
SELECT Productos.NombreProducto, Productos.PrecioUnidad  
FROM [Productos]  
Order By Productos.PrecioUnidad desc
```

Nota: para ver los 4 productos más caros:

```
SELECT top 4 Productos.NombreProducto, Productos.PrecioUnidad  
FROM [Productos]  
Order By Productos.PrecioUnidad desc
```

Selecciones con operaciones simples

Ahora vamos a pedir información que implique la realización de operaciones con los campos existentes.

Ejemplo 01-08

Cual es el valor de inventario de cada producto

```
SELECT Productos.IdProducto, Productos.NombreProducto,  
       Productos.PrecioUnidad * Productos.UnidadesEnExistencia as ValorDeInventario  
FROM [Productos]
```

Ejemplo 01-09

Cuales son los pedidos con peor tiempo de entrega

```
SELECT Pedidos.IdPedido, Pedidos.FechaPedido, Pedidos.FechaEntrega,  
       Pedidos.FechaEntrega - Pedidos.FechaPedido as DiasParaEntrega  
FROM [Pedidos]  
order by Pedidos.FechaEntrega - Pedidos.FechaPedido desc
```

Utilización de funciones

Ejemplo 01-10

Inicial del nombre y apellido de mis empleados

```
SELECT Left(Empleados.Nombre,1) & ' ' & Empleados.Apellidos  
FROM [Empleados]
```

Filtrando los datos

Vamos a introducir el concepto de criterios en la solicitud de información y utilización de parámetros.

Ejemplo 02-01

Quiero ver la fecha de pedido y entrega de los pedidos del cliente ANTON

```
SELECT Pedidos.IdPedido, Pedidos.FechaPedido, Pedidos.FechaEntrega
FROM [Pedidos]
WHERE Pedidos.IdCliente='ANTON'
Order By Pedidos.FechaPedido
```

Nota1: Para aprovechar la consulta en otros informes, se recomienda seleccionar todos los campos:

```
SELECT Pedidos.IdPedido, Pedidos.IdCliente, Pedidos.IdEmpleado, Pedidos.FechaPedido,
Pedidos.FechaEntrega, Pedidos.FechaEnvío, Pedidos.FormaEnvío, Pedidos.Cargo,
Pedidos.Destinatarario, Pedidos.DirecciónDestinatario, Pedidos.CiudadDestinatario,
Pedidos.RegiónDestinatario, Pedidos.CódPostalDestinatario, Pedidos.PaísDestinatario
FROM [Pedidos]
WHERE Pedidos.IdCliente='ANTON'
Order By Pedidos.FechaPedido
```

Nota2: Para hacer más portable la consulta, se recomienda la utilización de parámetros (explicados mas adelante):

```
SELECT
Pedidos.IdPedido, Pedidos.IdCliente, Pedidos.IdEmpleado, Pedidos.FechaPedido,
Pedidos.FechaEntrega, Pedidos.FechaEnvío, Pedidos.FormaEnvío, Pedidos.Cargo,
Pedidos.Destinatarario, Pedidos.DirecciónDestinatario, Pedidos.CiudadDestinatario,
Pedidos.RegiónDestinatario, Pedidos.CódPostalDestinatario, Pedidos.PaísDestinatario
FROM [Pedidos]
WHERE Pedidos.IdCliente='@PAR(Cliente)'
Order By Pedidos.FechaPedido
```

Nota3: Para hacer aún más portable la consulta, se recomienda no poner criterios de negocio en la consulta y en su lugar añadir un filtro en el proceso. El resultado es:

```
SELECT
Pedidos.IdPedido, Pedidos.IdCliente, Pedidos.IdEmpleado, Pedidos.FechaPedido,
Pedidos.FechaEntrega, Pedidos.FechaEnvío, Pedidos.FormaEnvío, Pedidos.Cargo,
Pedidos.Destinatarario, Pedidos.DirecciónDestinatario, Pedidos.CiudadDestinatario,
Pedidos.RegiónDestinatario, Pedidos.CódPostalDestinatario, Pedidos.PaísDestinatario
FROM [Pedidos]
Order By Pedidos.FechaPedido
```

Y añadir el siguiente filtro:

```
Pedidos.IdCliente='@PAR(Cliente)'
```


Ejemplo 02-02

Quiero ver la lista en orden alfabético de nuestros productos que contengan 'pan'

```
SELECT Productos.NombreProducto
FROM [Productos]
WHERE Productos.NombreProducto like '%pan%'
Order by Productos.NombreProducto
```

Nota: Recuerde la utilización de parámetros:

```
SELECT Productos.NombreProducto
FROM [Productos]
WHERE Productos.NombreProducto like '%@PAR(Producto)%'
Order by Productos.NombreProducto
```

Ejemplo 02-03

Lista de los proveedores de Italia o España

```
SELECT Proveedores.NombreCompañía, Proveedores.País, Proveedores.Ciudad
FROM [Proveedores]
WHERE Proveedores.País in ('Italia', 'España')
```

Nota: si quisiéramos la lista de los proveedores que no son de Italia o España:

```
SELECT Proveedores.NombreCompañía, Proveedores.País, Proveedores.Ciudad
FROM [Proveedores]
WHERE Proveedores.País not in ('Italia', 'España')
```

Ejemplo 02-04

Quiero saber que productos tienen un precio entre 80 y 300 dólares

```
SELECT Productos.IdProducto, Productos.NombreProducto, Productos.PrecioUnidad
FROM [Productos]
WHERE Productos.PrecioUnidad between 80 and 300
```

Ejemplo 02-05

Pedidos cuyo plazo de entrega superan las 2 semanas

```
SELECT Pedidos.IdPedido, Pedidos.FechaPedido, Pedidos.FechaEntrega
FROM [Pedidos]
WHERE Pedidos.FechaEntrega - Pedidos.FechaPedido >= 14
```

Ejemplo 02-06

Productos de la categoría 2, 4 o 8 o cuyo precio sea igual o superior a 100\$

```
SELECT Productos.NombreProducto, Productos.IdCategoría, Productos.PrecioUnidad
FROM [Productos]
WHERE Productos.IdCategoría in (2, 4, 8) OR Productos.PrecioUnidad >= 100
```

Nota: Compliquemos un poco la consulta: Necesito los productos de la categoría 2, 4 o 8 con precio superior a 40 o los productos con precio superior a 100 independientemente de su categoría.

```
SELECT Productos.NombreProducto, Productos.IdCategoría, Productos.PrecioUnidad
FROM [Productos]
WHERE (Productos.IdCategoría in (2, 4, 8) AND Productos.PrecioUnidad >= 40) OR Productos.PrecioUnidad >= 100
```

Consultas con varias tablas


Vamos a realizar una consulta en donde buscamos información de varias tablas al mismo tiempo.

El objetivo para este capítulo va a ser crear una consulta de los pedidos e información relacionada. Esta consulta nos permitirá conocer el sistema a seguir para la selección de información desde varias tablas a la vez.

Detalle de ventas

Queremos que la consulta devuelva una estructura como la siguiente:

| Cliente | País | Representante | Categoría | Producto | Año | Mes | Ventas | Unidades |
|---------|------|---------------|-----------|----------|-----|-----|--------|----------|
|---------|------|---------------|-----------|----------|-----|-----|--------|----------|

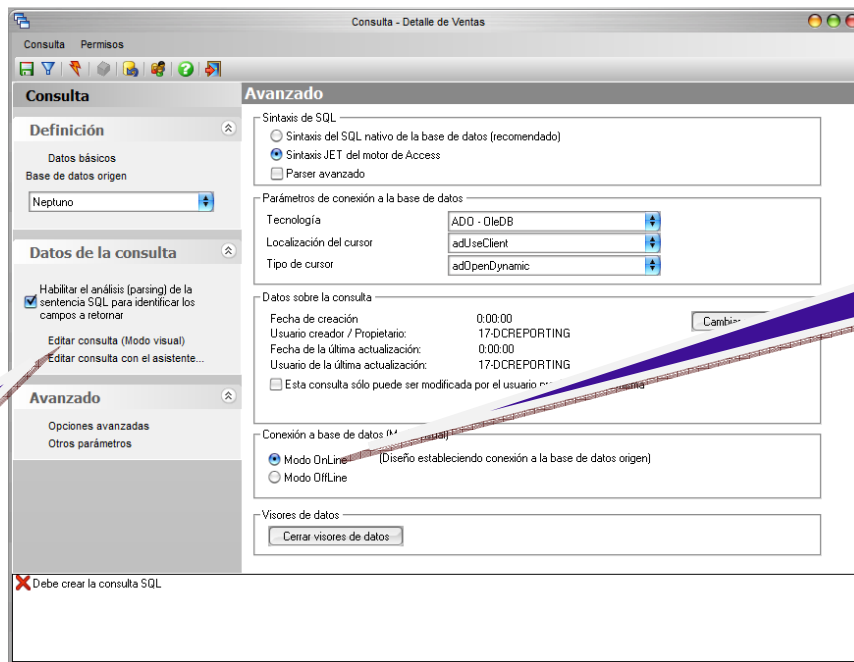
Nos situaremos en la sección de consultas y pulsar  sobre el botón de la barra de herramientas. Aparecerá la ventana de selección de bases de datos, seleccionaremos Neptuno. Aparecerá la ventana vacía de edición de la consulta.

1

Le asignamos el nombre:
Detalle de Ventas

2

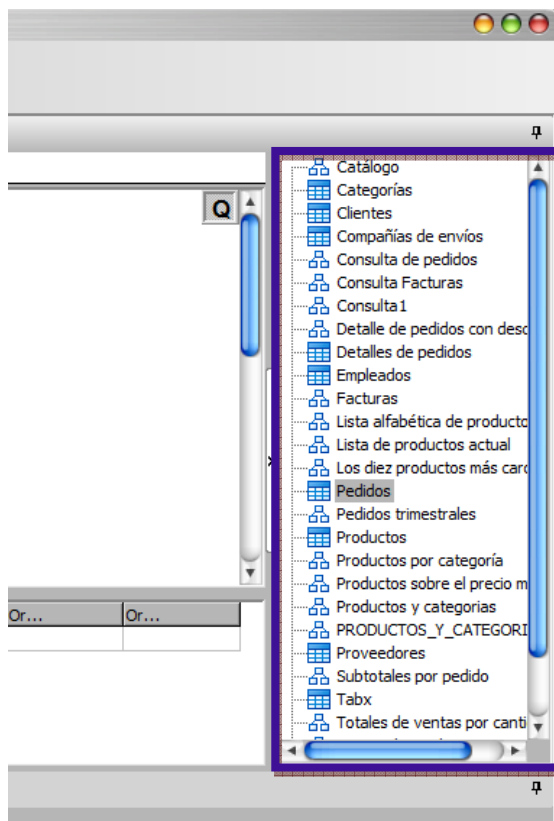
Vamos a las
Opciones avanzadas



4
 Editar Consulta
 (Modo Visual)

3
 Comprobamos que
 está seleccionado el
 Modo **On-Line**

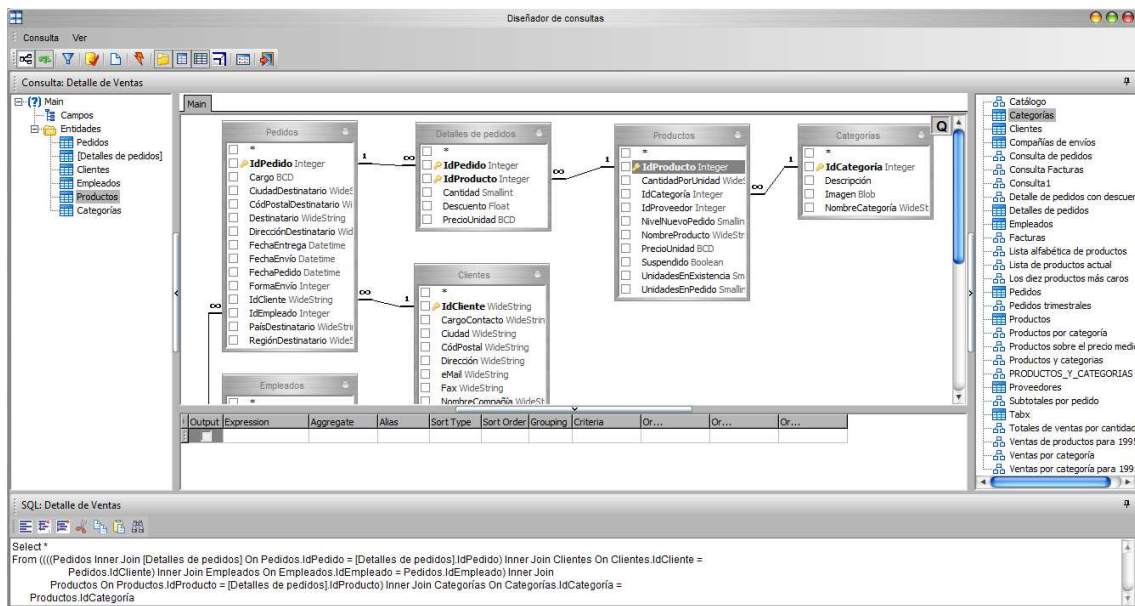
En el panel esquema base de datos iremos seleccionando las tablas que necesitamos para obtener los datos deseados en la consulta.



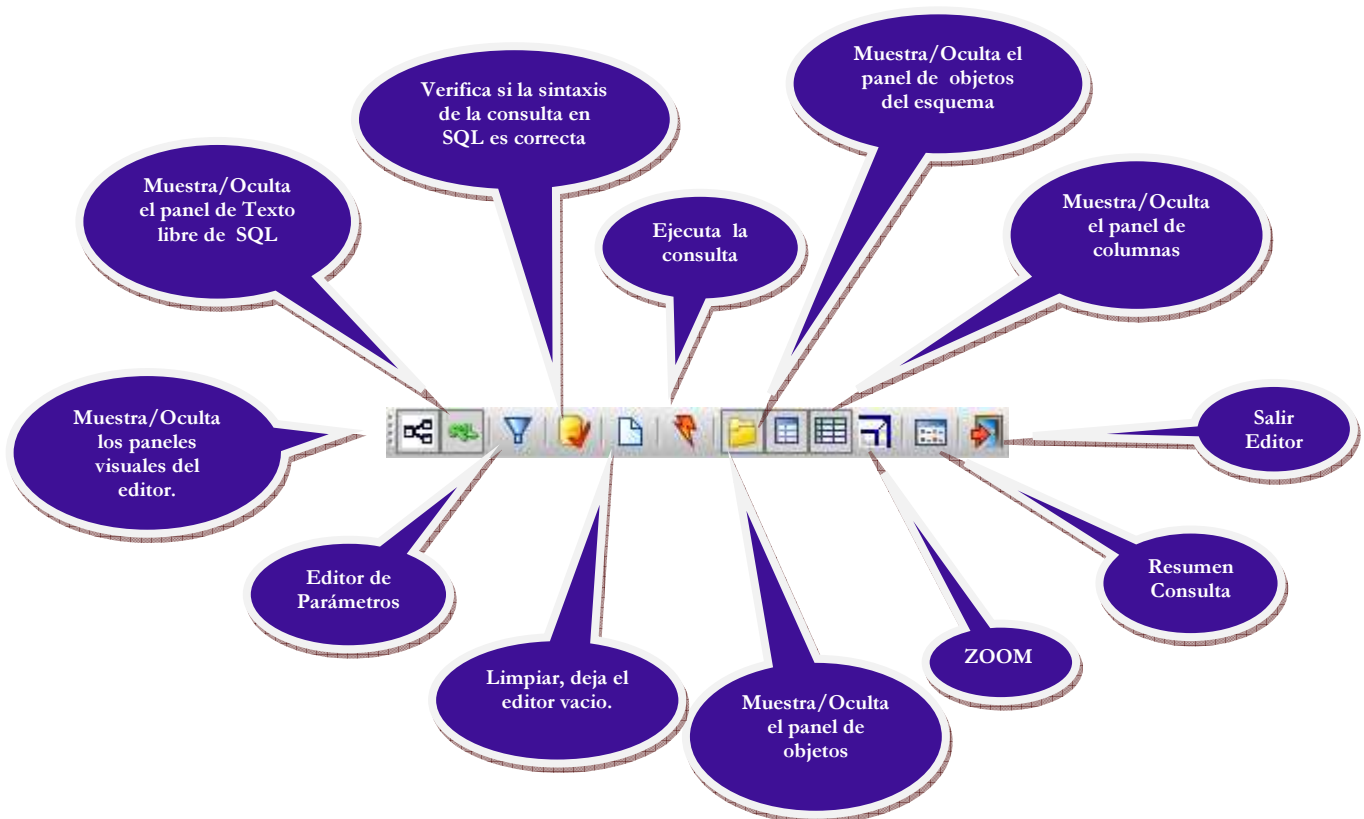
En nuestro caso seleccionaremos las tablas:

- Pedidos
- Detalles de pedidos
- Clientes
- Empleados
- Productos
- Categorías

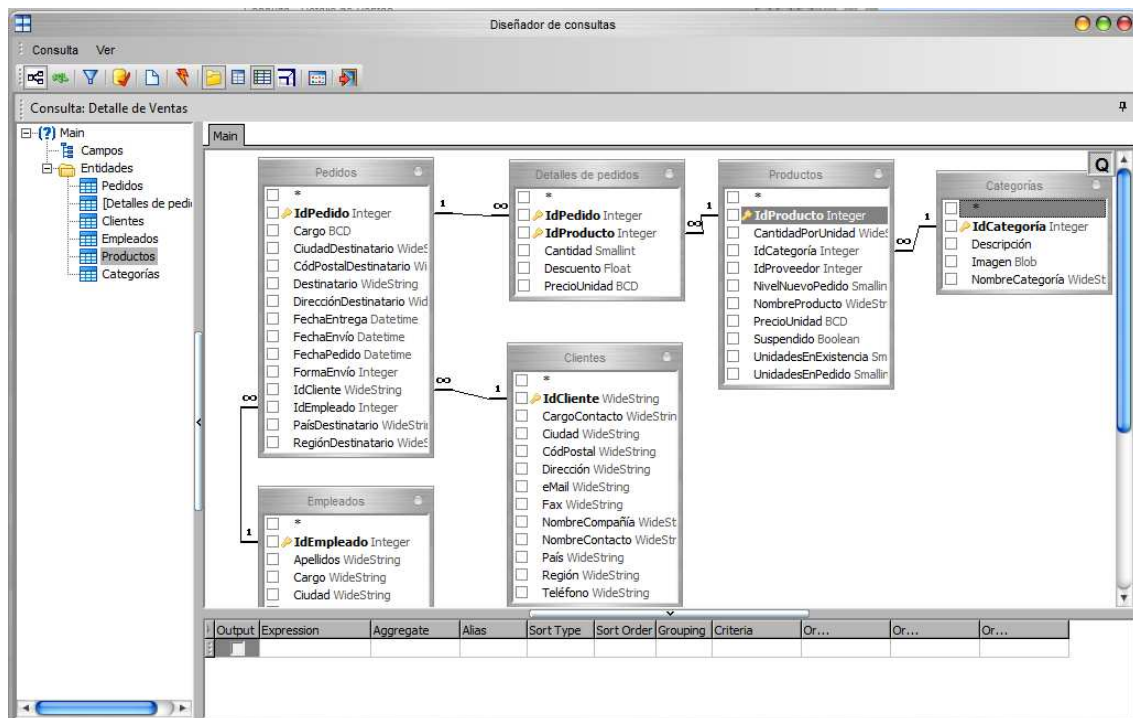
La ventana del diseñador de consultas quedará así:



Vamos a ver con más detalle lo que nos muestran los distintos paneles. Utilizaremos los botones de la barra de herramienta para tener una visión deseada de las diferentes secciones.



Ocultemo el panel de **objetos de esquema** y la sección de **SQL libre**, la visión será la siguiente:



El esquema presenta las tablas seleccionadas con las relaciones que hay entre las diferentes entidades. El editor puede representar las relaciones porque la BBDD Neptuno tiene definidas las relaciones entre entidades y la consulta está en modo **On-Line**.

Podemos ver que la entidad **Cliente** se relaciona con la entidad **Pedidos** por el campo **IdCliente**, que a su vez la entidad **Pedidos** se relaciona con la entidad **Detalle de pedidos** mediante el campo **IdPedido**...

Si observamos el panel de columnas vemos que no hay ningún campo seleccionado, vamos a seleccionar los campos que necesitamos. Recordemos que la consulta ha de contener los siguientes campos:

| | | | | | | | | |
|---------|------|---------------|-----------|----------|-----|-----|--------|----------|
| Cliente | País | Representante | Categoría | Producto | Año | Mes | Ventas | Unidades |
|---------|------|---------------|-----------|----------|-----|-----|--------|----------|

Los campos **Cliente** y **País** los obtenemos de la tabla **Cientes**, activamos las casillas de **NombreCompañía (Cliente)** y **País**.

Al seleccionar estos dos campos, podemos ver que el panel de columnas tenemos dos registros, uno por cada campo seleccionado:

| | Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria | Or... | Or... | Or... |
|--------|-------------------------------------|-------------------|-----------|-------|-----------|------------|--------------------------|----------|-------|-------|-------|
| Output | <input checked="" type="checkbox"/> | Cientes.NombreCom | | | | | <input type="checkbox"/> | | | | |
| Output | <input checked="" type="checkbox"/> | Cientes.Pais | | | | | <input type="checkbox"/> | | | | |
| Output | <input type="checkbox"/> | | | | | | | | | | |

Vamos a crear un alias para el campo **NombreCompañía**, que nosotros queremos que se llame **Cliente**. Vamos a la columna de Alias del registro que queremos crear el alias y creamos el alias.

| | Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria | Or... | Or... | Or... |
|--------|-------------------------------------|-------------------|-----------|---------|-----------|------------|--------------------------|----------|-------|-------|-------|
| Output | <input checked="" type="checkbox"/> | Cientes.NombreCom | | Cliente | | | <input type="checkbox"/> | | | | |
| Output | <input checked="" type="checkbox"/> | Cientes.Pais | | | | | <input type="checkbox"/> | | | | |
| Output | <input type="checkbox"/> | | | | | | | | | | |

El siguiente campo a poner es **Representante**, es decir el nombre del Empleado, que se encuentra en la tabla **Empleados**. En la tabla **Empleados** activamos la casilla del campo **Nombre** y creamos el alias **Representante**.

Tenemos que hacer la misma operación para añadir la **Categoría** que se encuentra en la tabla **Categorías** y es el campo **NombreCategoría**, y lo mismo para el **Producto** que se encuentra en la tabla de **Productos** y es el campo **NombreProducto**.

La consulta presenta el siguiente aspecto:

| Salida | Expresión | Agregado | Alias | Tipo de orde | Orden | Agrupado | Grupo | Orden | Orden | Orden |
|-------------------------------------|----------------------------|----------|-----------|--------------|-------|--------------------------|-------|-------|-------|-------|
| <input checked="" type="checkbox"/> | Cientes.NombreCompañía | | Cliente | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Cientes.Pais | | | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | Producto | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | Categoría | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Vendedor | | | <input type="checkbox"/> | | | | |

Ahora falta el **Mes** y el **Año**. Los sacaremos de la fecha del pedido. Añadimos **2 veces** el campo **FechaPedido** de la tabla **Pedidos**.

Para añadir una segunda vez un campo ya seleccionado simplemente tenemos que hacer un doble clic con el ratón el campo que deseamos añadir.

El panel de columnas presenta el siguiente aspecto:

| Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria |
|-------------------------------------|----------------------------|-----------|---------------|-----------|------------|--------------------------|----------|
| <input checked="" type="checkbox"/> | Cientes.NombreCompañía | | Cliente | | | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | Cientes.País | | | | | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Representante | | | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | Categoría | | | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | Producto | | | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | Pedidos.FechaPedido | | | | | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | Pedidos.FechaPedido | | FechaPedido1 | | | <input type="checkbox"/> | |

Hemos añadido dos veces la fecha de pedido por que el mes lo sacaremos recogiendo el mes de la fecha del pedido con ayuda de una función, y con el año haremos lo mismo.

Hacemos un clic sobre el primer campo **Pedidos.FechaPedido** de la lista del Panel de columnas en la casilla de expresión y nos permite modificarlo:

| | | | |
|-------------------------------------|---------------------|---|----------|
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Empleado |
| <input checked="" type="checkbox"/> | Pedidos.FechaPedido | ▼ | |
| <input checked="" type="checkbox"/> | Pedidos.FechaPedido | | |

Modificamos el texto para que sea: **Month(Pedidos.FechaPedido)** , y como alias le ponemos Mes.

Y haremos algo parecido con el segundo **Pedidos.FechaPedido**. Hacemos un clic sobre **Pedidos.FechaPedido** y cambiamos el texto para que sea: **Year(Pedidos.FechaPedido)** y como alias le ponemos Año.

El panel de columnas presenta el siguiente aspecto:

| Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria | Or... | Or... |
|-------------------------------------|----------------------------|-----------|---------------|-----------|------------|--------------------------|----------|-------|-------|
| <input checked="" type="checkbox"/> | Cientes.NombreCompañía | | Cliente | | | <input type="checkbox"/> | | | |
| <input checked="" type="checkbox"/> | Cientes.País | | | | | <input type="checkbox"/> | | | |
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Representante | | | <input type="checkbox"/> | | | |
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | Categoría | | | <input type="checkbox"/> | | | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | Producto | | | <input type="checkbox"/> | | | |
| <input checked="" type="checkbox"/> | Year(Pedidos.FechaPedido) | | Año | | | <input type="checkbox"/> | | | |
| <input checked="" type="checkbox"/> | Month(Pedidos.FechaPedido) | | Mes | | | <input type="checkbox"/> | | | |

El siguiente campo que debemos añadir es el **Importe**, es el resultado de multiplicar los campos **Cantidad** y **PrecioUnidad** de la tabla **Detalles de Pedidos**.

Primero añadimos el campo **Cantidad** de la tabla **Detalles de pedidos**, hacemos clic encima del campo en la lista del panel de columnas y modificamos el texto para que sea: **[Detalles de pedidos].Cantidad * [Detalles de pedidos].PrecioUnidad**, como alias del campo le pondremos **Ventas**.

| Output | Expression | Aggregate | Alias | Sort Typ |
|-------------------------------------|---|-----------|--------|----------|
| <input checked="" type="checkbox"/> | [Detalles de pedidos].Cantidad * Productos.PrecioUnidad | | Ventas | |

Y por ultimo añadiremos el campo **Cantidad** de la tabla **Detalles de pedidos** con el alias **Unidades**.


El panel de columnas presenta el siguiente aspecto:

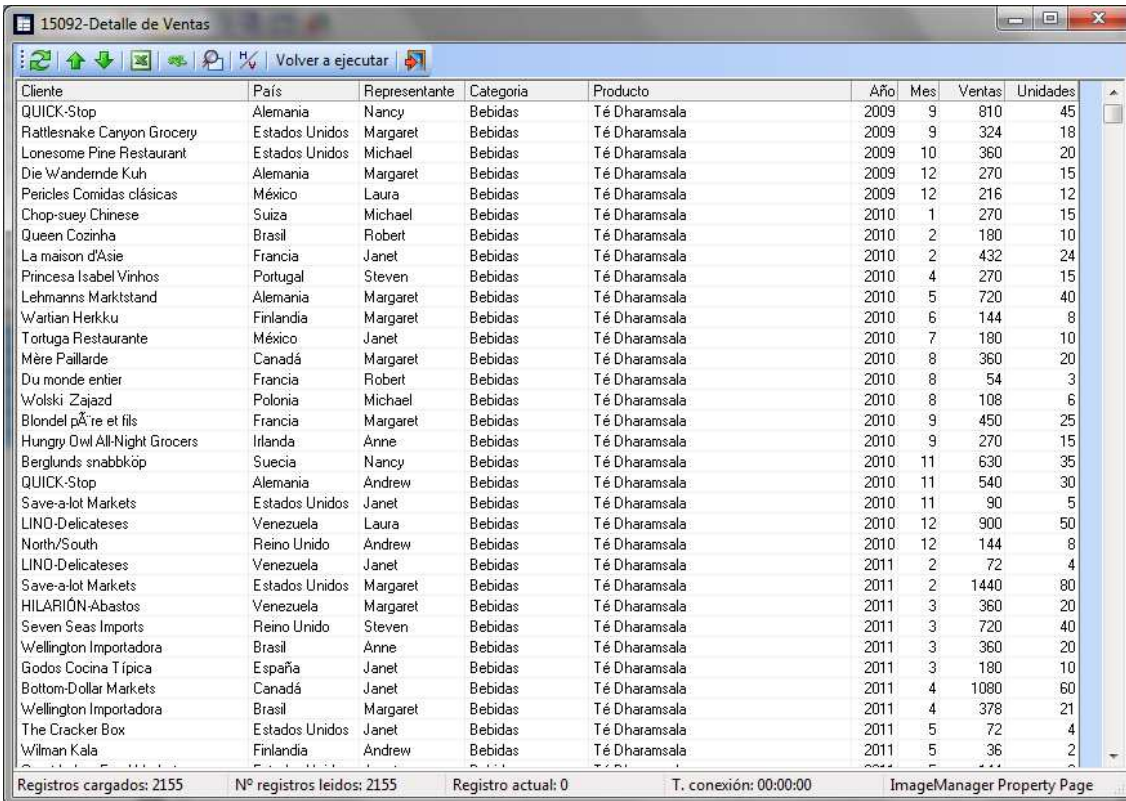
| Output | Expression | Aggregate | Alias | Sort Typ |
|-------------------------------------|---|-----------|---------------|----------|
| <input checked="" type="checkbox"/> | Cientes.NombreCompañía | | Cliente | |
| <input checked="" type="checkbox"/> | Cientes.País | | | |
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Representante | |
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | Categoría | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | Producto | |
| <input checked="" type="checkbox"/> | Year(Pedidos.FechaPedido) | | Año | |
| <input checked="" type="checkbox"/> | Month(Pedidos.FechaPedido) | | Mes | |
| <input checked="" type="checkbox"/> | [Detalles de pedidos].Cantidad * Productos.PrecioUnidad | | Ventas | |
| <input checked="" type="checkbox"/> | [Detalles de pedidos].Cantidad | | Unidades | |

Y el editor de consulta presenta el siguiente aspecto:

The screenshot shows the 'Diseñador de consultas' (Query Designer) window. The main query is named 'Consulta: Detalle de Ventas'. The query grid at the bottom is as follows:

| Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria | Or... | Or... | Or... |
|-------------------------------------|---|-----------|---------------|-----------|------------|--------------------------|----------|-------|-------|-------|
| <input checked="" type="checkbox"/> | Cientes.NombreCompañía | | Cliente | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Cientes.País | | | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Representante | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | Categoría | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | Producto | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Year(Pedidos.FechaPedido) | | Año | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Month(Pedidos.FechaPedido) | | Mes | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | [Detalles de pedidos].Cantidad * Productos.PrecioUnidad | | Ventas | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | [Detalles de pedidos].Cantidad | | Unidades | | | <input type="checkbox"/> | | | | |

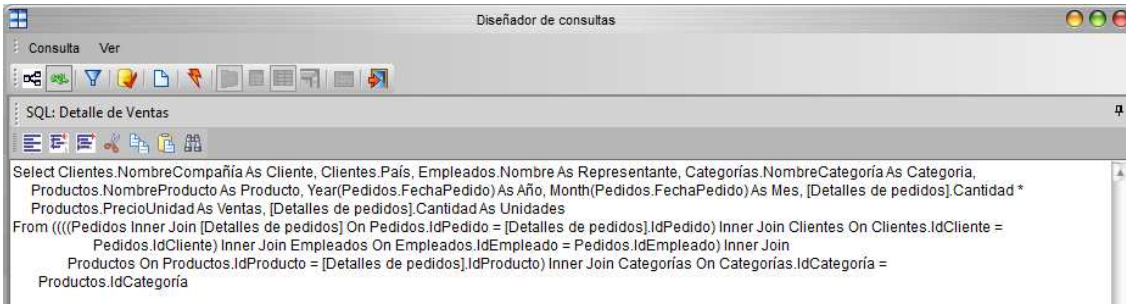
La consulta está terminada, por lo que ya podemos ejecutarla haciendo clic en . El resultado de ejecutar nuestra consulta es:



| Cliente | País | Representante | Categoría | Producto | Año | Mes | Ventas | Unidades |
|------------------------------|----------------|---------------|-----------|---------------|------|-----|--------|----------|
| QUICK-Stop | Alemania | Nancy | Bebidas | Té Dharamsala | 2009 | 9 | 810 | 45 |
| Rattlesnake Canyon Grocery | Estados Unidos | Margaret | Bebidas | Té Dharamsala | 2009 | 9 | 324 | 18 |
| Lonesome Pine Restaurant | Estados Unidos | Michael | Bebidas | Té Dharamsala | 2009 | 10 | 360 | 20 |
| Die Wandemde Kuh | Alemania | Margaret | Bebidas | Té Dharamsala | 2009 | 12 | 270 | 15 |
| Pericles Comidas clásicas | México | Laura | Bebidas | Té Dharamsala | 2009 | 12 | 216 | 12 |
| Chop-suey Chinese | Suiza | Michael | Bebidas | Té Dharamsala | 2010 | 1 | 270 | 15 |
| Queen Cozinha | Brasil | Robert | Bebidas | Té Dharamsala | 2010 | 2 | 180 | 10 |
| La maison d'Asie | Francia | Janet | Bebidas | Té Dharamsala | 2010 | 2 | 432 | 24 |
| Princesa Isabel Vinhos | Portugal | Steven | Bebidas | Té Dharamsala | 2010 | 4 | 270 | 15 |
| Lehmanns Marktstand | Alemania | Margaret | Bebidas | Té Dharamsala | 2010 | 5 | 720 | 40 |
| Wartian Herkku | Finlandia | Margaret | Bebidas | Té Dharamsala | 2010 | 6 | 144 | 8 |
| Tortuga Restaurante | México | Janet | Bebidas | Té Dharamsala | 2010 | 7 | 180 | 10 |
| Mère Paillearde | Canadá | Margaret | Bebidas | Té Dharamsala | 2010 | 8 | 360 | 20 |
| Du monde entier | Francia | Robert | Bebidas | Té Dharamsala | 2010 | 8 | 54 | 3 |
| Wolski Zajazd | Polonia | Michael | Bebidas | Té Dharamsala | 2010 | 8 | 108 | 6 |
| Blondel père et fils | Francia | Margaret | Bebidas | Té Dharamsala | 2010 | 9 | 450 | 25 |
| Hungry Owl All-Night Grocers | Irlanda | Anne | Bebidas | Té Dharamsala | 2010 | 9 | 270 | 15 |
| Berglunds snabbköp | Suecia | Nancy | Bebidas | Té Dharamsala | 2010 | 11 | 630 | 35 |
| QUICK-Stop | Alemania | Andrew | Bebidas | Té Dharamsala | 2010 | 11 | 540 | 30 |
| Save-a-lot Markets | Estados Unidos | Janet | Bebidas | Té Dharamsala | 2010 | 11 | 90 | 5 |
| LIND-Delicatesses | Venezuela | Laura | Bebidas | Té Dharamsala | 2010 | 12 | 900 | 50 |
| North/South | Reino Unido | Andrew | Bebidas | Té Dharamsala | 2010 | 12 | 144 | 8 |
| LIND-Delicatesses | Venezuela | Janet | Bebidas | Té Dharamsala | 2011 | 2 | 72 | 4 |
| Save-a-lot Markets | Estados Unidos | Margaret | Bebidas | Té Dharamsala | 2011 | 2 | 1440 | 80 |
| HILARIÓN-Abastos | Venezuela | Margaret | Bebidas | Té Dharamsala | 2011 | 3 | 360 | 20 |
| Seven Seas Imports | Reino Unido | Steven | Bebidas | Té Dharamsala | 2011 | 3 | 720 | 40 |
| Wellington Importadora | Brasil | Anne | Bebidas | Té Dharamsala | 2011 | 3 | 360 | 20 |
| Godos Cocina Típica | España | Janet | Bebidas | Té Dharamsala | 2011 | 3 | 180 | 10 |
| Bottom-Dollar Markets | Canadá | Janet | Bebidas | Té Dharamsala | 2011 | 4 | 1080 | 60 |
| Wellington Importadora | Brasil | Margaret | Bebidas | Té Dharamsala | 2011 | 4 | 378 | 21 |
| The Cracker Box | Estados Unidos | Janet | Bebidas | Té Dharamsala | 2011 | 5 | 72 | 4 |
| Wilman Kala | Finlandia | Andrew | Bebidas | Té Dharamsala | 2011 | 5 | 36 | 2 |

Cerramos la ventana de datos con el icono  de la barra de herramientas.



En el editor de consultas activamos el panel de texto SQL libre y desactivamos los paneles visuales, presentando el siguiente aspecto:



```

Select Clientes.NombreCompañía As Cliente, Clientes.País, Empleados.Nombre As Representante, Categorías.NombreCategoría As Categoría,
  Productos.NombreProducto As Producto, Year(Pedidos.FechaPedido) As Año, Month(Pedidos.FechaPedido) As Mes, [Detalles de pedidos].Cantidad *
  Productos.PrecioUnidad As Ventas, [Detalles de pedidos].Cantidad As Unidades
From (((Pedidos Inner Join [Detalles de pedidos] On Pedidos.IdPedido = [Detalles de pedidos].IdPedido) Inner Join Clientes On Clientes.IdCliente =
  Pedidos.IdCliente) Inner Join Empleados On Empleados.IdEmpleado = Pedidos.IdEmpleado) Inner Join
  Productos On Productos.IdProducto = [Detalles de pedidos].IdProducto) Inner Join Categorías On Categorías.IdCategoría =
  Productos.IdCategoría
  
```

En este panel podemos ver el código SQL que se ha generado a partir de las acciones que hemos realizado en los paneles visuales. Se puede utilizar este panel para generar directamente el código SQL sin necesidad de utilizar los paneles visuales, escribiendo el código SQL directamente en el panel. Las modificaciones que se realicen en este editor quedarán reflejadas en los paneles visuales.

Finalmente, pulsaremos el botón  para grabar la consulta y sobre  para salir de la ventana de edición de consulta.



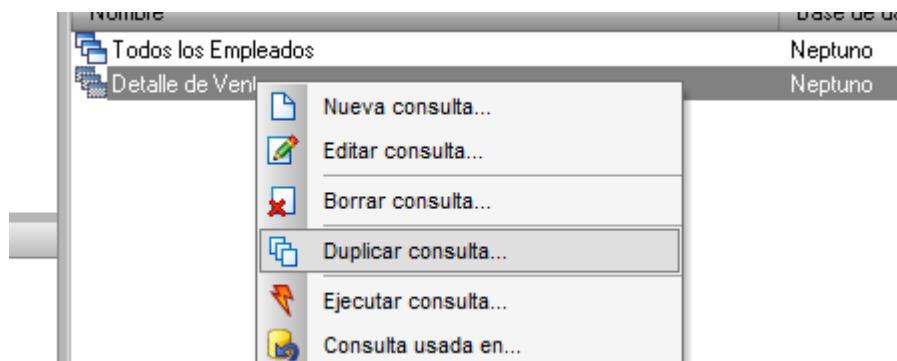
Esta consultas desarrollada con el editor en modo visual, está explicada con el editor tradicional al final en el **“Anexo Diseño de consultas con el editor tradicional”**.

Añadir un parámetro

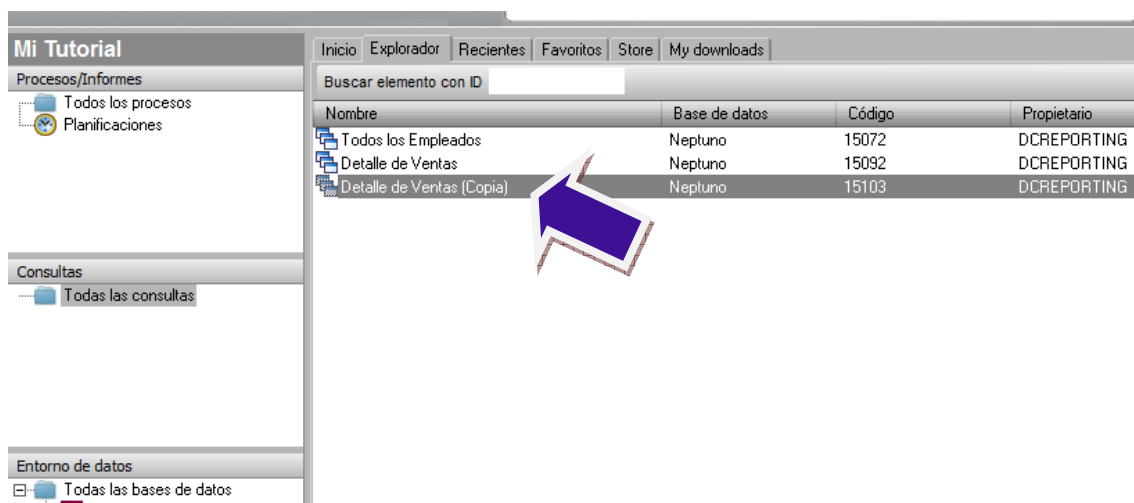
DC Reporting permite crear parámetros para reutilizar las Consultas. Mediante el diseño de Consultas SQL genéricas se reduce drásticamente el número de las mismas, dando más libertad al diseñador de Procesos en su uso al poder discriminar la información devuelta utilizando filtros.


Vamos a añadir un parámetro en la consulta de detalle ventas para que nos dé solo los datos de un año, que será introducido cuando se ejecute la consulta.

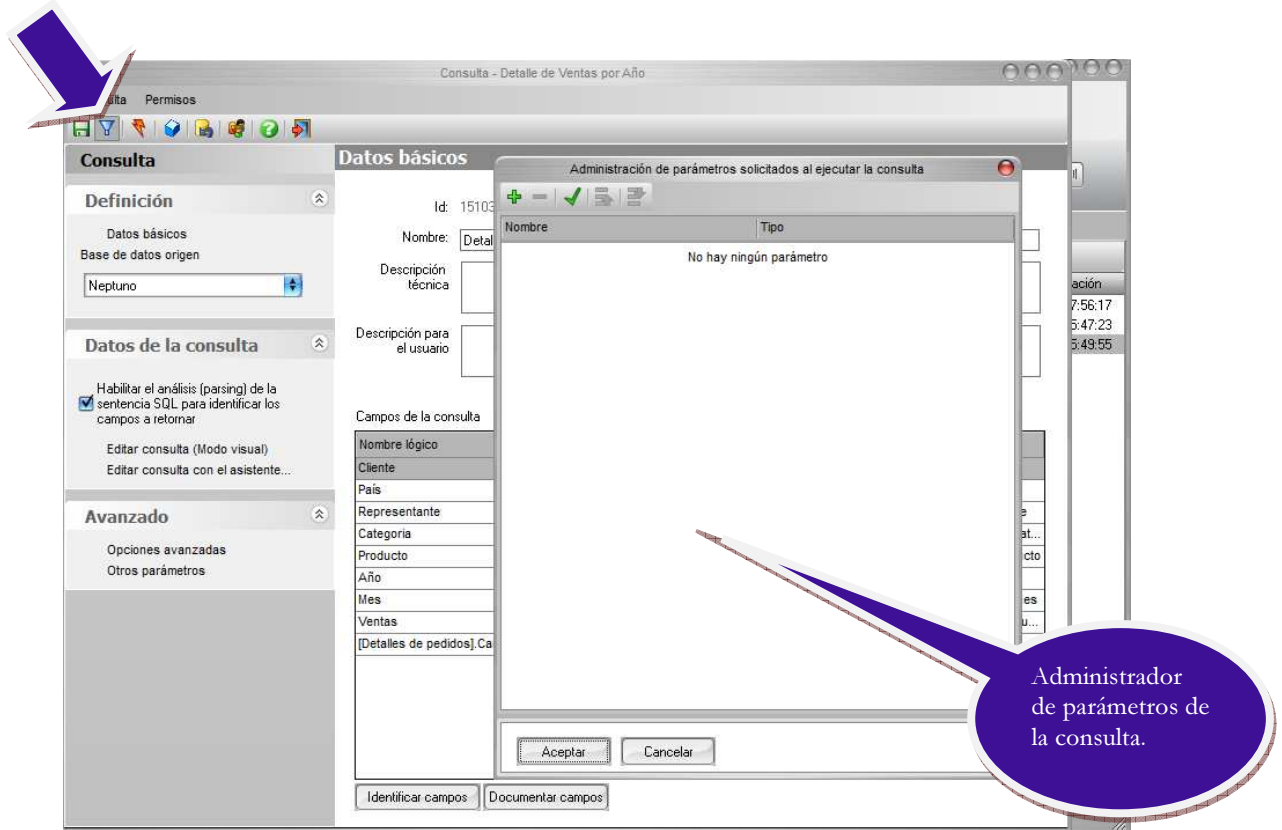
Lo primero que haremos es hacer una copia de la consulta **Detalle de Ventas**. Haciendo clic con el botón derecho del ratón sobre la consulta seleccionaremos **Duplicar consulta**.




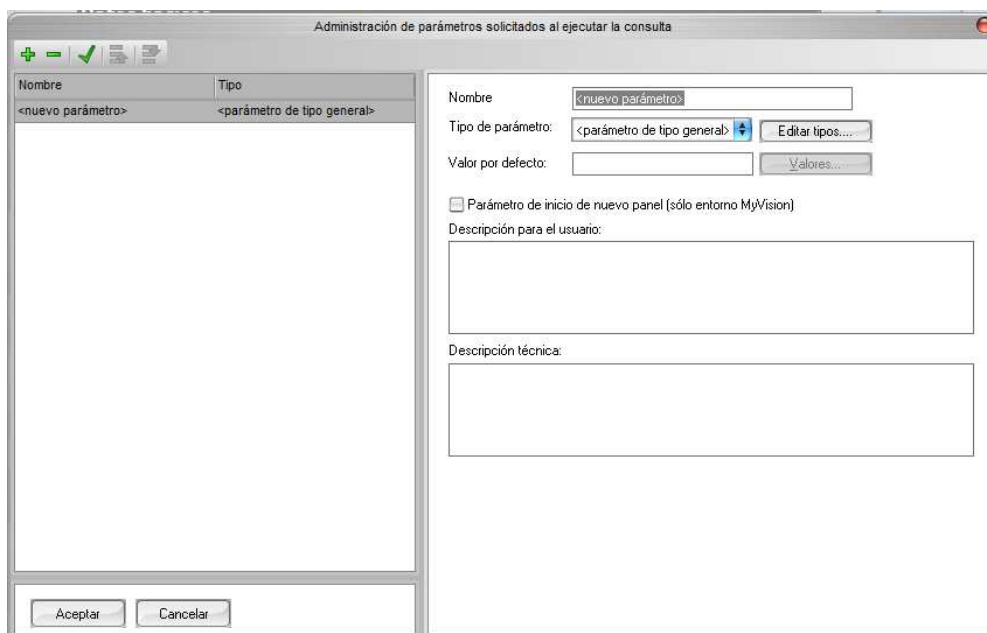
Se crea una consulta que es idéntica a la original con el mismo nombre al que le añade **(copia)**.



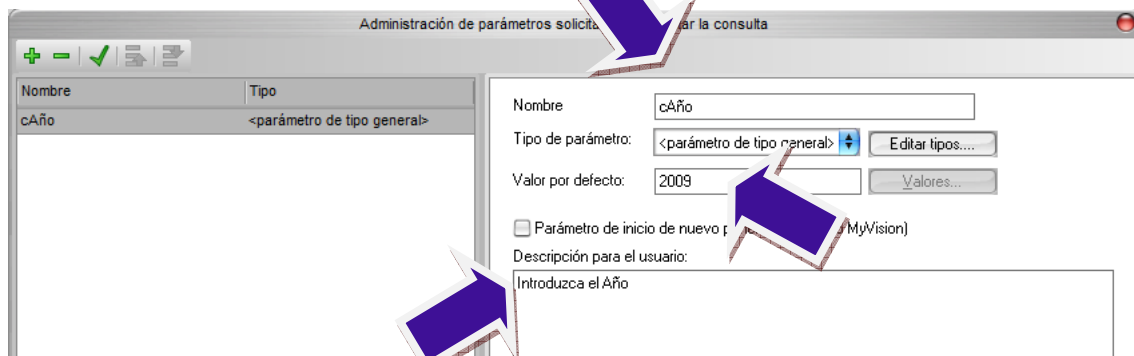
Editamos la nueva consulta, le cambiamos el nombre por **Detalle de Ventas por año**. Para añadir el parámetro pulsaremos el icono  de la barra de herramientas. Nos mostrará el administrador de parámetros de la consulta.



Hacemos clic con el ratón en el botón  de la barra de herramientas del administrador de parámetros. El administrador presentará el siguiente aspecto:

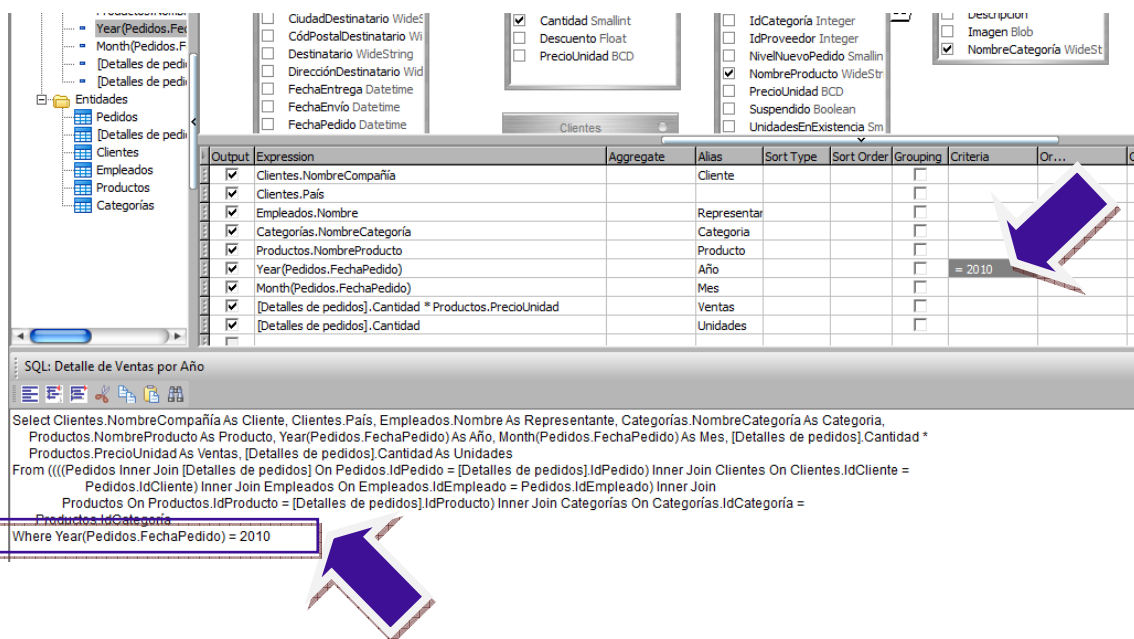


Le asignamos el nombre **cAño**, valor por defecto le ponemos **2009** y **Descripción por defecto** le indicamos el texto **Introduzca el Año**.



Cerramos la ventana con el botón de **Aceptar**, editamos la consulta en Modo visual para modificar la consulta para añadir el parámetro nuevo año.

Lo que queremos es **filtrar** los datos para un año, en el panel de columnas buscaremos el campo año y en la columna **Criterio** pondremos un valor cualquiera por ejemplo 2010.



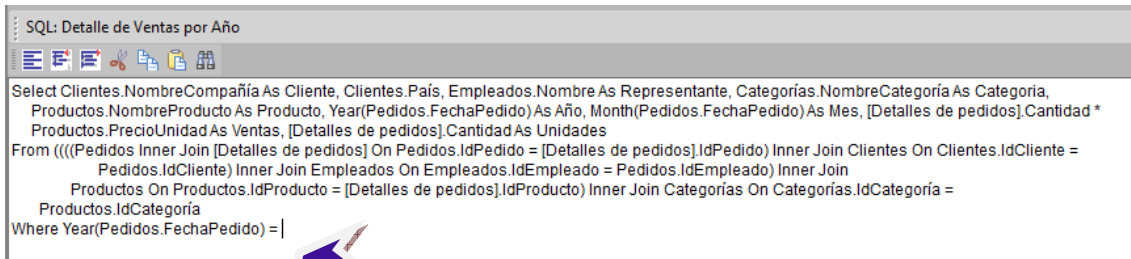
Podemos observar que en el texto de SQL ha añadido una nueva línea:

Where Year(Pedidos.FechaPedidos) = 2010

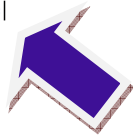
Que es la manera de decir en SQL que me devuelva solo los registros que cumplan esta condición que el año sea igual a 2010. Pero nosotros no queremos que el año sea fijo y tengamos que cambiar cada vez la consulta, esto nos lo facilita los parámetros.

En el panel de SQL texto libre sustituiremos el valor 2010 por el parámetro:

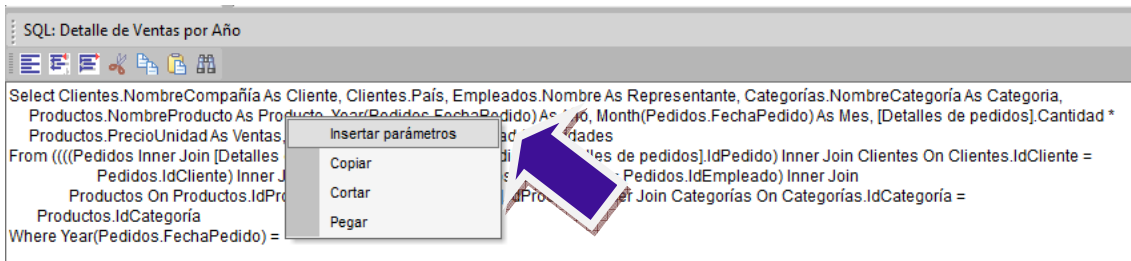
- 1) En la línea `Where Year(pedidos.FechaPedidos) = 2010`, borramos **2010**



```
SQL: Detalle de Ventas por Año
Select Clientes.NombreCompañía As Cliente, Clientes.País, Empleados.Nombre As Representante, Categorías.NombreCategoría As Categoría,
  Productos.NombreProducto As Producto, Year(Pedidos.FechaPedido) As Año, Month(Pedidos.FechaPedido) As Mes, [Detalles de pedidos].Cantidad *
  Productos.PrecioUnidad As Ventas, [Detalles de pedidos].Cantidad As Unidades
From (((Pedidos Inner Join [Detalles de pedidos] On Pedidos.IdPedido = [Detalles de pedidos].IdPedido) Inner Join Clientes On Clientes.IdCliente =
  Pedidos.IdCliente) Inner Join Empleados On Empleados.IdEmpleado = Pedidos.IdEmpleado) Inner Join
  Productos On Productos.IdProducto = [Detalles de pedidos].IdProducto) Inner Join Categorías On Categorías.IdCategoría =
  Productos.IdCategoría
Where Year(Pedidos.FechaPedido) = |
```



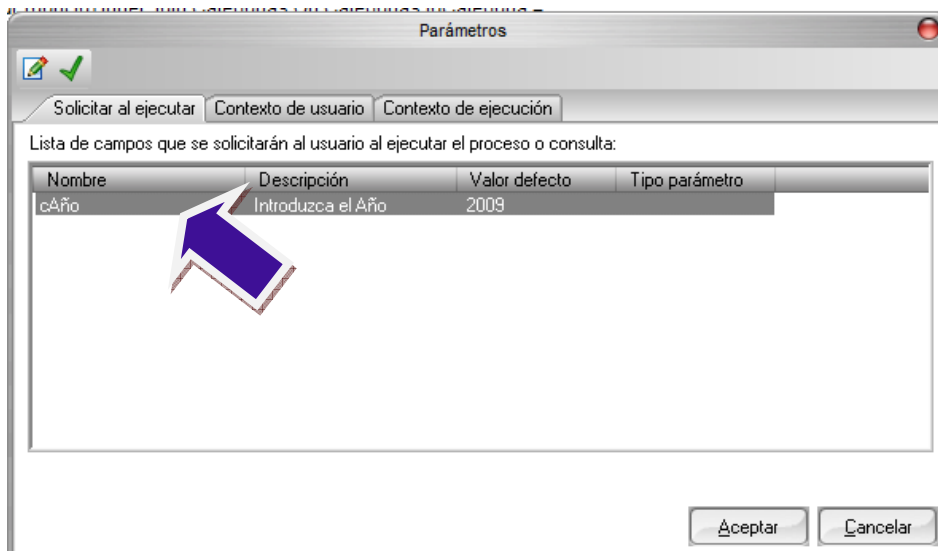
- 2) Hacemos clic con el botón derecho del ratón justo después del `=` y elegimos **Insertar parámetros**.



```
SQL: Detalle de Ventas por Año
Select Clientes.NombreCompañía As Cliente, Clientes.País, Empleados.Nombre As Representante, Categorías.NombreCategoría As Categoría,
  Productos.NombreProducto As Producto, Year(Pedidos.FechaPedido) As Año, Month(Pedidos.FechaPedido) As Mes, [Detalles de pedidos].Cantidad *
  Productos.PrecioUnidad As Ventas, [Detalles de pedidos].Cantidad As Unidades
From (((Pedidos Inner Join [Detalles de pedidos] On Pedidos.IdPedido = [Detalles de pedidos].IdPedido) Inner Join Clientes On Clientes.IdCliente =
  Pedidos.IdCliente) Inner Join Empleados On Empleados.IdEmpleado = Pedidos.IdEmpleado) Inner Join
  Productos On Productos.IdProducto = [Detalles de pedidos].IdProducto) Inner Join Categorías On Categorías.IdCategoría =
  Productos.IdCategoría
Where Year(Pedidos.FechaPedido) =
```



- 3) Nos aparece la ventana de **Parámetros**. Seleccionamos el parámetro **cAño**, que es el que hemos creado anteriormente, y **Aceptamos**.



- 4) Vemos que en el panel de SQL texto el parámetro se ha insertado, pero hasta que no hagamos clic con el ratón en otro panel no se actualizará la columna criterio del panel de columnas.

The screenshot shows the SQL Server Enterprise Manager interface. The 'SQL: Detalle de Ventas por Año' window contains the following SQL query:

```
Select Clientes.NombreCompañía As Cliente, Clientes.País, Empleados.Nombre As Representante, Categorías.NombreCategoría As Categoría,
Productos.NombreProducto As Producto, Year(Pedidos.FechaPedido) As Año, Month(Pedidos.FechaPedido) As Mes, [Detalles de pedidos].Cantidad *
Productos.PrecioUnidad As Ventas, [Detalles de pedidos].Cantidad As Unidades
From (((Pedidos Inner Join [Detalles de pedidos] On Pedidos.IdPedido = [Detalles de pedidos].IdPedido) Inner Join Clientes On Clientes.IdCliente =
Pedidos.IdCliente) Inner Join Empleados On Empleados.IdEmpleado = Pedidos.IdEmpleado) Inner Join
Productos On Productos.IdProducto = [Detalles de pedidos].IdProducto) Inner Join Categorías On Categorías.IdCategoría =
Productos.IdCategoría
Where Year(Pedidos.FechaPedido) = @PAR(cAño)
```

The column grid below the query shows the following columns:

| Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria | Or... | Or... | Or... |
|-------------------------------------|--------------------------------|-----------|---------------|-----------|------------|--------------------------|----------|-------|-------|-------|
| <input checked="" type="checkbox"/> | Clientes.NombreCompañía | | Cliente | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Clientes.País | | | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Empleados.Nombre | | Representante | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | Categoría | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | Producto | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | Year(Pedidos.FechaPedido) | | Año | | | <input type="checkbox"/> | = 2010 | | | |
| <input checked="" type="checkbox"/> | Month(Pedidos.FechaPedido) | | Mes | | | <input type="checkbox"/> | | | | |
| <input checked="" type="checkbox"/> | [Detalles de pedidos].Cantidad | | Ventas | | | <input type="checkbox"/> | | | | |

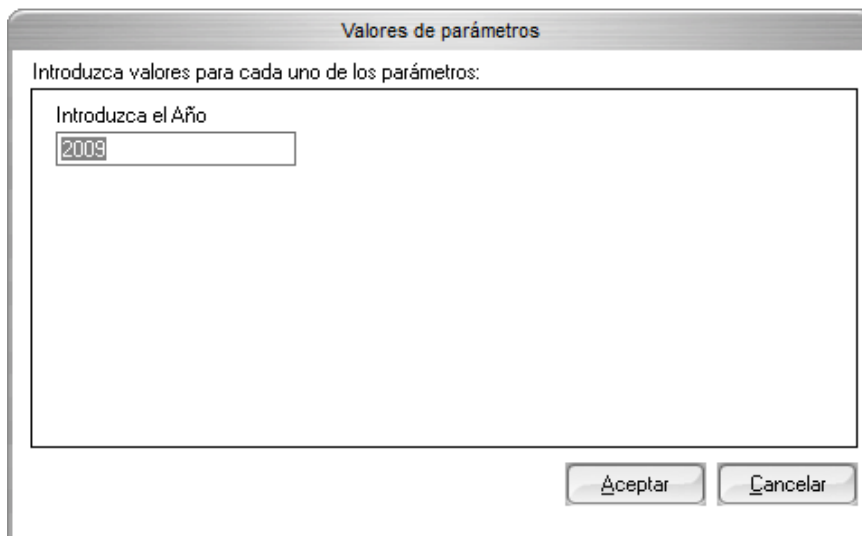
A blue arrow points to the 'Criteria' column for the 'Year(Pedidos.FechaPedido)' row, which contains the value '= 2010'.

- 5) Salimos del panel de SQL texto, haciendo clic con el ratón en el panel de objetos de la consulta. Y podemos ver que el cambio que hemos realizado en la ventana de SQL texto se aplica a los paneles visuales.

The screenshot shows the same SQL Server Enterprise Manager interface as above, but the 'Criteria' column for the 'Year(Pedidos.FechaPedido)' row now contains the value '@[PARcAño]'. A blue arrow points to this updated value.

Salimos del editor de consultas, Guardamos y ya podemos ejecutar de nuevo la consulta.

Cuando ejecutamos nos aparece una ventana donde nos solicita los valores de los parámetros de la consulta en nuestro caso nos solicita el valor del parámetro año y nos ofrece el valor 2009, que es el que definimos como valor por defecto en la creación del parámetro.



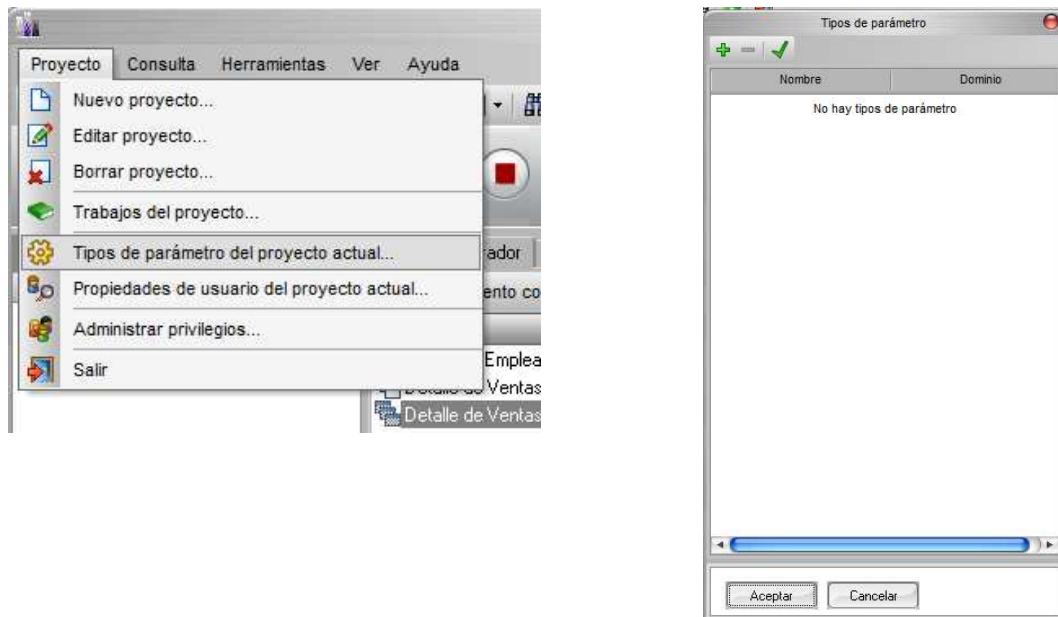
Podemos comprobar que el resultado de la consulta es el correcto, solo los registros del año que hemos solicitado.


| Cliente | País | Representante | Categoría | Producto | Año | Mes | Ventas | Unidades |
|------------------------|-----------|---------------|-----------------|--|------|-----|--------|----------|
| Wilman Kala | Finlandia | Steven | Lácteos | Queso Cabrales | 2009 | 8 | 252 | 12 |
| Wilman Kala | Finlandia | Steven | Granos/Cereales | Tallarines de Singapur | 2009 | 8 | 140 | 10 |
| Wilman Kala | Finlandia | Steven | Lácteos | Queso Mozzarella Giovanni | 2009 | 8 | 174 | 5 |
| Toms Spezialitäten | Alemania | Michael | Frutas/Verduras | Cuajada de judías | 2009 | 8 | 209,25 | 9 |
| Toms Spezialitäten | Alemania | Michael | Frutas/Verduras | Manzanas secas Manjimup | 2009 | 8 | 2120 | 40 |
| Hanari Carnes | Brasil | Margaret | Pescado/Marisco | Crema de almejas estilo Nueva Inglaterra | 2009 | 8 | 96,5 | 10 |
| Hanari Carnes | Brasil | Margaret | Frutas/Verduras | Manzanas secas Manjimup | 2009 | 8 | 1855 | 35 |
| Hanari Carnes | Brasil | Margaret | Condimentos | Salsa de pimiento picante de Luisiana | 2009 | 8 | 315,75 | 15 |
| Victuailles en stock | Francia | Janet | Granos/Cereales | Pan de centeno crujiente estilo Gustaf's | 2009 | 8 | 126 | 6 |
| Victuailles en stock | Francia | Janet | Granos/Cereales | Raviolis Angelo | 2009 | 8 | 292,5 | 15 |
| Victuailles en stock | Francia | Janet | Condimentos | Salsa de pimiento picante de Luisiana | 2009 | 8 | 421 | 20 |
| Suprêmes délices | Bélgica | Margaret | Repostería | Mermelada de Sir Rodney's | 2009 | 8 | 3240 | 40 |
| Suprêmes délices | Bélgica | Margaret | Lácteos | Queso de cabra | 2009 | 8 | 62,5 | 25 |
| Suprêmes délices | Bélgica | Margaret | Lácteos | Camembert Pierrot | 2009 | 8 | 1360 | 40 |
| Hanari Carnes | Brasil | Janet | Lácteos | Queso gorgonzola Telino | 2009 | 8 | 250 | 20 |
| Hanari Carnes | Brasil | Janet | Bebidas | Licor verde Chartreuse | 2009 | 8 | 756 | 42 |
| Hanari Carnes | Brasil | Janet | Repostería | Regaliz | 2009 | 8 | 800 | 40 |
| Chop-suey Chinese | Suiza | Steven | Bebidas | Refresco Guaraná Fantástica | 2009 | 8 | 67,5 | 15 |
| Chop-suey Chinese | Suiza | Steven | Carnes | Paté chino | 2009 | 8 | 504 | 21 |
| Chop-suey Chinese | Suiza | Steven | Frutas/Verduras | Queso de soja Longlife | 2009 | 8 | 210 | 21 |
| Richter Supermarkt | Suiza | Anne | Bebidas | Cerveza tibetana Barley | 2009 | 8 | 380 | 20 |
| Richter Supermarkt | Suiza | Anne | Repostería | Postre de merengue Pavlova | 2009 | 8 | 610,75 | 35 |
| Richter Supermarkt | Suiza | Anne | Pescado/Marisco | Escabeche de arenque | 2009 | 8 | 475 | 25 |
| Richter Supermarkt | Suiza | Anne | Lácteos | Raclet de queso Courdavault | 2009 | 8 | 1650 | 30 |
| Wellington Importadora | Brasil | Janet | Carnes | Empanada de carne | 2009 | 8 | 492 | 15 |
| Wellington Importadora | Brasil | Janet | Condimentos | Salsa verde original Frankfurter | 2009 | 8 | 156 | 12 |

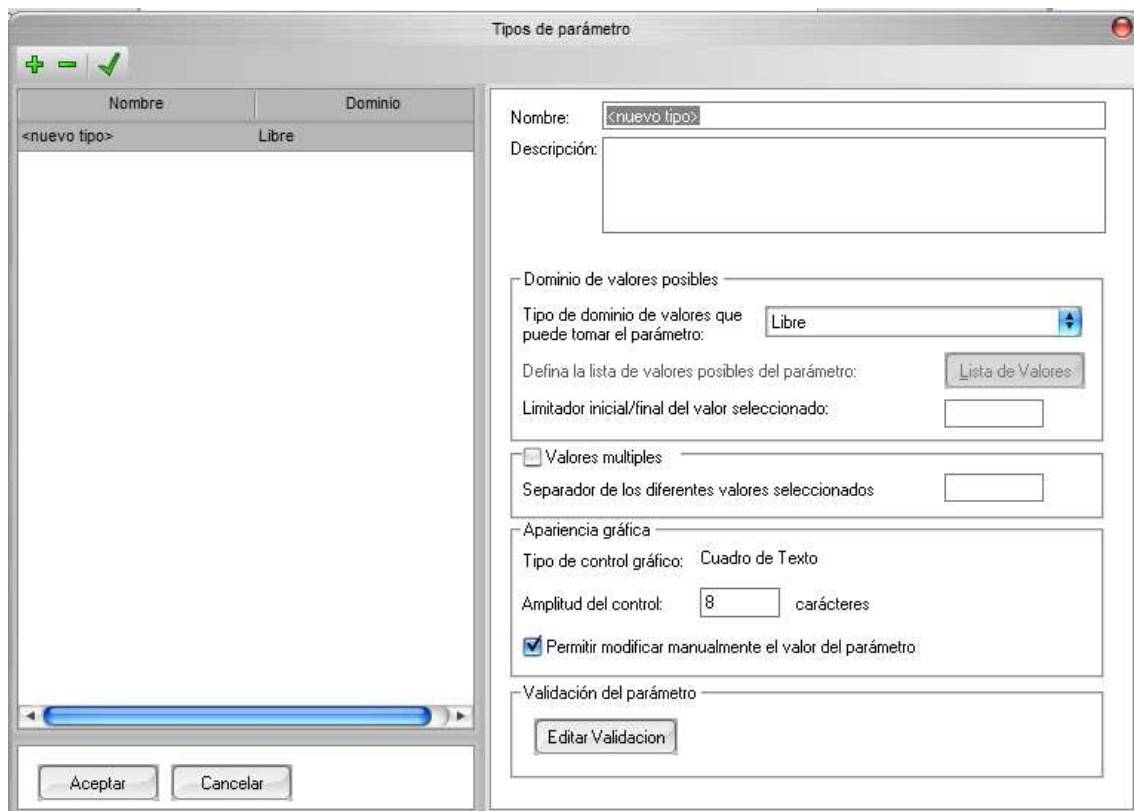
Añadir un parámetro tipo Lista de Valores

Vamos a añadir un nuevo parámetro que nos permita seleccionar el mes además del año, para que la consulta Detalle de Ventas solo devuelva los registros de un año y mes concreto.

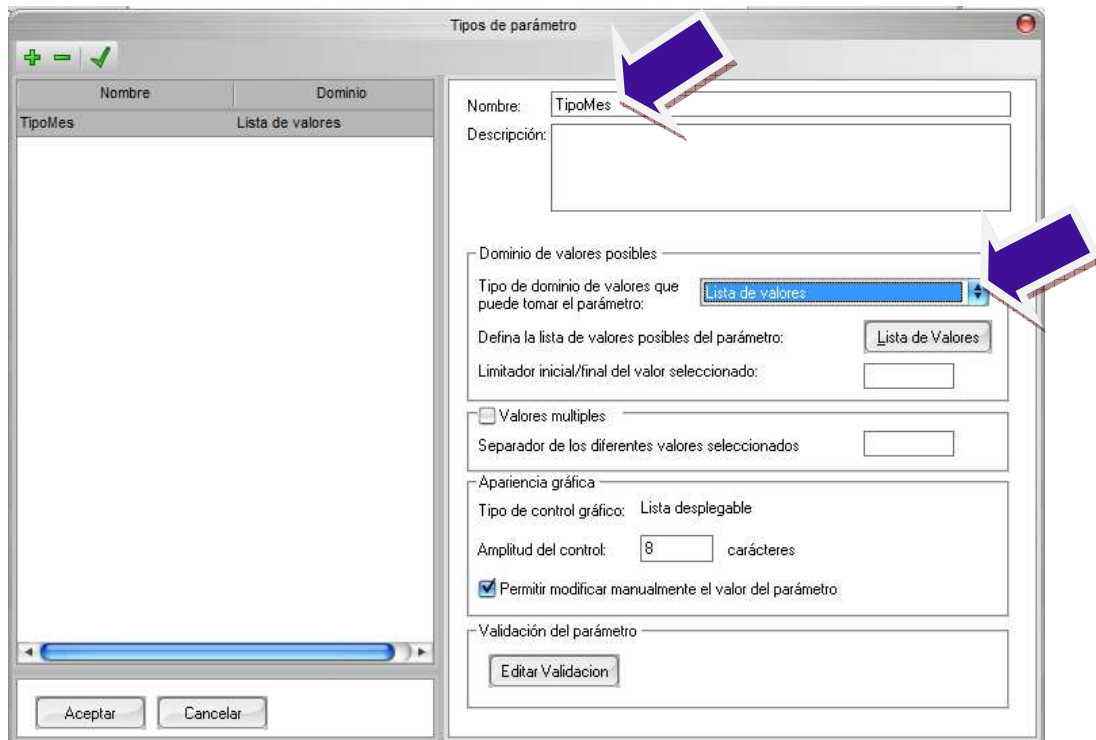
Antes de cambiar la consulta, vamos a crear un nuevo tipo de parámetro, en la pantalla principal de DC Reporting vamos a **Menú Proyecto** → **Tipos de parámetros del proyecto actual**. Nos aparece la ventana de Tipos de parámetros



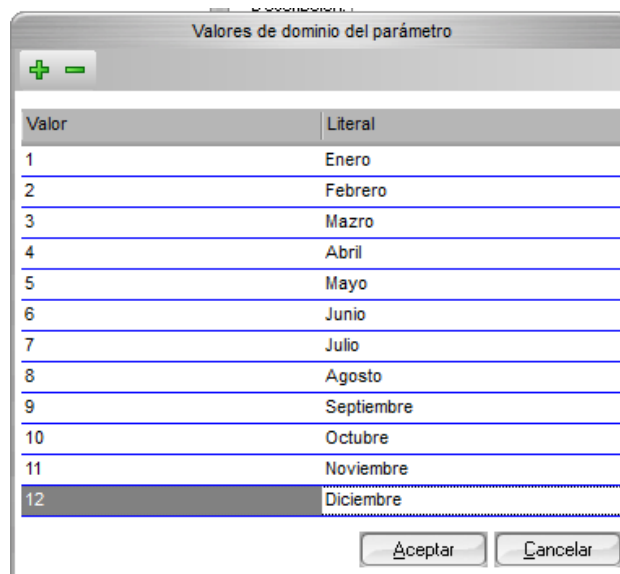
Hacemos clic con el ratón en el icono  de la barra de herramientas de esta nueva ventana., nos presentará el siguiente aspecto:



Le asignamos el nombre **TipoMes** y en la lista desplegable de tipos de dominios elegimos **Lista de Valores**.



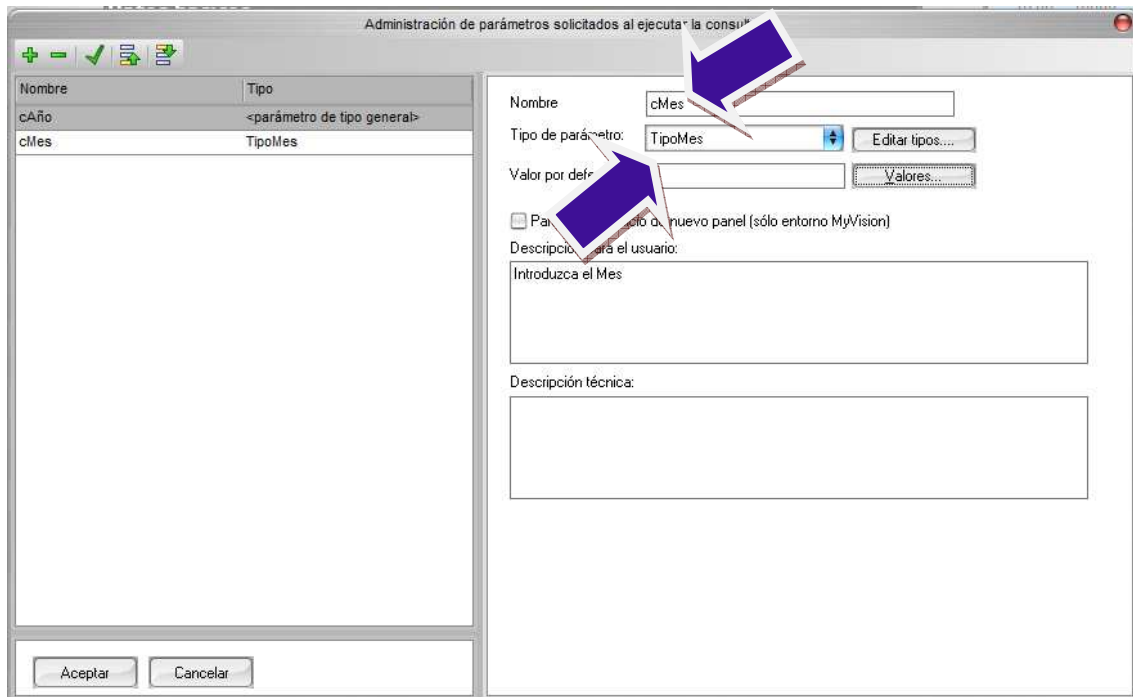
Hacemos ahora clic con el ratón en el botón **Lista de Valores**, donde aparece una ventana para que se introduzca la lista de valores. Introduciremos los elementos uno a uno utilizando la flecha hacia abajo para añadir un nuevo elemento. Para cada elemento de la lista en la casilla **valor** introduciremos el número de mes y en la casilla **Literal** la descripción. Después de introducir los 12 valores el aspecto de esta ventana tendría que ser el siguiente:



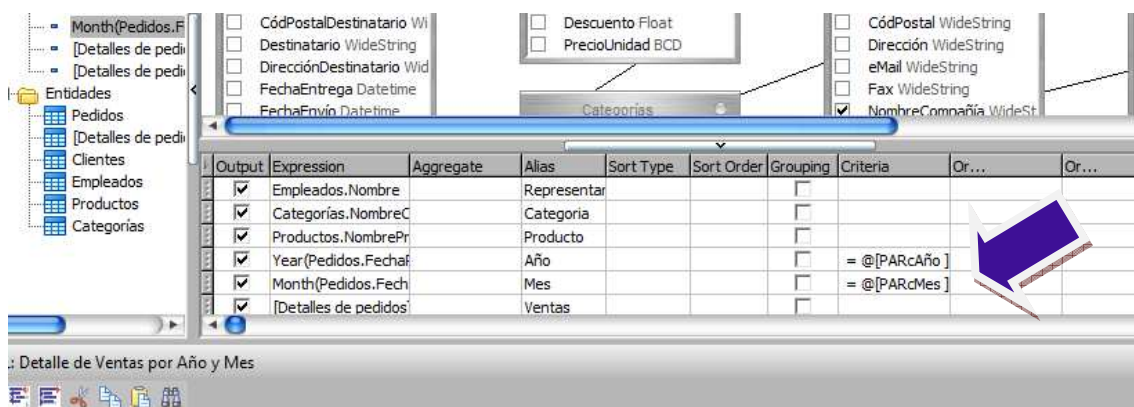
Aceptamos para salir de la ventana de valores y aceptamos para salir de la ventana de tipos de parámetros.

Duplicamos la consulta **Detalle de Ventas por año**, la editamos, le cambiamos el nombre por **Detalle de Ventas por año y mes**. Nos vamos al editor de parámetros de la consulta para añadirle un nuevo parámetro.

Al nuevo parámetro le llamaremos **Mes** y en la lista desplegable de **Tipo de parámetros** elegimos **TipoMes** (que es el tipo que hemos creado anteriormente). Le podríamos añadir un valor por defecto pero vamos a dejarlo en blanco.



Aceptamos y editamos la consulta con el editor Modo Visual. Y hacemos los mismos pasos que hicimos para añadir el parámetro años en el apartado anterior. Pero en este caso utilizaremos el campo **Month(Pedidos.FechaPedidos)** que es el campo Mes. El aspecto que ha de tener la consulta después de las modificaciones debería ser:



```

Clientes.NombreCompañía As Cliente, Clientes.País, Empleados.Nombre As Representante, Categorías.NombreCategoría As Categoría,
Productos.NombreProducto As Producto, Year(Pedidos.FechaPedido) As Año, Month(Pedidos.FechaPedido) As Mes, [Detalles de pedidos].Cantidad *
Productos.PrecioUnidad As Ventas, [Detalles de pedidos].Cantidad As Unidades
((((Pedidos Inner Join [Detalles de pedidos] On Pedidos.IdPedido = [Detalles de pedidos].IdPedido) Inner Join Clientes On Clientes.IdCliente =
Pedidos.IdCliente) Inner Join Empleados On Empleados.IdEmpleado = Pedidos.IdEmpleado) Inner Join
Productos On Productos.IdProducto = [Detalles de pedidos].IdProducto) Inner Join Categorías On Categorías.IdCategoría =
Productos.IdCategoría
Year(Pedidos.FechaPedido) = @[PAR(cAño)] And Month(Pedidos.FechaPedido) = @[PAR(cMes)]

```

Salimos del modo asistente de consultas, grabamos la consulta y la ejecutamos. Podemos ver que nos pide valores para los dos parámetros.

Valores de parámetros

Introduzca valores para cada uno de los parámetros:

Introduzca el Año
2009

Introduzca el Mes
Enero
Febrero
Mazro
Abril
Mayo
Junio
Julio
Agosto

Aceptar Cancelar


Le damos valores, aceptamos y podremos comprobar que el resultado de la consulta es el deseado. Los registros del año y mes seleccionados en la ventana de parámetros al ejecutar la consulta.

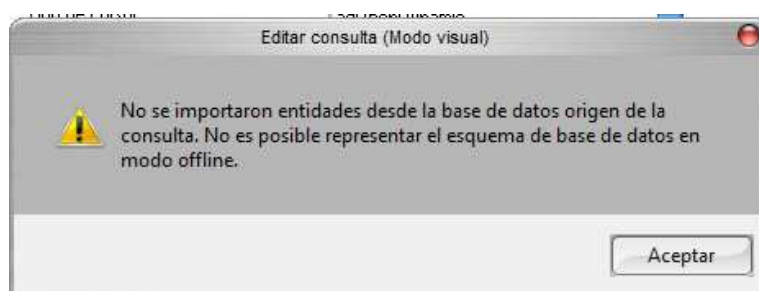
| Cliente | País | Representante | Categoría | Producto | Año | Mes | Ventas | Unidades |
|------------------------|-----------|---------------|-----------------|--|------|-----|--------|----------|
| Wilman Kala | Finlandia | Steven | Lácteos | Queso Cabrales | 2009 | 8 | 252 | 12 |
| Wilman Kala | Finlandia | Steven | Granos/Cereales | Tallarines de Singapur | 2009 | 8 | 140 | 10 |
| Wilman Kala | Finlandia | Steven | Lácteos | Queso Mozzarella Giovanni | 2009 | 8 | 174 | 5 |
| Toms Spezialitäten | Alemania | Michael | Frutas/Verduras | Cuajada de judías | 2009 | 8 | 209,25 | 9 |
| Toms Spezialitäten | Alemania | Michael | Frutas/Verduras | Manzanas secas Manjimup | 2009 | 8 | 2120 | 40 |
| Hanari Carnes | Brasil | Margaret | Pescado/Marisco | Crema de almejas estilo Nueva Inglaterra | 2009 | 8 | 96,5 | 10 |
| Hanari Carnes | Brasil | Margaret | Frutas/Verduras | Manzanas secas Manjimup | 2009 | 8 | 1855 | 35 |
| Hanari Carnes | Brasil | Margaret | Condimentos | Salsa de pimiento picante de Luisiana | 2009 | 8 | 315,75 | 15 |
| Victuailles en stock | Francia | Janet | Granos/Cereales | Pan de centeno crujiente estilo Gustaf's | 2009 | 8 | 126 | 6 |
| Victuailles en stock | Francia | Janet | Granos/Cereales | Raviolis Angelo | 2009 | 8 | 292,5 | 15 |
| Victuailles en stock | Francia | Janet | Condimentos | Salsa de pimiento picante de Luisiana | 2009 | 8 | 421 | 20 |
| Suprêmes délices | Bélgica | Margaret | Repostería | Mermelada de Sir Rodney's | 2009 | 8 | 3240 | 40 |
| Suprêmes délices | Bélgica | Margaret | Lácteos | Queso de cabra | 2009 | 8 | 62,5 | 25 |
| Suprêmes délices | Bélgica | Margaret | Lácteos | Camembert Pierrot | 2009 | 8 | 1360 | 40 |
| Hanari Carnes | Brasil | Janet | Lácteos | Queso gorgonzola Telino | 2009 | 8 | 250 | 20 |
| Hanari Carnes | Brasil | Janet | Bebidas | Licor verde Chartreuse | 2009 | 8 | 756 | 42 |
| Hanari Carnes | Brasil | Janet | Repostería | Regaliz | 2009 | 8 | 800 | 40 |
| Chop-suey Chinese | Suiza | Steven | Bebidas | Refresco Guaraná Fantástica | 2009 | 8 | 67,5 | 15 |
| Chop-suey Chinese | Suiza | Steven | Carnes | Paté chino | 2009 | 8 | 504 | 21 |
| Chop-suey Chinese | Suiza | Steven | Frutas/Verduras | Queso de soja Longlife | 2009 | 8 | 210 | 21 |
| Richter Supermarkt | Suiza | Anne | Bebidas | Cerveza tibetana Barley | 2009 | 8 | 390 | 20 |
| Richter Supermarkt | Suiza | Anne | Repostería | Postre de merengue Pavlova | 2009 | 8 | 610,75 | 35 |
| Richter Supermarkt | Suiza | Anne | Pescado/Marisco | Escabeche de arenque | 2009 | 8 | 475 | 25 |
| Richter Supermarkt | Suiza | Anne | Lácteos | Raclet de queso Courdavault | 2009 | 8 | 1650 | 30 |
| Wellington Importadora | Brasil | Janet | Carnes | Empanada de carne | 2009 | 8 | 492 | 15 |
| Wellington Importadora | Brasil | Janet | Condimentos | Salsa verde original Frankfurter | 2009 | 8 | 156 | 12 |
| HII ARIÍN-Ahastns | Venezuela | Margaret | Repostería | Chocolate Schoggi | 2009 | 8 | 1097,5 | 25 |

Registros cargados: 59 Nº registros leídos: 59 Registro actual: 0 T. conexión: 00:00:01 ImageManager Property Page

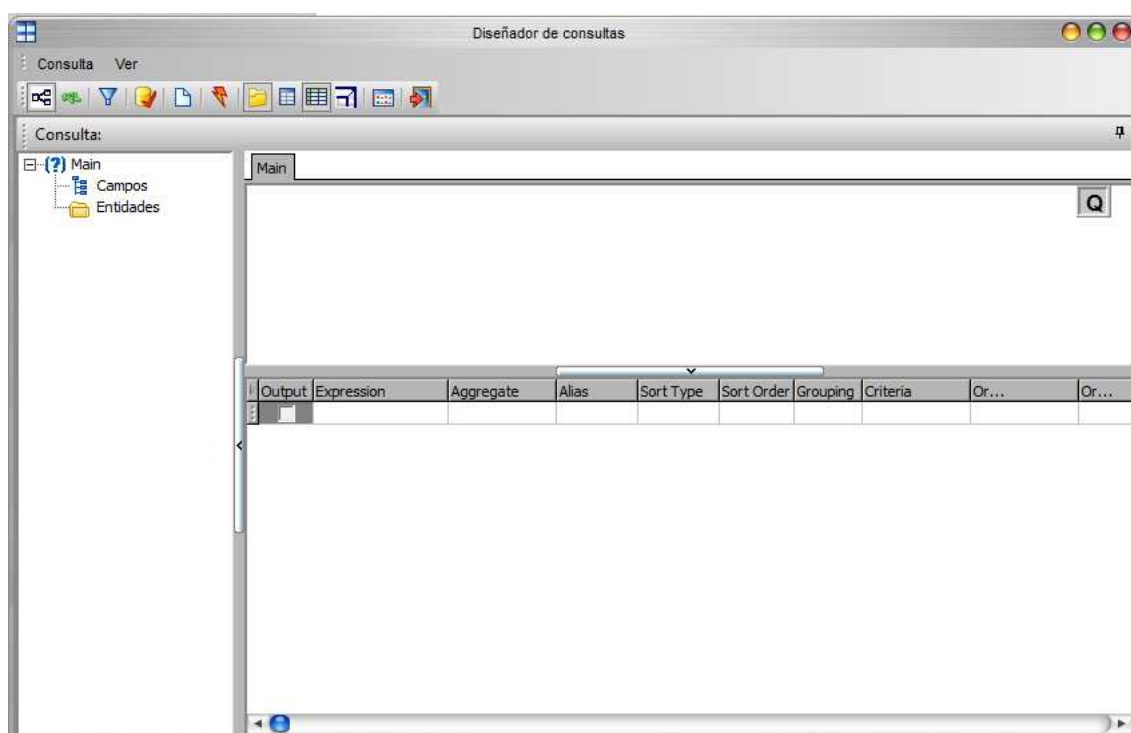
Categorías y productos sin existencias

Vamos a crear otra consulta, en este caso vamos a obtener aquellos productos con sus categorías que están en pedidos y no hay existencias.

Nos situaremos en la sección de consultas y pulsar sobre el botón  de la barra de herramientas. Aparecerá la ventana de selección de bases de datos, seleccionaremos Neptuno. Aparecerá la ventana vacía de edición de la consulta. Le asignamos el nombre de **Productos sin existencias**. En **Opciones avanzadas** en el apartado **Conexión de la base de datos** activaremos el **Modo OffLine**. Grabamos la consulta y a continuación editamos la consulta en **Modo visual**. Nos aparecerá una ventana de aviso:



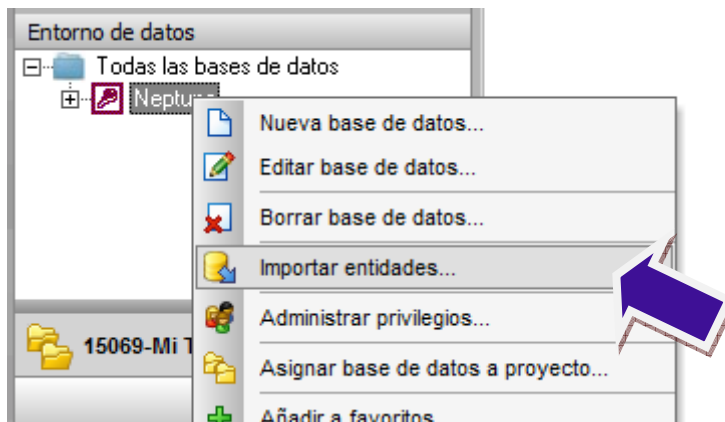
Aceptamos y nos muestra el editor de consultas con el siguiente aspecto:



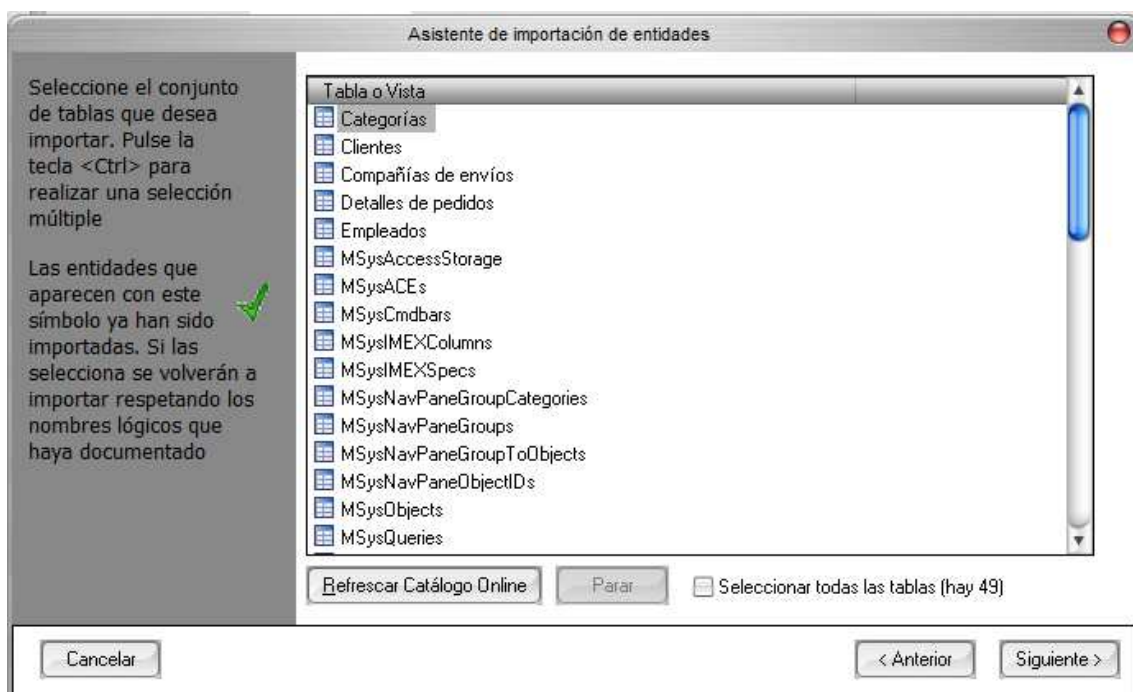
Como podemos observar no tenemos ninguna tabla para seleccionar y diseñar nuestra consulta. Esto es debido a que estamos en modo OffLine. Cuando estamos en este modo el editor no se conecta a la BBDD para obtener la información de las entidades y relaciones que existen en ella. Salimos del editor y de la consulta.

Importar catalogo

A continuación vamos a decidir qué **tablas** o **ficheros** (**entidades** en la terminología de DataCycle) vamos a usar para hacer la consulta. Hacer clic con el ratón en la sección de base de datos, en menú Base de datos seleccionar **Importar catálogo**.



Aparecerá el siguiente asistente:



Se muestra todas las tablas de la base de datos Neptuno y nos permite seleccionar qué tablas son las que queremos usar.

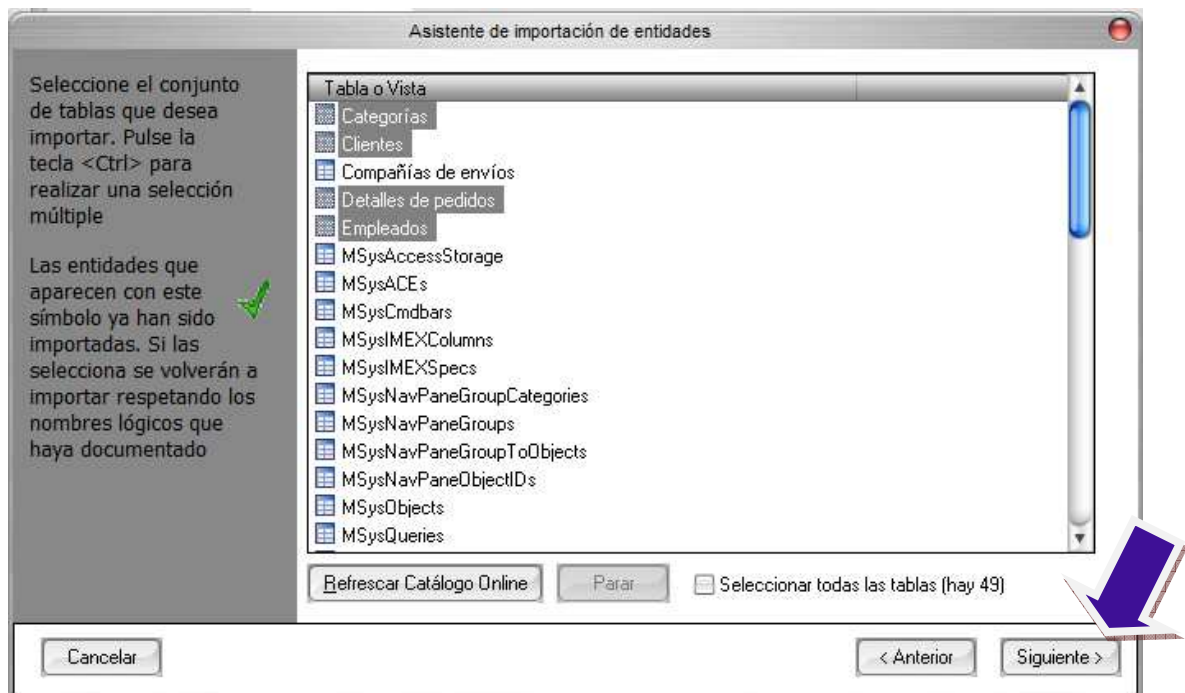
Es importante destacar que en este paso la operación que realizamos es realmente **importar la estructura o definición de las tablas** que pensamos usar en nuestras Consultas. Es una operación que se realiza pocas veces (al principio es imprescindible si se van a diseñar las consultas en modo OffLine) y posteriormente podremos incluir más tablas si las necesitamos. De este modo, DataCycle Reporting conoce la estructura de campos de las tablas de la base de

datos Neptuno. Es interesante recalcar que este proceso **no importa los datos** de las tablas, sino que **sólo se importa la estructura de las tablas**.

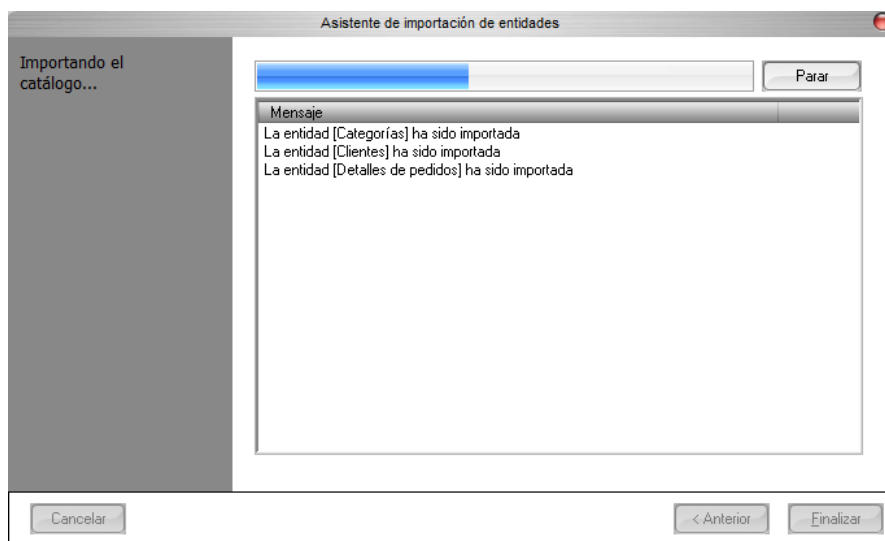
En nuestro caso, vamos a seleccionar las tablas siguientes (aunque solo necesitaríamos Categorías y Productos) mediante su señalización con un clic del ratón y la tecla CTRL pulsada (vamos a hacer una selección múltiple):

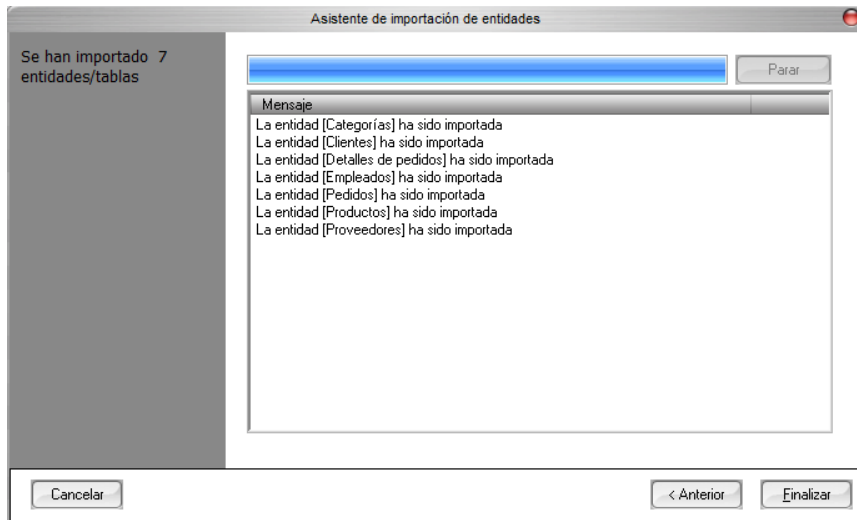
- **Categorías, Clientes, Compañías de envíos, Detalles de pedidos, Empleados, Pedidos, Productos y Proveedores**

Usar la barra de desplazamiento lateral para buscar las entidades.



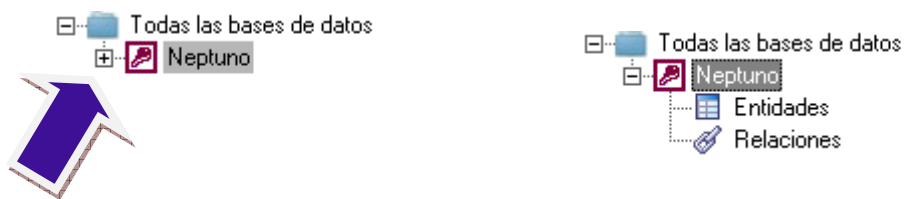
Pulsamos el botón **Siguiete** para el siguiente paso. Iniciaré el proceso de importación de la estructura de campos.



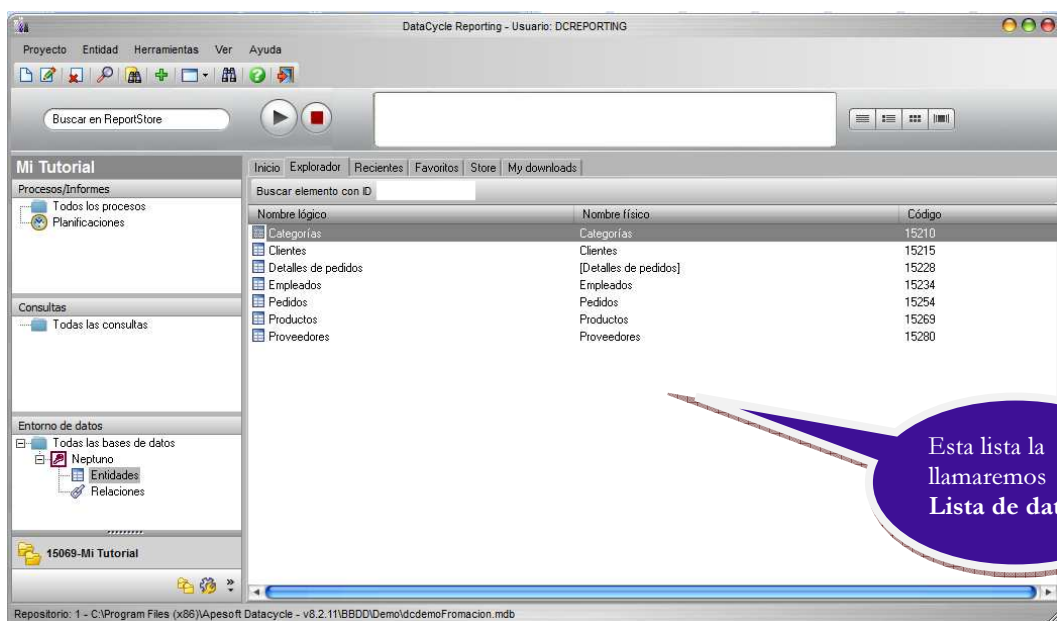


Una vez finalizado el proceso pulsamos el botón **Finalizar** para volver a la ventana principal de DataCycle.

En el entorno de datos expandimos el elemento DB Neptuno nos encontraremos con dos elementos:

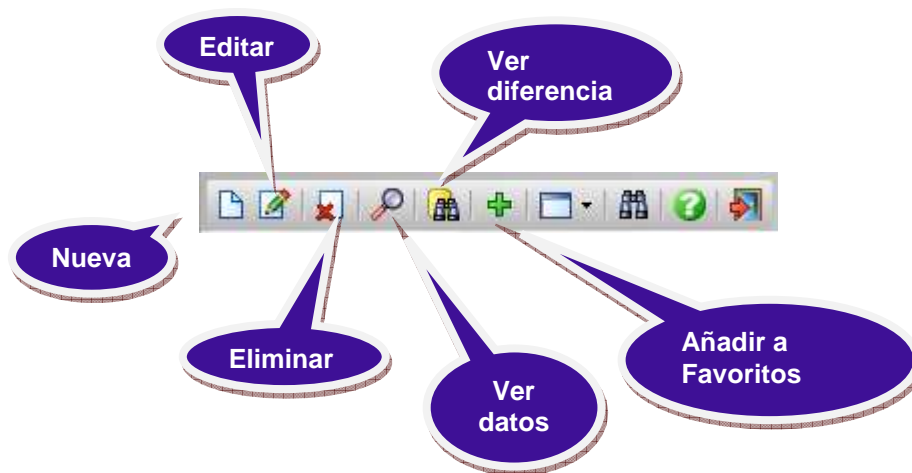
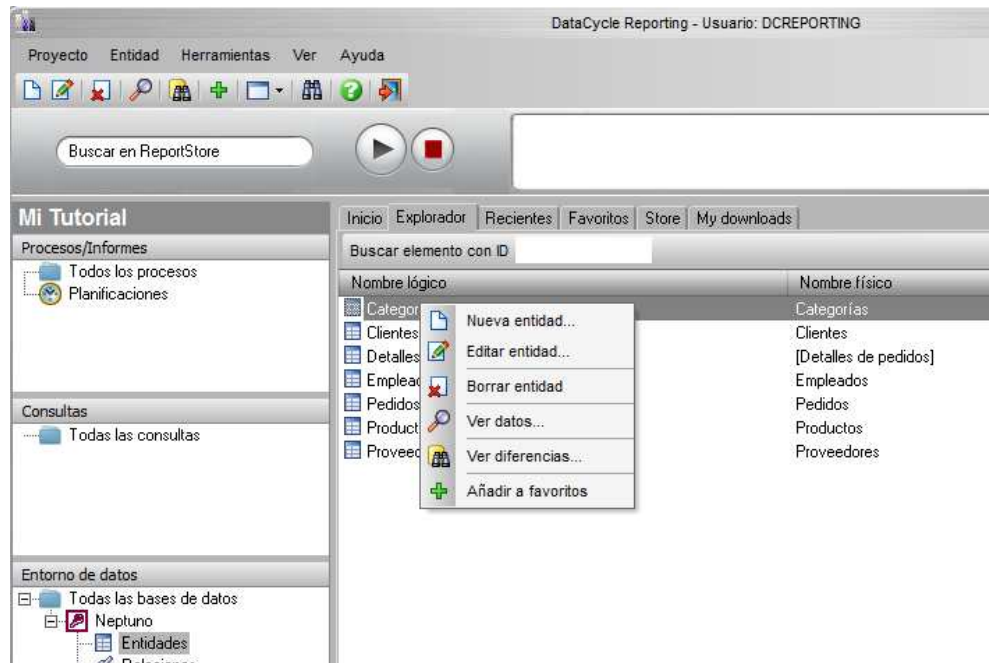


Seleccionamos con el ratón el elemento **Entidades** y se muestran en la parte derecha de la ventana principal, una lista de **tablas**, precisamente las que acabamos de importar (sólo su definición) dentro de la base de datos Neptuno.



A partir de ahora vamos a llamar *lista de datos* a la lista de la derecha de la ventana principal.

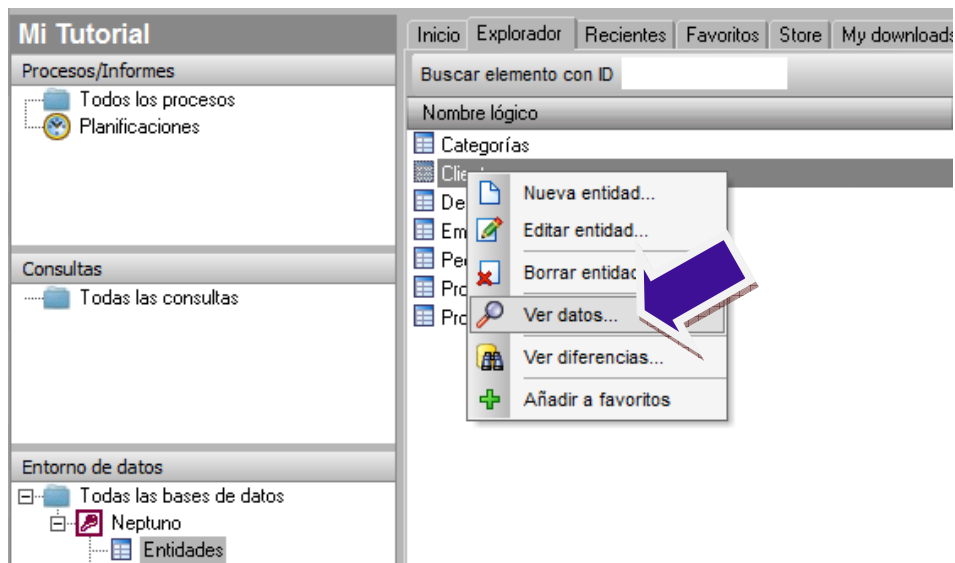
Los elementos de la lista de datos también tienen su menú de opciones que se activa con el botón derecho del ratón sobre un elemento de la lista. Las opciones del menú se corresponden con los iconos de la barra de herramientas superior:



Vamos a consultar los datos de la tabla **Cientes**. Esto representará conectarse "on-line" para que DataCycle Reporting consulte los datos y los muestre por pantalla. Realizamos un clic con el botón derecho del ratón sobre la **lista de datos**, concretamente sobre el elemento de la lista que representa la tabla de clientes:

Vamos a consultar los datos de la tabla **Cientes**. Esto representará conectarse "on-line" para que DataCycle Reporting consulte los datos y los muestre por pantalla. Realizamos un clic con el botón derecho del ratón sobre la **lista de datos**, concretamente sobre el elemento de la lista que representa la tabla de clientes:

Seleccionamos la opción **Ver datos**.



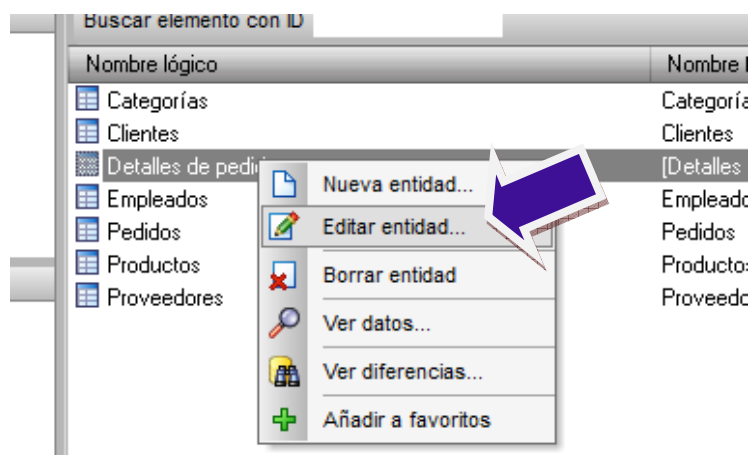
En este momento DataCycle Reporting consulta el contenido de la tabla **Cientes** y muestra la ventana de visor de datos:

The screenshot shows a data viewer window titled '0-Valores de la tabla Cientes'. The table contains the following data:

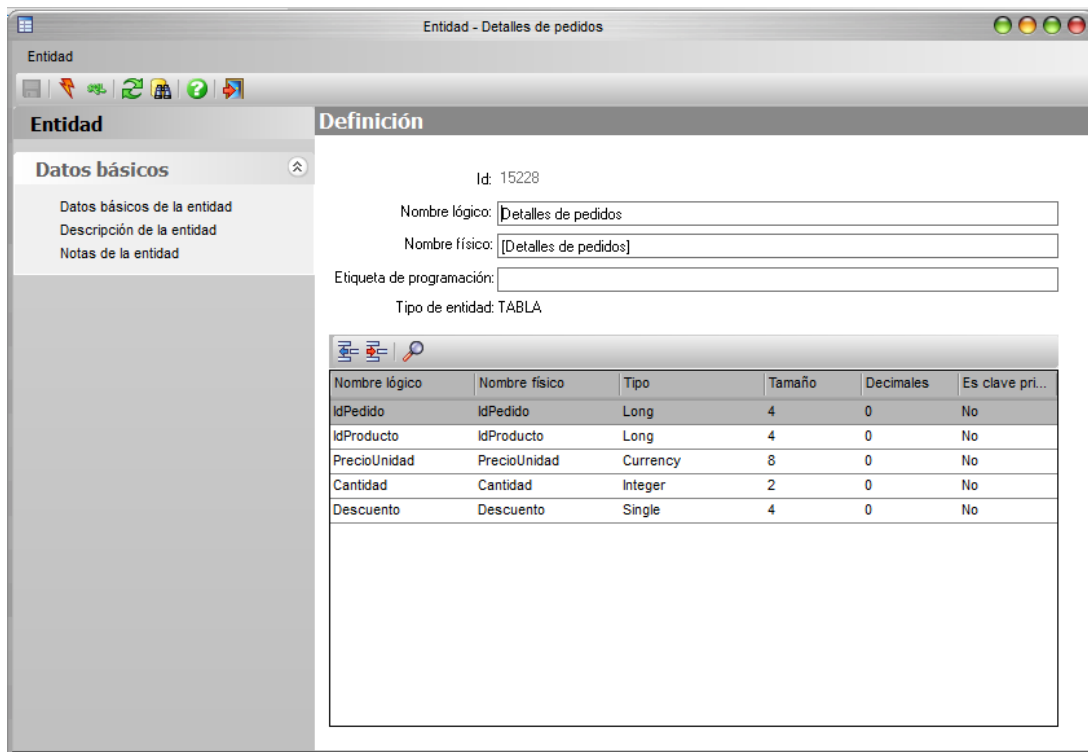
| IdCliente | NombreCompañía | NombreContacto | CargoContacto | Dirección | Ciudad | Región |
|-----------|------------------------------------|--------------------|--------------------------|-------------------------------|--------------|--------|
| ALFKJ | Alfki | Maria Anders | Representante de ventas | Obere Str. 57 | Berlín | |
| ANAT | Ana Trujillo Emparedados y helados | Ana Trujillo | Propietario | Avda. de la Constitución 2222 | México D.F. | |
| ANTON | Antonio Moreno Taquer | Antonio Moreno | Propietario | Maladeros 2312 | México D.F. | |
| AROUT | Around the Horn | Thomas Hardy | Representante de ventas | 120 Hanover Sq | Londres | |
| BERGS | Berglunds snabbköp | Christina Berglund | Administrador de pedidos | Berguvsvägen 8 | Luleå | |
| BLAUS | Blauer See Delikatessen | Hanna Moos | Representante de ventas | Forsterstr. 57 | Mannheim | |
| BLONP | Blondel père et fils | Frédérique Citeaux | Gerente de marketing | 24, place Kléber | Estrasburgo | |
| BOLID | Bólido Comidas preparadas | Martin Sommer | Propietario | C/ Araquil, 67 | Madrid | |
| BONAP | Bon app' | Laurence Lebihan | Propietario | 12, rue des Bouchers | Marzella | |
| BOTTM | Bottom-Dollar Markets | Elizabeth Lincoln | Gerente de contabilidad | 23 Tsawassen Blvd. | Tsawassen | BC |
| BSBEV | B's Beverages | Victoria Ashworth | Representante de ventas | Fauntleroy Circus | Londres | |
| CACTU | Cactus Comidas para llevar | Patricio Simpson | Agente de ventas | Cerrito 333 | Buenos Aires | |
| CENTC | Centro comercial Moctezuma | Francisco Chang | Gerente de marketing | Sierras de Granada 9993 | México D.F. | |
| CHOPS | Chop-suey Chinese | Yang Wang | Propietario | Hauptstr. 29 | Berna | |
| COMMI | Comércio Mineiro | Pedro Alonso | Asistente de ventas | Av. dos Lusíadas, 23 | São Paulo | SP |
| CDNSH | Consolidated Holdings | Elizabeth Brown | Representante de ventas | Berkeley Gardens | Londres | |
| DRACD | Drachenblut Delikatessen | Sven Ottlieb | Administrador de pedidos | Walsenweg 21 | Aachen | |
| DUMDN | Du monde entier | Janine Labruno | Propietario | 67, rue des Cinquante Otages | Nantes | |
| EASTC | Eastern Connection | Ann Devon | Agente de ventas | 35 King George | Londres | |
| ERNSH | Ernst Handel | Roland Mendel | Gerente de ventas | Kirchgasse 6 | Graz | |
| FAMIA | Familia Arquibaldo | Ária Cruz | Asistente de marketing | Rua Piratininga, 92 | San Paulo | SP |


At the bottom of the window, it shows: 'Registros cargados: 91', 'Nº registros leídos: 91', 'Registro actual: 0', 'T. conexión: 00:00:01', and 'ImageManager Property Page'.

El catálogo de entidades contiene la definición de las mismas, sus campos y tipos. Haciendo doble clic sobre la entidad o bien seleccionando la opción Edición.

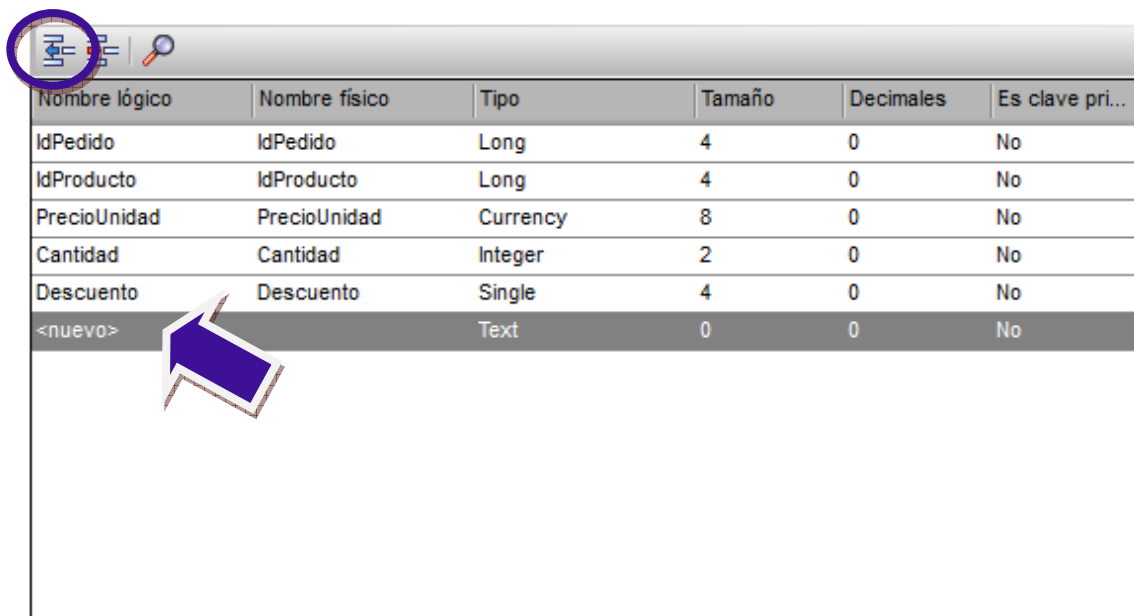


Tendremos la definición de la entidad seleccionada:

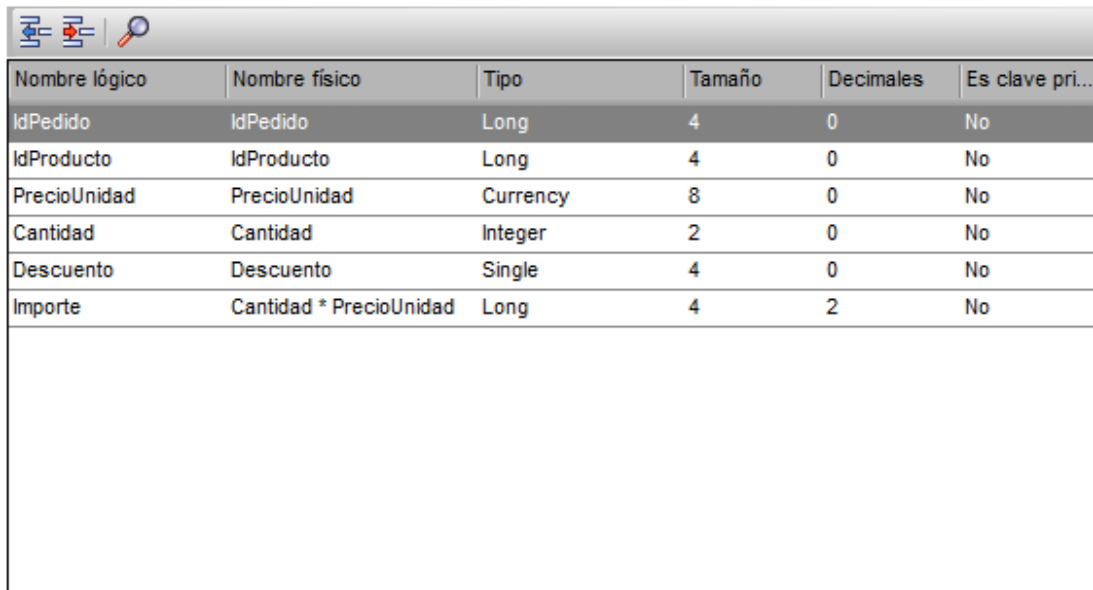


Mediante el botón  (Nuevo Atributo), podemos crear nuevos campos que podrán ser utilizados en las consultas SQL que posteriormente utilicen esta tabla.

Por ejemplo vamos a añadir un campo en la entidad Detalles de Pedidos llamado Importe como multiplicación de la cantidad y precio por unidad. Aunque la Base de Datos real no dispone de dicho campo, desde la consulta SQL se puede usar de forma transparente.





Informar el nombre lógico, la expresión en el nombre físico y el tipo de datos con los siguientes indicados en la figura. Aunque la Base de Datos real no dispone de dicho campo, desde la consulta SQL se puede usar de forma transparente.



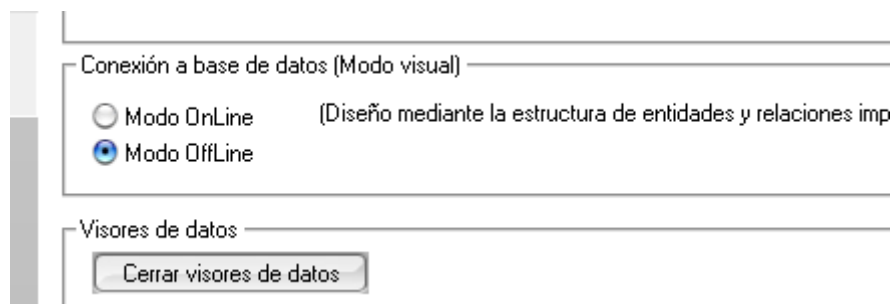
| Nombre lógico | Nombre físico | Tipo | Tamaño | Decimales | Es clave pri... |
|---------------|-------------------------|----------|--------|-----------|-----------------|
| IdPedido | IdPedido | Long | 4 | 0 | No |
| IdProducto | IdProducto | Long | 4 | 0 | No |
| PrecioUnidad | PrecioUnidad | Currency | 8 | 0 | No |
| Cantidad | Cantidad | Integer | 2 | 0 | No |
| Descuento | Descuento | Single | 4 | 0 | No |
| Importe | Cantidad * PrecioUnidad | Long | 4 | 2 | No |



Importante: Este nuevo campo solo se podrá utilizar en el editor de consultas tradicional y no en el nuevo editor de Modo visual.

Finalmente, grabaremos la entidad  y saldremos 

Ahora ya tenemos la definición de las entidades en DC Reporting, por lo que podemos volver a la creación de una consulta en modo OffLine.



Conexión a base de datos (Modo visual)

Modo OnLine (Diseño mediante la estructura de entidades y relaciones imp)

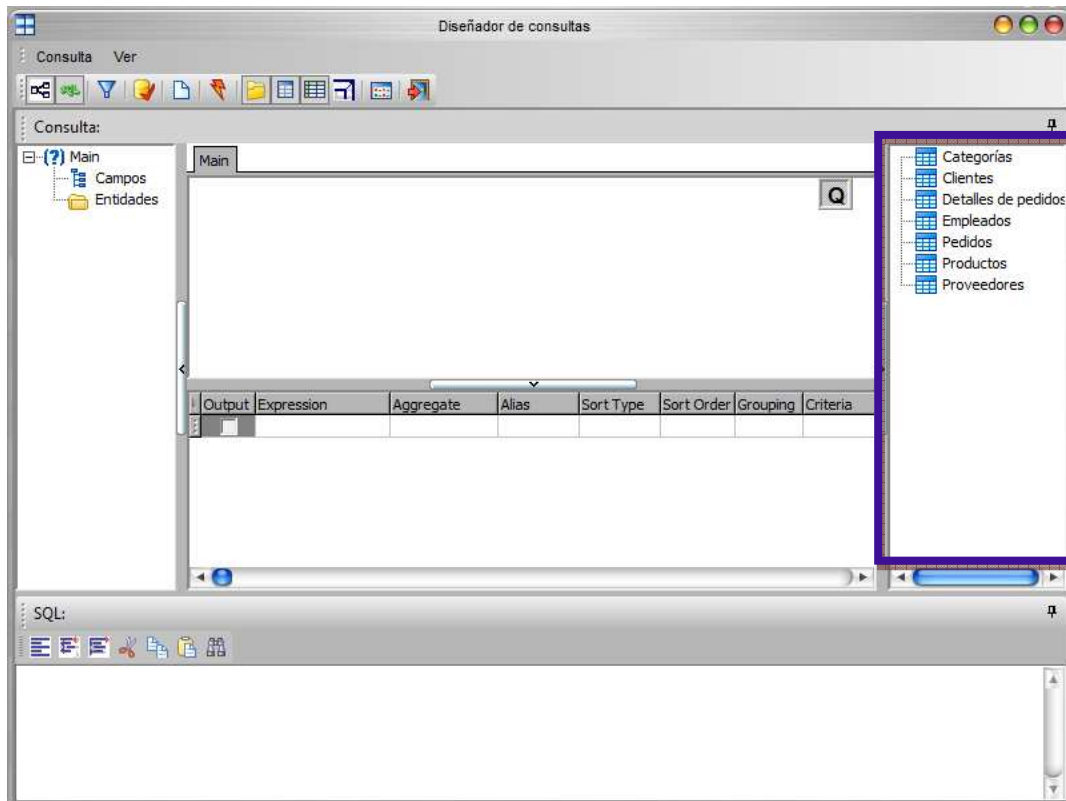
Modo OffLine

Visores de datos

Cerrar visores de datos

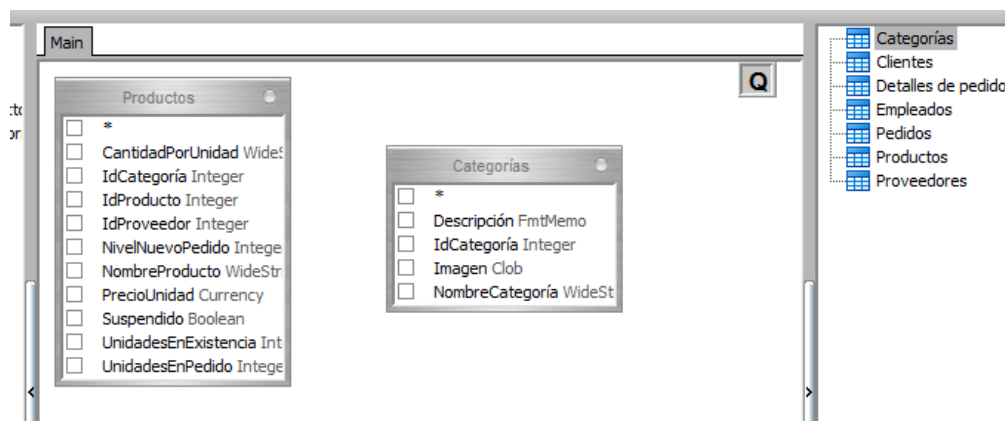
Editamos en **Modo Visual**.

Podemos observar que las entidades que hemos importado nos aparecen en el panel de esquema de Base de Datos:



Recordamos lo que queremos obtener con nuestra consulta, los productos y su categoría que están en pedidos y no hay suficiente unidades en stock.

Arrastramos las entidades Productos y categorías desde el panel de esquema base de datos al panel entidad-relación:

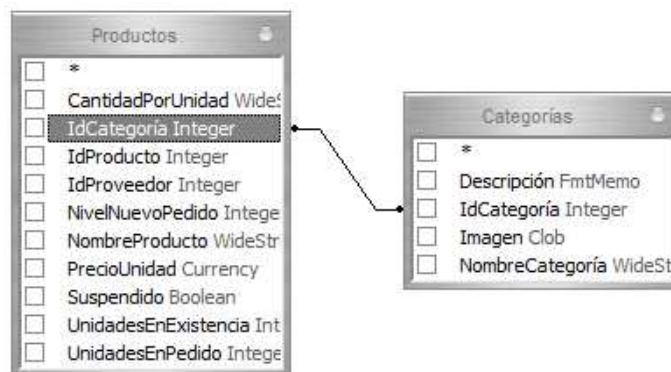


Podemos observar que el editor no ha relacionado las dos entidades, esto es debido a que en modo Offline el editor no está conectado a la BBDD para obtener las relaciones. Cuando anteriormente hemos importado las entidades solo hemos obtenido la estructura y no las relaciones que hay definidas entre las diferentes entidades.



Importante: En modo Online el editor visual tampoco relacionará las entidades si en la base de datos no están definidas estas.

Nosotros podemos relacionar manualmente estas dos tablas. Para ello identificamos el campo que relaciona ambas tablas, en este caso podemos ver que este campo es **IdCategorías**. Hacemos clic con el ratón sobre el campo en la **IdCategorías** entidad **Producto** y sin soltar arrastramos sobre el campo **IdCategorías** de la entidad **Categorías**.



Seleccionamos ahora los campos que nos interesan, en nuestro caso de la entidad **Categorías** el **NombreCategoría** y de **Productos** los campos **NombreProducto**, **UnidadesEnPedido** y **UnidadesEnExistencia**.

The screenshot shows the 'Diseñador de consultas' (Query Designer) interface. The 'Main' window displays the 'Productos' and 'Categorías' entities. The 'Output' table is populated with the following fields:

| Output | Expression | Aggregate | Alias | Sort.Type | Sort Order | Grouping | Criteria | Or... |
|-------------------------------------|--------------------------------|-----------|-------|-----------|------------|--------------------------|----------|-------|
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | | | | <input type="checkbox"/> | | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | | | | <input type="checkbox"/> | | |
| <input checked="" type="checkbox"/> | Productos.UnidadesEnExistencia | | | | | <input type="checkbox"/> | | |
| <input checked="" type="checkbox"/> | Productos.UnidadesEnPedido | | | | | <input type="checkbox"/> | | |

The 'SQL' window shows the following query:

```
Select Categorías.NombreCategoría, Productos.NombreProducto, Productos.UnidadesEnExistencia, Productos.UnidadesEnPedido
From Productos Inner Join Categorías On Productos.IdCategoría = Categorías.IdCategoría
```

Vamos a ejecutar la consulta para ver lo que obtenemos:

| NombreCategoría | NombreProducto | UnidadesEnExistencia | UnidadesEnPedido |
|-----------------|-------------------------------|----------------------|------------------|
| Bebidas | Té Dharamsala | 39 | 0 |
| Bebidas | Cerveza tibetana Barley | 17 | 40 |
| Bebidas | Refresco Guaraná Fantástica | 20 | 0 |
| Bebidas | Cerveza Sasquatch | 111 | 0 |
| Bebidas | Cerveza negra Steeleye | 20 | 0 |
| Bebidas | Vino Côte de Blaye | 17 | 0 |
| Bebidas | Licor verde Chartreuse | 69 | 0 |
| Bebidas | Calé de Malasia | 17 | 10 |
| Bebidas | Cerveza Laughing Lumberjack | 52 | 0 |
| Bebidas | Cerveza Outback | 15 | 10 |
| Bebidas | Cerveza Klosterbier Rhönbräu | 125 | 0 |
| Bebidas | Licor Cloudberry | 57 | 0 |
| Condimentos | Sirope de regaliz | 13 | 70 |
| Condimentos | Especias Cajun del chef Anton | 53 | 0 |

En el visor de datos podemos apreciar que hemos obtenido toda la lista de productos con la categoría, las unidades en pedido y las unidades que tenemos en existencias.

El siguiente paso será filtrar esta lista para que nos devuelva los productos que en existencias no tengan las unidades suficientes para servir el pedido. Es decir seleccionar aquellos productos que las unidades de pedido sean superior a las unidades en existencias.

Para poner un filtro en nuestra consulta nos dirigiremos al panel de columnas y en el campo **Productos.UnidadesEnPedido** hacemos clic con ratón en la celda de Criterio (para poder editarla) y escribimos **>Productos.UnidadesEnExistencia**.

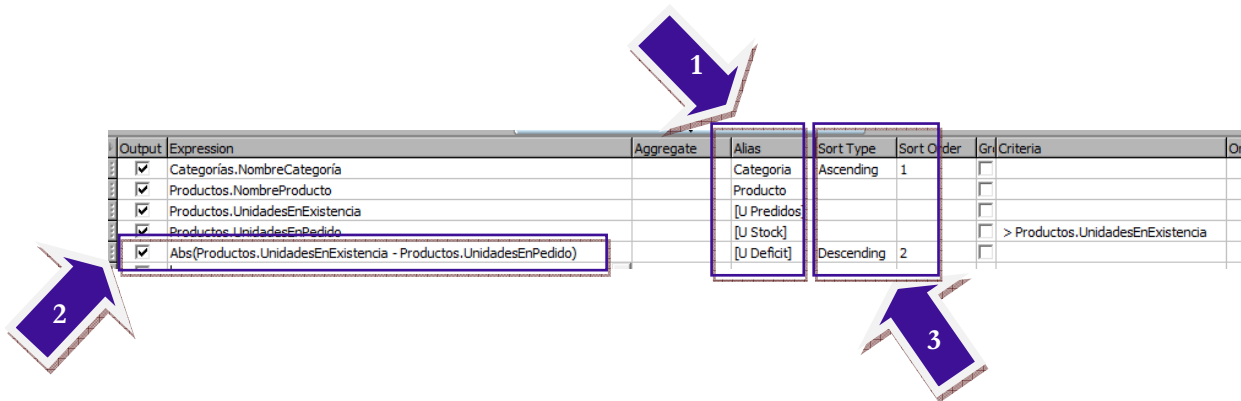
| Output | Expression | Aggregate | Alias | Sort Type | Sort Order | Grouping | Criteria | Or... |
|-------------------------------------|--------------------------------|-----------|-------|-----------|------------|--------------------------|----------------------------------|-------|
| <input checked="" type="checkbox"/> | Categorías.NombreCategoría | | | | | <input type="checkbox"/> | | |
| <input checked="" type="checkbox"/> | Productos.NombreProducto | | | | | <input type="checkbox"/> | | |
| <input checked="" type="checkbox"/> | Productos.UnidadesEnExistencia | | | | | <input type="checkbox"/> | | |
| <input checked="" type="checkbox"/> | Productos.UnidadesEnPedido | | | | | <input type="checkbox"/> | > Productos.UnidadesEnExistencia | |
| <input type="checkbox"/> | | | | | | <input type="checkbox"/> | | |

Si ejecutamos de nuevo la consulta, podemos apreciar que el resultado es el deseado, solo nos devuelve los productos que estando en pedidos no vamos a poder servir por falta de unidades en existencias.

| NombreCategoría | NombreProducto | UnidadesEnExistencia | UnidadesEnPedido |
|-----------------|-------------------------------|----------------------|------------------|
| Bebidas | Cerveza tibetana Barley | 17 | 40 |
| Condimentos | Sirope de regaliz | 13 | 70 |
| Lácteos | Queso Cabrales | 22 | 30 |
| Repostería | Bollos de Sir Rodney's | 3 | 40 |
| Lácteos | Queso gorgonzola Telino | 0 | 70 |
| Lácteos | Queso Mascarpone Fabioli | 9 | 40 |
| Pescado/Marisco | Salmón ahumado Gravad | 11 | 50 |
| Pescado/Marisco | Arenque ahumado | 5 | 70 |
| Repostería | Chocolate holandés | 15 | 70 |
| Repostería | Regaliz | 10 | 60 |
| Granos/Cereales | Bollos de pan de Wimmer | 22 | 80 |
| Condimentos | Especias picantes de Luisiana | 4 | 100 |
| Repostería | Barra de pan de Escocia | 6 | 10 |
| Frutas/Verduras | Queso de soja Longlife | 4 | 20 |

Cerramos el visor de datos y terminamos de formatear nuestra consulta:

1. Añadimos alias a los campos.
2. Añadimos un nuevo campo que nos dé el número de unidades que nos faltan, utilizamos la función **Abs()** para obtener el valor absoluto.
3. Ordenamos los registros por Categorías y por el nuevo campo.




Si volvemos a ejecutar la consulta vemos que el resultado es el deseado. La consulta devuelve solamente los registros que tienen un déficit de unidades (diferencia entre unidades en pedidos y unidades en stock), ordenados por categorías y entre estas de mayor a menor déficit de unidades.

| Categoría | Producto | U Predidos | U Stock | U Deficit |
|-----------------|-------------------------------|------------|---------|-----------|
| Bebidas | Cerveza tibetana Barley | 17 | 40 | 23 |
| Condimentos | Especias picantes de Luisiana | 4 | 100 | 96 |
| Condimentos | Sirope de regaliz | 13 | 70 | 57 |
| Frutas/Verduras | Queso de soja Longlife | 4 | 20 | 16 |
| Granos/Cereales | Bollos de pan de Wimmer | 22 | 80 | 58 |
| Lácteos | Queso gorgonzola Telino | 0 | 70 | 70 |
| Lácteos | Queso Mascarpone Fabioli | 9 | 40 | 31 |
| Lácteos | Queso Cabrales | 22 | 30 | 8 |
| Pescado/Marisco | Arenque ahumado | 5 | 70 | 65 |
| Pescado/Marisco | Salmón ahumado Gravad | 11 | 50 | 39 |
| Repostería | Chocolate holandés | 15 | 70 | 55 |
| Repostería | Regaliz | 10 | 60 | 50 |
| Repostería | Bollos de Sir Rodney's | 3 | 40 | 37 |
| Repostería | Barras de pan de Escocia | 6 | 10 | 4 |

Registros cargados: 14 N° registros leídos: 14 Registro actual: 0 T. conexión: 00:00:00

Cerramos el visor de datos, salimos del editor de consultas y grabamos la consulta.

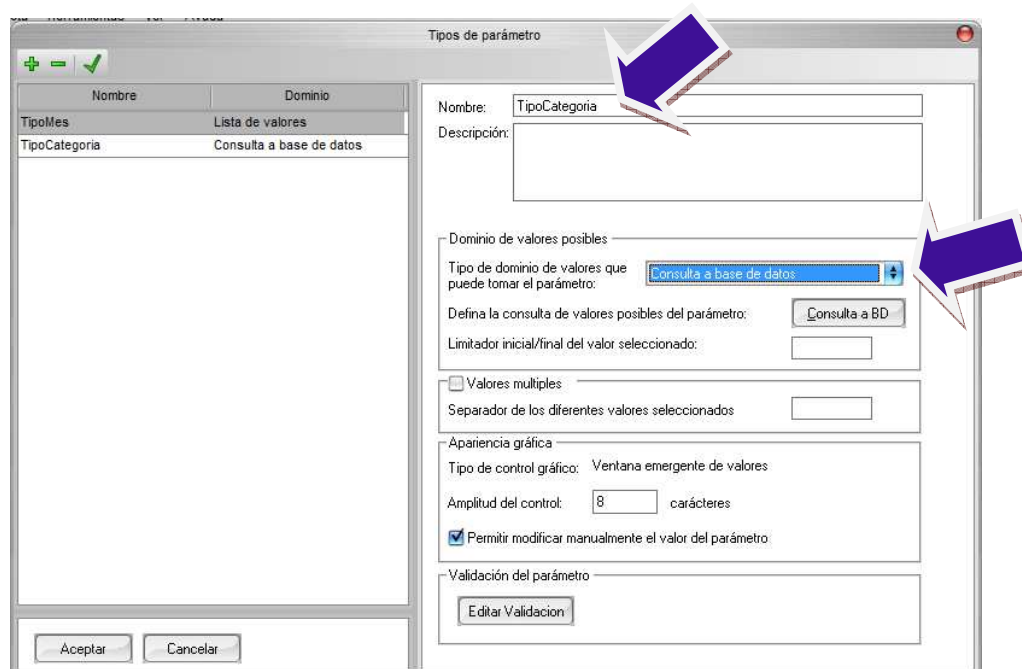
 En la guía de referencia de DC Reporting se puede consultar las diferentes funcionalidades del editor de consultas.

Añadir Parámetro tipo Consulta de Base de Datos

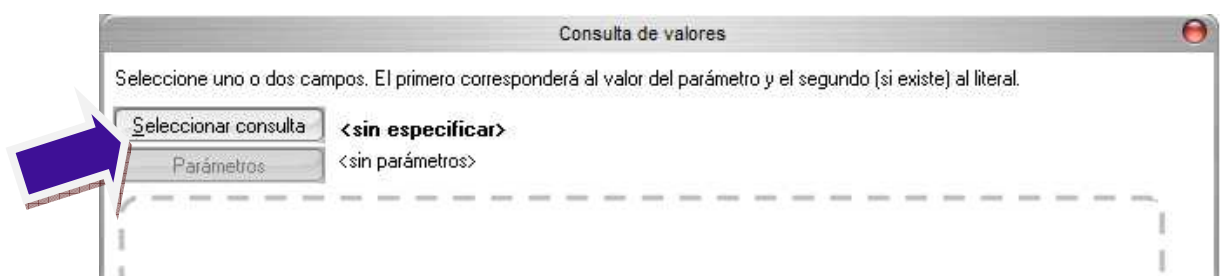
Vamos ahora a añadir un parámetro a la consulta que acabamos de hacer, para que nos de los productos que estén en déficit de unidades pero de las categorías que nosotros le indiquemos.

Como la lista de Categorías puede variar en el tiempo no podemos utilizar un parámetro de tipo lista de valores, ya que este lo tendríamos que ir cambiando cada vez que el maestro de categorías cambiara. Para evitar esto en DC- Reporting existe un tipo de parámetro que te permite obtener la lista de valores de una consulta de base de datos. Utilizaremos este tipo de parámetro, pero antes de crear lo necesitaremos crear una consulta sencilla que devuelva todas las categorías que tenemos. Concretamente los campos **IdCategoría** y **NombreCategoría**, le llamaremos **Lista de Categorías**.

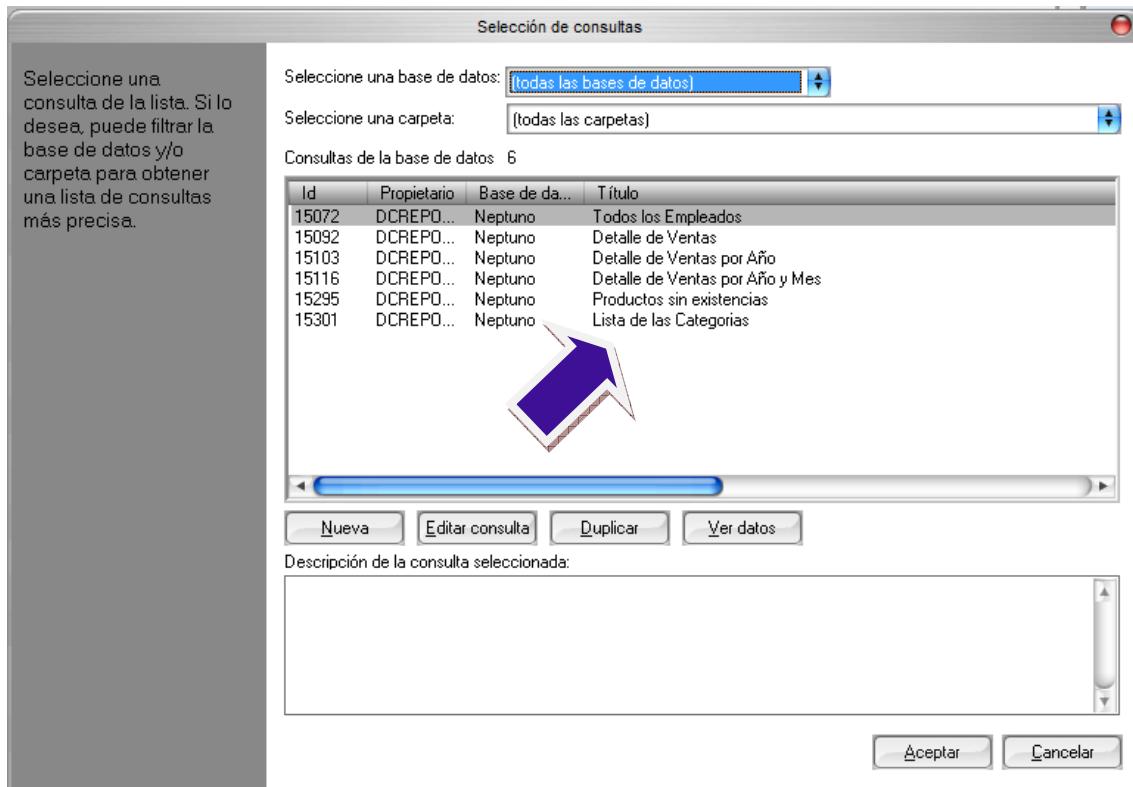
Una vez que tengamos creada la consulta Lista de categorías vamos a crear el nuevo tipo de parámetro. Igual que hicimos al crear el **TipoMes** iremos al administrador de tipos de parámetros: **Menú Proyecto** → **Tipos de parámetros del proyecto actual**. Añadimos un nuevo tipo que le llamaremos **TipoCategorías** y en la lista de dominios erigiremos **Consulta de base de datos**.







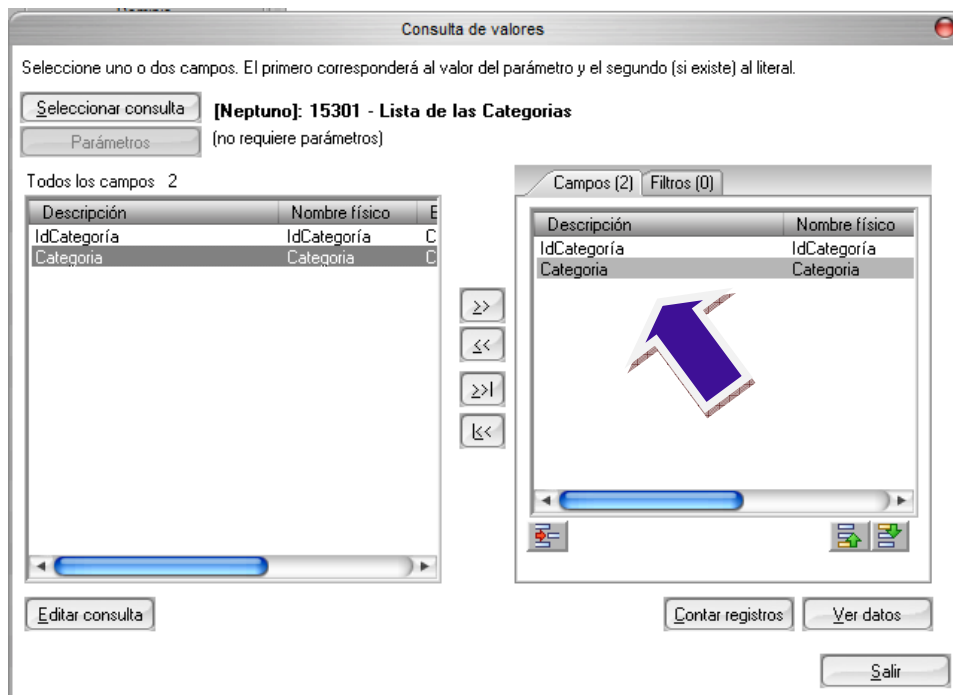
A continuación pulsamos el botón **Consulta a BD**. Nos saldrá una nueva ventana, donde pulsaremos el botón **Seleccionar consulta**.



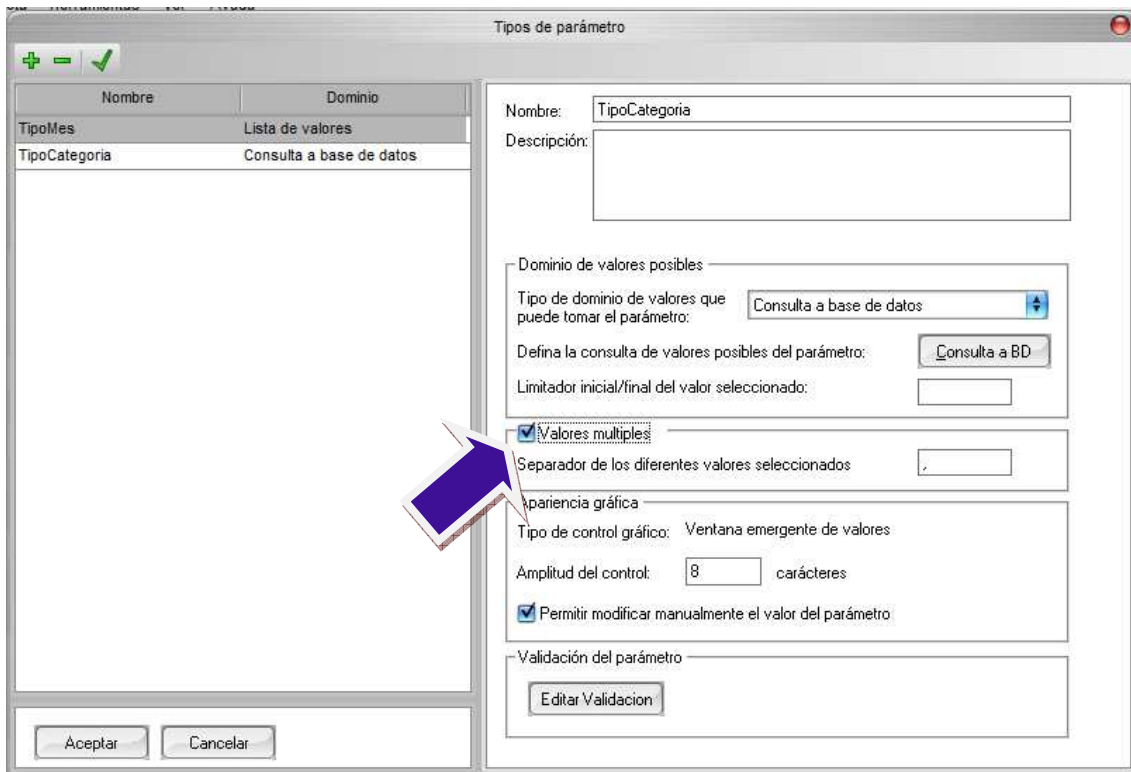
Nos muestra una ventana donde podemos seleccionar la consulta que nos va devolver los datos de la lista de valores. Seleccionamos la consulta **Lista de Categorías**. Y pulsamos **Aceptar**.



Una vez seleccionada la consulta tenemos que seleccionar que campos vamos a necesitar de la consulta. En nuestro caso **IdCategorías** y **NombreCategoría**. Para seleccionar los campo hemos de pasar los campos de la parte izquierda de la ventana a la parte izquierda de la ventana, utilizando los botones , ,  y .

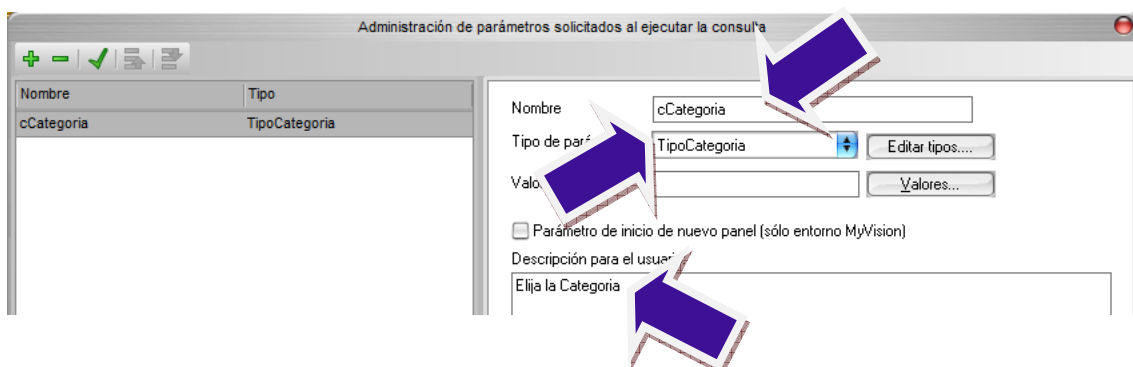


Una vez que tengamos seleccionado los campos de la consulta pulsamos Salir. Volviendo a la ventana de administrador de **Tipos de Parámetros**. Finalmente activaremos la casilla de **Valores múltiples**, para permitir elegir más de un valor. Y aceptamos para terminar de crear el tipo de parámetro **TipoCategorias**.

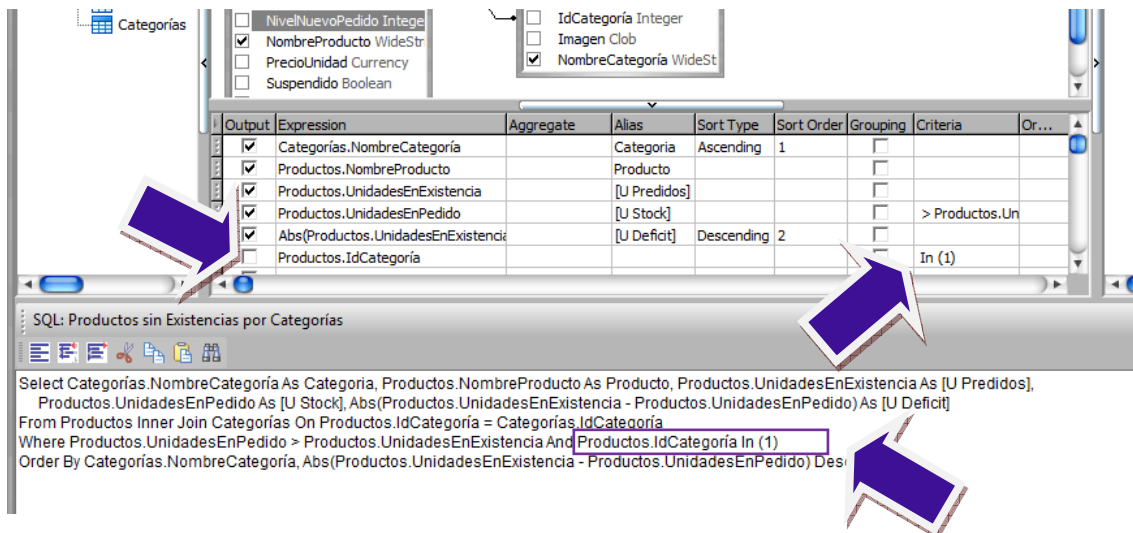


Duplicamos la consulta **Productos sin Existencias**, la editamos, le cambiamos el nombre por **Productos sin Existencias por categorías**. Nos vamos al editor de parámetros de la consulta para añadirle un nuevo parámetro.

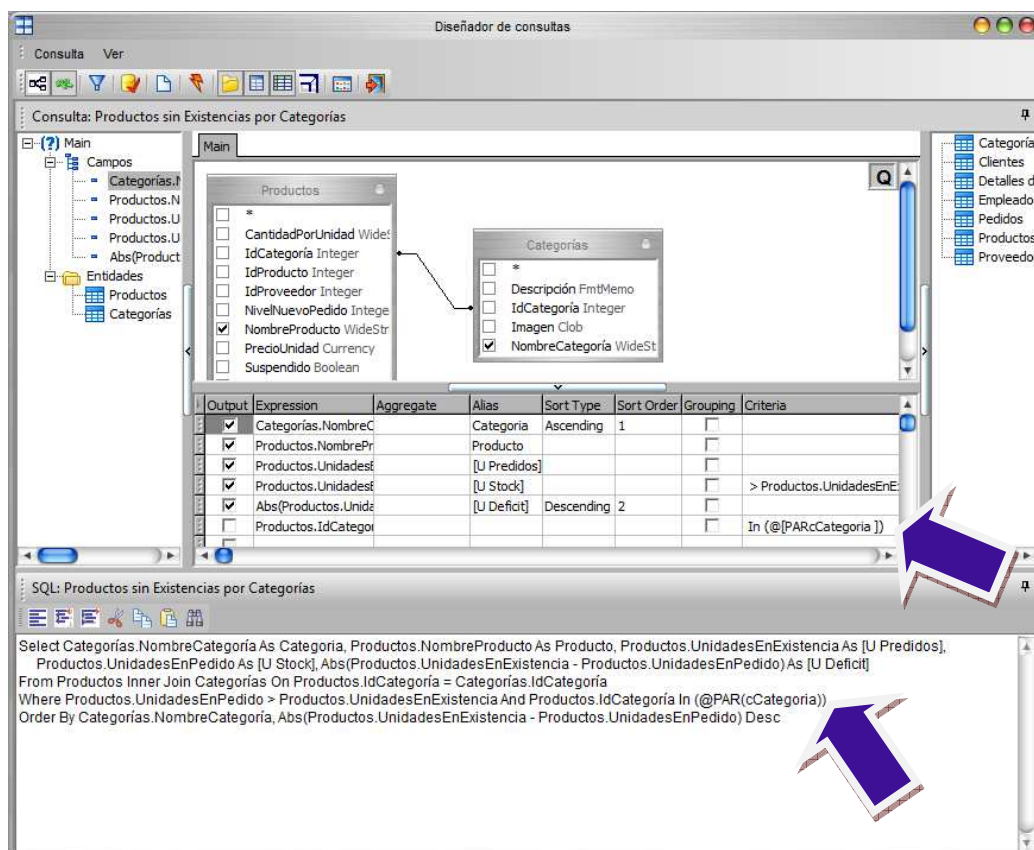
Al nuevo parámetro le llamaremos **cCategorías** y en la lista desplegable de **Tipo de parámetros** elegimos **TipoCategorías** (que es el tipo que hemos creado anteriormente).



Aceptamos y editamos la consulta con el **editor Modo Visual**. Como lo que queremos es filtrar las categorías vamos a añadir el campo **IdCategorías** haciendo doble clic con el ratón en el campo **IdCategoría** de la tabla **Categorías**. En el panel de columnas se habrá añadido el nuevo campo, dejaremos la casilla de **Salida** sin marcar (no queremos que nos devuelva el **idCategorías**), en la columna **Criterio** introducimos el siguiente texto **IN (1)**. En el panel de texto SQL se habrá añadido una nueva condición en la sentencia **Where**.



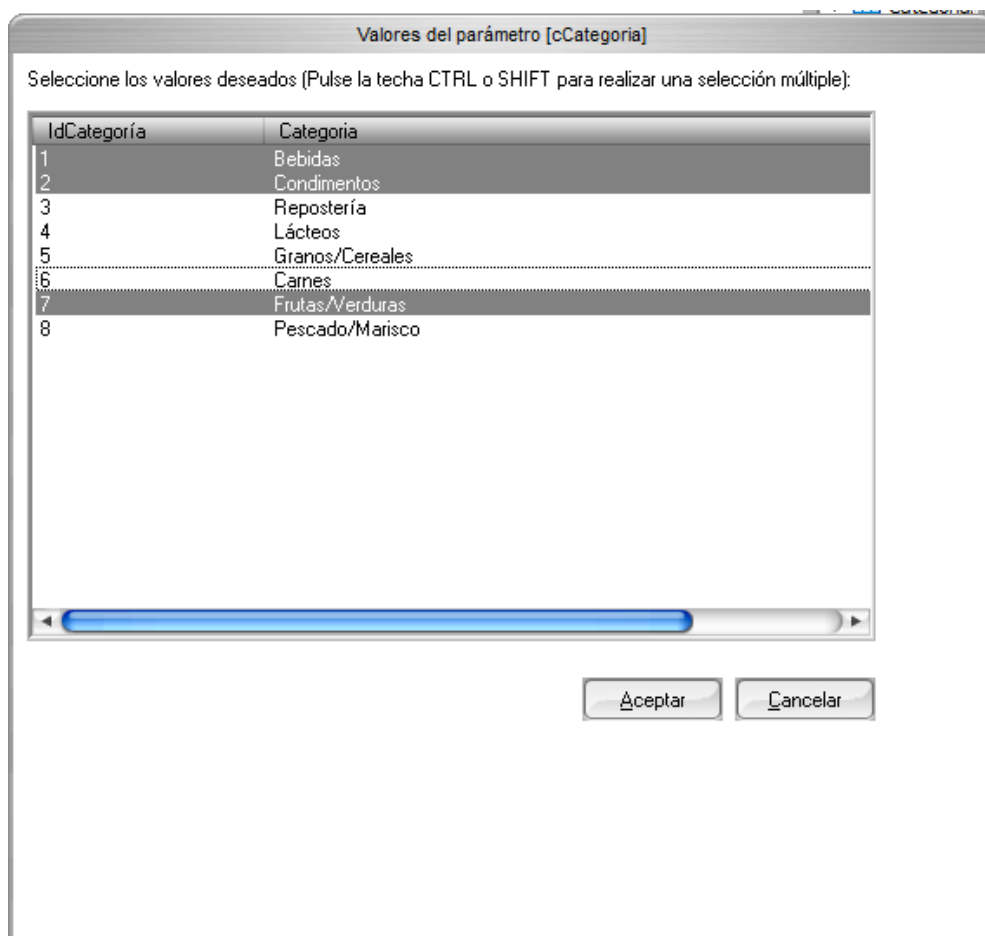
Nos situamos en el editor de Texto SQL, borramos el **1** de **IN(1)** y con el botón derecho del ratón entre los paréntesis insertamos el parámetro **Categorías**. Salimos del editor de texto SQL haciendo clic con el ratón en otro panel y el resultado ha de ser:



Salimos del editor, guardamos la consulta. Al ejecutar la consulta nos saldrá la ventana de solicitud de valores del parámetro **cCategoria**:



Si pulsamos el botón Valores, nos mostrará la lista de valores disponibles a seleccionar. Podemos seleccionar más de un valor manteniendo pulsada la tecla Ctrl.



Una vez seleccionados los valores aceptamos. Podemos ver que se han seleccionado los códigos de las categorías seleccionadas separados por comas.

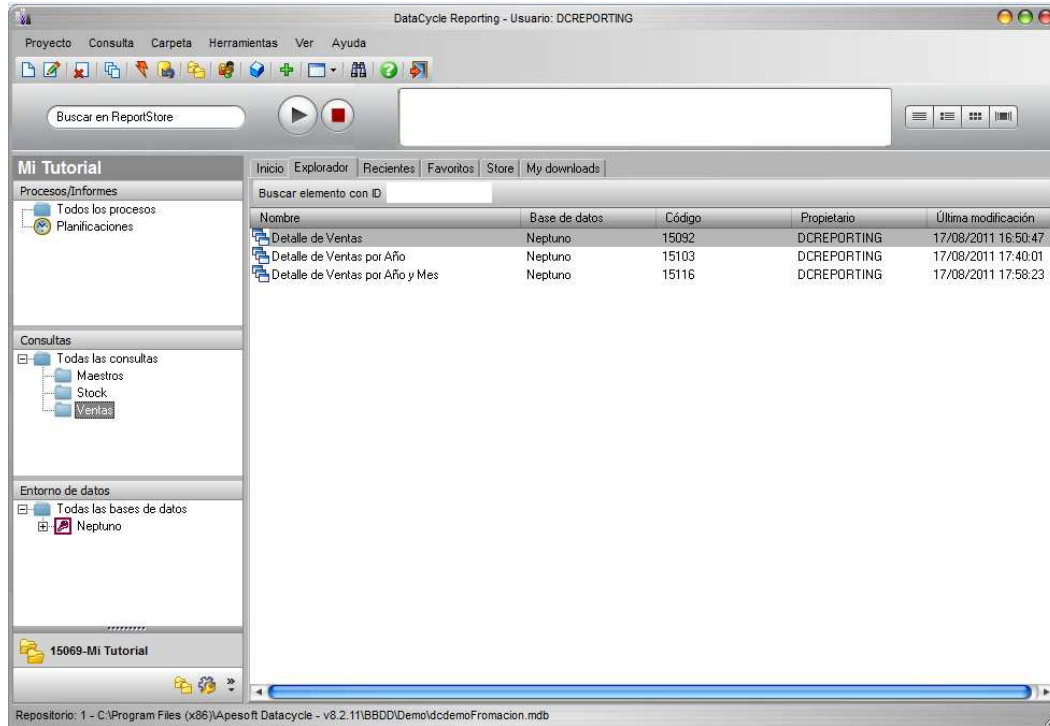


Aceptamos y vemos en el resultado que solo nos muestra los productos de las categorías solicitadas.

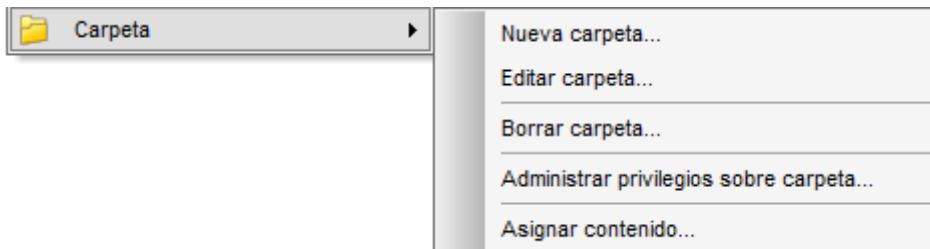
| Categoría | Producto | U Predidos | U Stock | U Deficit |
|-----------------|-------------------------------|------------|---------|-----------|
| Bebidas | Cerveza tibetana Barley | 17 | 40 | 23 |
| Condimentos | Espicias picantes de Luisiana | 4 | 100 | 96 |
| Condimentos | Sirope de regaliz | 13 | 70 | 57 |
| Frutas/Verduras | Queso de soja Longlife | 4 | 20 | 16 |

Gestión de carpetas

Una de las capacidades más agradecidas consiste en la posibilidad de clasificar las consultas y procesos en carpetas. Puede definir la jerarquía de procesos y carpetas según sus intereses.



Varias operaciones le permiten gestionar las carpetas y su contenido:



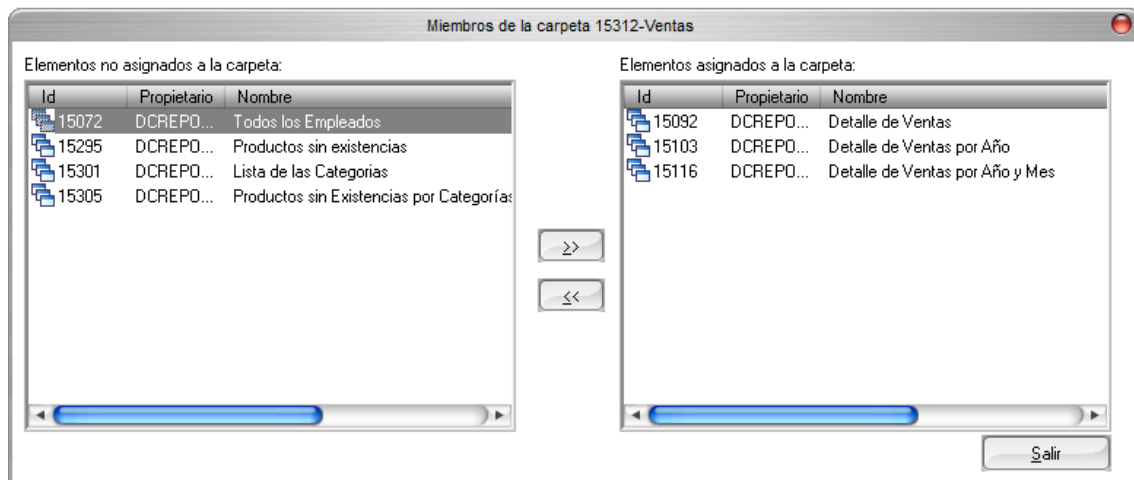
- Crear nueva carpeta
- Editar una carpeta existente
- Borrar carpeta
- Privilegios sobre carpeta
- Asignar contenido a carpeta

Creación, edición y eliminación de carpetas

Estas opciones le permitirán crear nuevas carpetas, cambiarle el nombre y eliminarlas. Recuerde que no se eliminan las consultas o procesos asociadas a una carpeta, tan sólo se elimina esta.

Clasificación de consultas y procesos

Podemos asociar consultas y procesos a una carpeta mediante el siguiente cuadro de diálogo:



Para realizar asignaciones de muchas consultas podemos realizar selecciones múltiples (utilizando las teclas Control o Mayúsculas). Además podemos ordenar las consultas por código, propietario o nombre.



Una consulta puede estar asignada a múltiples carpetas simultáneamente.

Ejercicios

Una vez que hemos visto como relacionar varias tablas desde el asistente de consultas, vamos a realizar los siguientes ejercicios.

Consultando múltiples tablas

Inner Join

Permite unir tablas mediante los campos que forman la clave principal en una tabla y foránea en la otra. El Inner Join solo devuelve resultados en caso de que un registro de una tabla sea referenciado desde la otra.

Por ejemplo, si tenemos una Base de Datos universitaria con alumnos y asignaturas, pero alguno de los alumnos está matriculado pero aún no está apuntado a ninguna asignatura, o bien la asignatura existe pero nadie se ha apuntado a ella, no aparecerían dichos registros en caso de consultas en las que se relacionasen alumnos y asignaturas.

Ejemplo 03-01

Quiero ver los productos y el nombre de su categoría

```
SELECT Productos.NombreProducto, Categorías.NombreCategoría
FROM [Categorías], [Productos]
WHERE Productos.IdCategoría = Categorías.IdCategoría
Order By Categorías.NombreCategoría
```

Nota: al haber dos tablas es necesario indicar a que tabla pertenecen los campos en caso de ambigüedad

Nota:

Ejemplo 03-02

Necesito conocer todos los pedidos y sus líneas con el nombre del cliente y el nombre de los productos

```
SELECT Pedidos.IdPedido, Clientes.NombreCompañía, Pedidos.FechaPedido,
Pedidos.FechaEntrega, Productos.NombreProducto,
[Detalles de pedidos].PrecioUnidad, [Detalles de pedidos].Cantidad,
[Detalles de pedidos].Cantidad * [Detalles de pedidos].PrecioUnidad
FROM [Pedidos], [Detalles de pedidos], [Productos], [Clientes]
WHERE [Detalles de pedidos].IdProducto = Productos.IdProducto AND
Pedidos.IdPedido = [Detalles de pedidos].IdPedido AND
Pedidos.IdCliente = Clientes.IdCliente
Order By [Detalles de pedidos].IdPedido, [Detalles de pedidos].IdProducto
```


Outer Join

De la misma forma que el Inner Join, relaciona tablas mediante sus claves principal en una tabla y foránea en la otra. Como diferencia con el Inner Join, en este caso, no se discriminan aquellos registros cuya clave de join aún no haya sido informada o para los que no existan registros relacionados en la otra tabla.

Ejemplo 03-04

Que productos nunca han sido pedidos

```
SELECT Productos.NombreProducto
FROM [Productos] left join [Detalles de pedidos]
    on Productos.IdProducto = [Detalles de pedidos].IdProducto
WHERE isnull([Detalles de pedidos].IdPedido)
```

Ejemplo 03-05

Vendedores que aún no han hecho honor a su cargo

```
SELECT Empleados.Nombre, Empleados.Apellidos
FROM [Empleados] left join [Pedidos]
    on Empleados.IdEmpleado = Pedidos.IdEmpleado
WHERE isnull(idpedido)
```

Sumarizando y agrupando los datos

Ejemplo 04-01

Quiero saber cuantos empleados tiene mi empresa

```
SELECT count(*) as TotalEmpleados
FROM [Empleados]
```

Nota: Si quiero saber cuantos de ellos son del Reino Unido:

```
SELECT count(*) as TotalEmpleados
FROM [Empleados]
WHERE [Empleados].País = 'Reino Unido'
```

Ejemplo 04-02

Cual es el mayor monto de una línea de pedido

```
SELECT max([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad) as MayorLineaPedido
FROM [Detalles de pedidos]
```

Ejemplo 04-03

Cuanto valen los pedidos realizados de cada producto

```
SELECT [Detalles de pedidos].IdProducto as IdProducto,
    sum([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad) as ValorPedidos
FROM [Detalles de pedidos]
GROUP BY [Detalles de pedidos].IdProducto
```

Ejemplo 04-04

Para valorar las comisiones que debemos abonar a nuestros vendedores por las ventas del mes actual tenemos la siguiente consulta. La comisión representa el 15% de sus ventas

```
SELECT Empleados.Tratamiento, Empleados.Nombre, Empleados.Apellidos,
       sum([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad),
       sum([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad) * 0.15
FROM [Empleados], [Pedidos], [Detalles de pedidos]
WHERE Pedidos.IdEmpleado = Empleados.IdEmpleado AND
      Pedidos.IdPedido = [Detalles de pedidos].IdPedido and
      month(Pedidos.FechaPedido)=month(now()) and
      year(pedidos.fechapedido)=year(now())
Group By Empleados.Tratamiento, Empleados.Nombre, Empleados.Apellidos
Order By Empleados.Apellidos
```

Nota1: porque no aparece ningún valor?

Nota2: vamos a preguntar por las ventas de agosto de 2005:

```
Sustituimos:
month(Pedidos.FechaPedido)=month(now()) and year(pedidos.fechapedido)=year(now())
Por:
month(Pedidos.FechaPedido)=8 and year(pedidos.fechapedido)=2004
```

Filtrar datos agrupados

Hasta ahora habíamos aplicado un criterio a los registros de forma individual. Ahora vamos a aplicar un criterio al resultado de la agrupación de varios registros.

Ejemplo 04-06

Conocer cuáles son los clientes que compran un importe bajo de productos de una categoría:

```
SELECT
    Categorías.NombreCategoría,
    Clientes.NombreCompañía,
    sum([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad ) as VentasTotales
FROM
    Categorías, Clientes, Pedidos, Productos, [Detalles de pedidos]
WHERE
    [Detalles de pedidos].IdProducto = Productos.IdProducto AND
    Pedidos.IdCliente = Clientes.IdCliente AND
    Pedidos.IdPedido = [Detalles de pedidos].IdPedido AND
    Productos.IdCategoría = Categorías.IdCategoría
GROUP BY
    Categorías.NombreCategoría, Clientes.NombreCompañía
having
    sum([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad ) < 3000
```

Union

Los conjuntos que forman parte de la Union deben contener el mismo número de campos y del mismo tipo.

Ejemplo 04-07

Nombre y dirección de todos los clientes y empleados ordenados por código postal

```
SELECT DISTINCT Clientes.NombreCompañía, Clientes.País, Clientes.Ciudad, clientes.códpostal
FROM [Clientes]
union
SELECT DISTINCT Empleados.Nombre & " " & empleados.apellidos, empleados.País, empleados.Ciudad,
empleados.códpostal
FROM [Empleados]
order by códpostal
```

Subconsultas

Ejemplos 05-01

Vendedores que han hecho alguna venta a clientes de Italia

```
SELECT DISTINCT
  Empleados.Apellidos as Apellidos
FROM
  empleados , pedidos
WHERE
  pedidos.idempleado = empleados.idempleado and pedidos.idcliente in
  (SELECT DISTINCT
    Pedidos.IdCliente
  FROM
    Pedidos, Clientes
  WHERE
    Pedidos.IdCliente = Clientes.IdCliente and Clientes.País = "Italia" )
```

Ejemplo 05-02

Proveedores de los 20 productos más vendidos durante este mes.

Igual que en los ejemplos anteriores, sustituimos el mes y año actual por octubre de 2005 para obtener datos válidos.

```
SELECT DISTINCT proveedores.nombrecompañía
FROM (
  SELECT TOP 20
    productos.idproducto, Productos.NombreProducto as prod, productos.idproveedor, sum([Detalles
de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad ) as imp
  FROM
    [Productos], [Detalles de pedidos], [Pedidos]
  WHERE
    [Detalles de pedidos].IdProducto = Productos.IdProducto AND Pedidos.IdPedido = [Detalles de
pedidos].IdPedido AND year(Pedidos.FechaPedido) = 2005 and month(pedidos.fechapedido)=10
  GROUP BY
    productos.idproducto,Productos.NombreProducto, productos.idproveedor
  ORDER BY
    sum([Detalles de pedidos].Cantidad*[Detalles de pedidos].PrecioUnidad ) desc ) as
  ProductosMasVendidos, Proveedores
WHERE ProductosMasVendidos.IdProveedor = Proveedores.IdProveedor
```

Ejemplo 05-03

Productos que son más caros que cualquiera de la categoría de Mariscos

```
SELECT Productos.NombreProducto, productos.precioUnidad
FROM [Productos]
WHERE Productos.PrecioUnidad > all
  (select productos.precioUnidad
  from [productos],[categorías]
  where Productos.IdCategoría = Categorías.IdCategoría and
  [Categorías].NombreCategoría like "%marisc%")
```

Nota: si se quiere saber aquellos productos que son más caros que algún marisco se sustituye ALL por ANY

Ejemplo 05-04

Cientes que han comprado alguna repostería y algún queso

```
SELECT Clientes.NombreCompañía, Clientes.Dirección, Clientes.Ciudad, Clientes.Región,  
       Clientes.CódPostal, Clientes.País, Clientes.Teléfono  
FROM [Clientes]  
WHERE exists  
  (SELECT *  
   FROM [Pedidos], [Detalles de pedidos], [Productos], [Categorías]  
   WHERE [Detalles de pedidos].IdProducto = Productos.IdProducto AND  
         Pedidos.IdPedido = [Detalles de pedidos].IdPedido AND  
         Productos.IdCategoría = Categorías.IdCategoría and  
         categorías.nombrecategoría like '%repostería%')  
and exists  
  (SELECT *  
   FROM [Pedidos], [Detalles de pedidos], [Productos]  
   WHERE [Detalles de pedidos].IdProducto = Productos.IdProducto AND  
         Pedidos.IdPedido = [Detalles de pedidos].IdPedido AND  
         productos.nombreproducto like '%queso%')
```

Ejemplo 05-05

Cientes que no han comprado nada durante el año 2005

```
SELECT Clientes.NombreCompañía  
FROM [Clientes]  
WHERE not exists  
  (select *  
   from [Pedidos]  
   where Pedidos.IdCliente = Clientes.IdCliente AND year(pedidos.fechapedido) = 2005)
```

Ejemplo 05-06

Lista de productos y la fecha del último pedido en el que aparece

```
SELECT Productos.NombreProducto,  
       (select max (pedidos.fechapedido)  
        from pedidos, [detalles de pedidos]  
        where [Detalles de pedidos].IdProducto = Productos.IdProducto AND  
              Pedidos.IdPedido = [Detalles de pedidos].IdPedido)  
From productos
```

Prácticas

Consultas simples

Quiero conocer el nombre de todos mis proveedores.

De que países son mis clientes.

Necesito toda la información disponible sobre las compañías de envío.

Deme la fecha de entrega de todos los pedidos y su destino ordenado por país y fecha más lejana a más cercana.

Quiero las ciudades por orden alfabético de mis proveedores con el nombre de todos ellos, también por orden alfabético.

Cuantos trienios tiene cada uno de mis empleados.

Filtrando los datos

Cuanto supone un descuento de un 12% en el pedido 10251.

Que proveedores no tienen Web.

Necesito conocer los pedidos del mes de agosto de 2005.

Que pedidos han sido entregados en 14 días.

Sumarizando y agrupando los datos

De cuantos países son mis clientes.

Cuál es el valor total de los pedidos mensualmente.

Lista de los empleados con la fecha del último pedido realizado. Esta consulta la suele realizar el departamento de RRHH.

Cuáles son nuestros 10 clientes más antiguos.

Cuantos productos se pidieron en el pedido 10252 y cuál es el importe total del mismo.

Que vendedor ha conseguido ventas superiores a 1000\$ durante el último mes.

Que categorías están un 20% por encima del resto en cuanto a importe facturado.

Consultando múltiples tablas

Quiero saber cuáles son los clientes que realizan pedidos de marisco de diferente tipo.

Vendedores que nunca han vendido bebidas y fecha de contratación del mismo.

Dime los clientes que han realizado pedidos de un mínimo de tres categorías diferentes durante el último mes.

Dime los clientes que han tenido trato con un mínimo de tres vendedores.

Vamos a recuperar a los clientes que no nos han hecho ninguna compra durante este año.

Quiero saber que compañías de envíos utilizo más en los pedidos al extranjero.

Quiero saber con los productos de que proveedor he factura más, he hecho más pedidos y he vendido más unidades.

Lista de los clientes que han comprado marisco o bien pedidos con destino a Bélgica.

Lista de mis empleados con los clientes que viven en la misma ciudad.

Subconsultas

Pedidos en el que todas las líneas sean de más de 10\$.

Clientes cuyo importe total de pedidos está por encima de la media en un 50%.

Productos que no han sido pedidos durante el último mes.

Vendedores que han hecho alguna venta a clientes de Italia.

Deseo ver los pedidos de los artículos que tengan pedidos por importe superior a.....

Categorías para cuyos productos ha habido más de 5 pedidos mensuales.

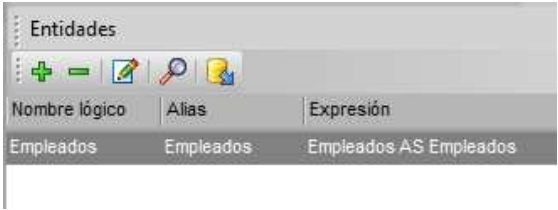
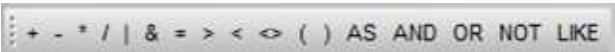
Vendedores que han vendido todos los productos alguna vez.

6

Anexos

Anexo: Consultas con asistente (Tradicional)

Veamos un poco con detalle las partes del **asistente de consultas**.

| | |
|---|---|
| <p>Esta sección es donde se definen las tablas que participan en la consulta. Podemos añadir nuevas tablas a la consulta, editar una tabla, ver los datos de una tabla, importar nuevas tablas y quitar tablas.</p>  | <p>Esta sección se usa para editar expresiones cuando especificamos el cálculo para un campo.</p>  |
| <p>Esta es la sección de campos de las tablas. Cuando tengamos seleccionada una tabla en la sección superior (la lista de tablas) aparecerán los campos de esa tabla. El objetivo es seleccionar los campos que tendrá la consulta y arrastrándolos a la pestaña del select.</p> | <p>Esta es la lista de campos y cálculos que formarán la consulta. Se estructura en varias pestañas:</p> <ul style="list-style-type: none">Select lista de camposFrom lista de tablas (rara vez lo tocaremos)Where condiciones y criterios de la consultaGroup By agrupaciones y roturasOrder By ordenacionesHaving condiciones para los agrupados |

| Nombre lógico | Nombre físico | Tipo |
|-------------------|-------------------|-----------------|
| IdEmpleado | IdEmpleado | Long (4) |
| Apellidos | Apellidos | Text (20) |
| Nombre | Nombre | Text (10) |
| Cargo | Cargo | Text (30) |
| Tratamiento | Tratamiento | Text (25) |
| FechaNacimiento | FechaNacimiento | Date / Time (8) |
| FechaContratación | FechaContratación | Date / Time (8) |
| Dirección | Dirección | Text (60) |
| Ciudad | Ciudad | Text (15) |
| Región | Región | Text (15) |
| CódPostal | CódPostal | Text (10) |
| País | País | Text (15) |
| TelDomicilio | TelDomicilio | Text (24) |
| Extensión | Extensión | Text (4) |

| Select |
|--|
| Empleados.IdEmpleado as IdEmpleado |
| Empleados.Apellidos as Apellidos |
| Empleados.Nombre as Nombre |
| Empleados.Cargo as Cargo |
| Empleados.Tratamiento as Tratamiento |
| Empleados.FechaNacimiento as FechaNacimiento |
| Empleados.FechaContratación as FechaContratacion |
| Empleados.Dirección as Direccion |
| Empleados.Ciudad as Ciudad |
| Empleados.Región as Region |
| Empleados.CódPostal as CodPostal |
| Empleados.País as Pais |
| Empleados.TelDomicilio as TelDomicilio |
| Empleados.Extensión as Extension |
| Empleados.Foto as Foto |
| Empleados.Notas as Notas |
| Empleados.Jefe as Jefe |
| Empleados.eMail as eMail |
| Empleados.IdDataCycle as IdDataCycle |

Select From * Where Group By Having Order By

Acciones de la consulta:



Ver Asistente.



Ver modo SQL.



Verificar sintaxis.



Limpiar, borra la consulta.



Ejecutar la consulta.

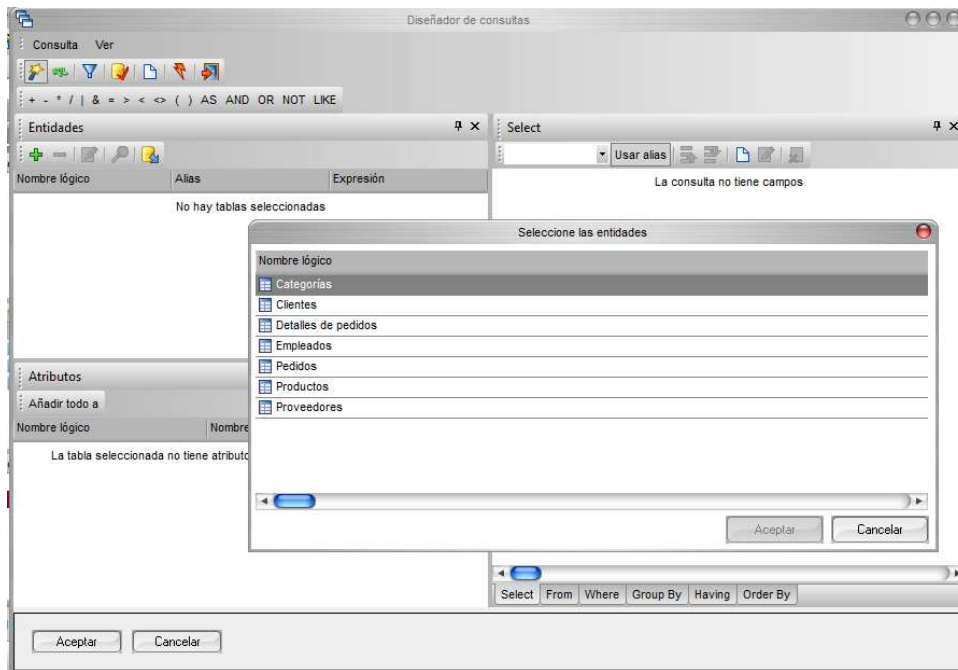


guardar la consulta y salir,

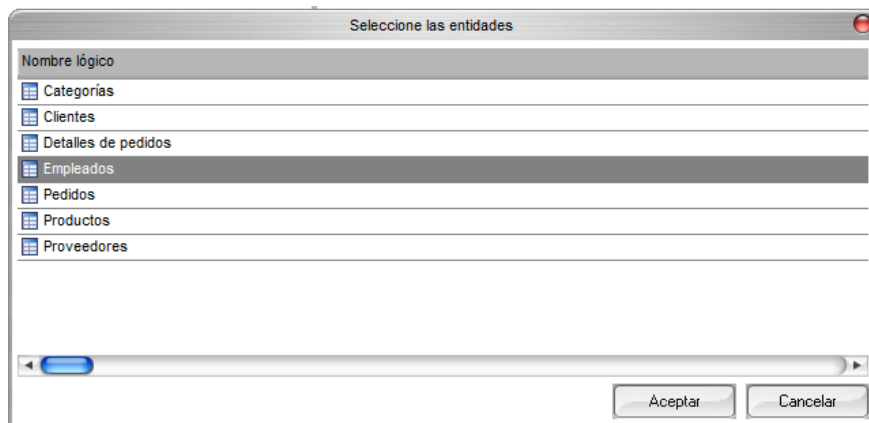


cancela los cambios.

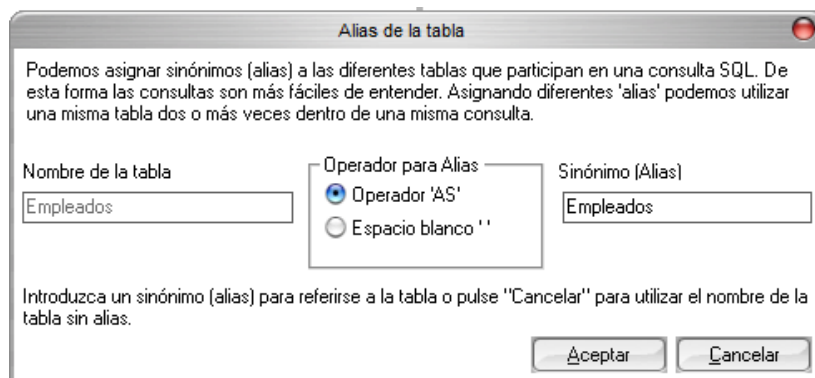
Aparece la siguiente ventana, donde ya se nos solicitan las tablas que deseamos consultar (también se pueden añadir con posterioridad).



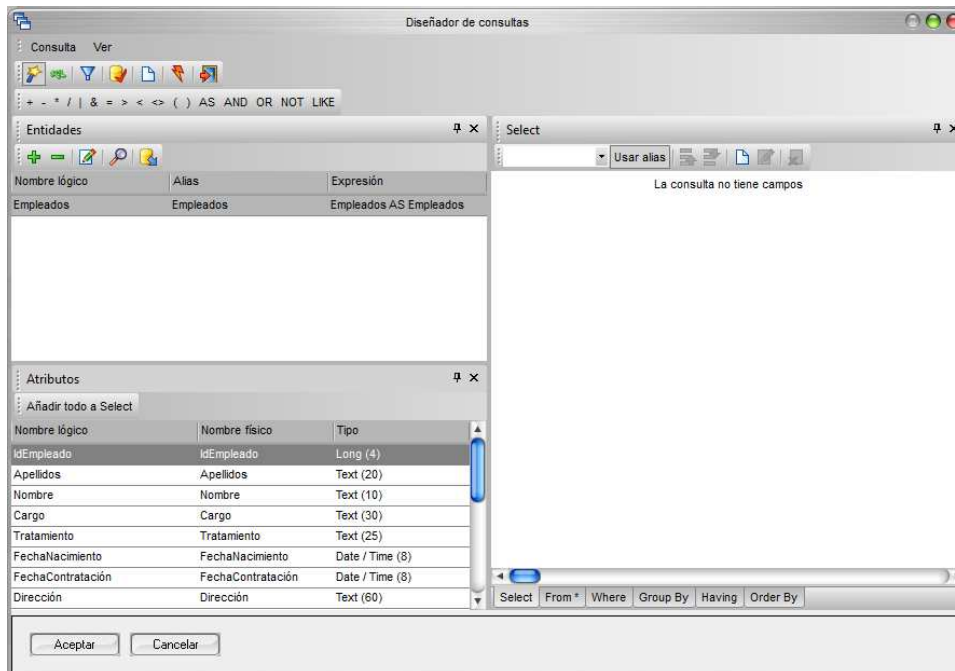
Con la ventana de selección en primer plano, usaremos la tecla Ctrl para escoger aquellas tablas que deseamos consultar. En nuestro ejemplo seleccionaremos la tabla **[Empleados]**



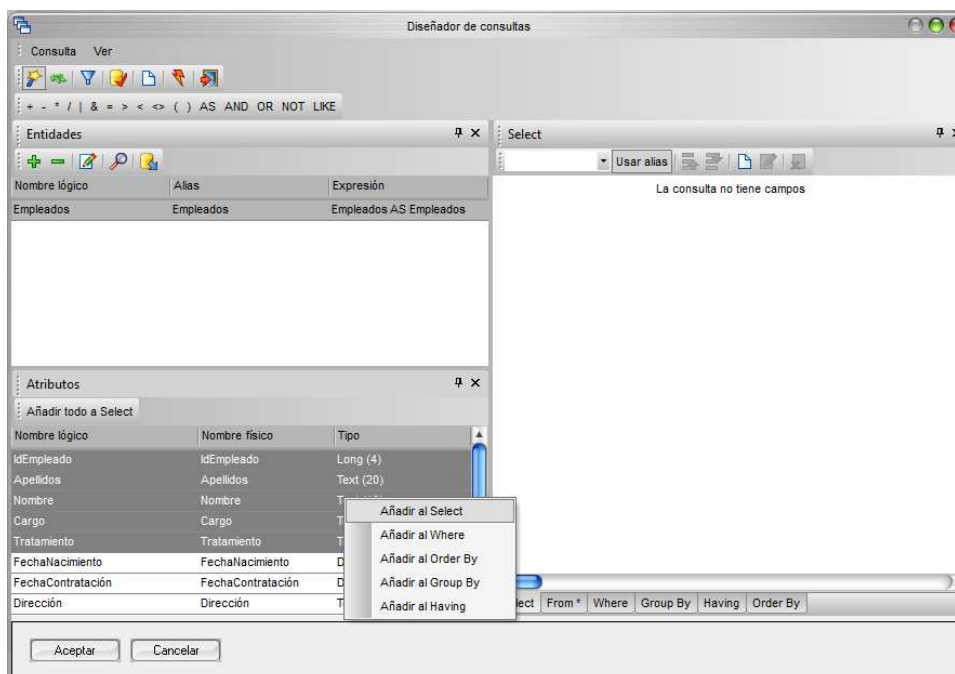
Al pulsar Aceptar aparecerá una ventana en que se sugerirá [Operador para Alias], pulsaremos de nuevo Aceptar, puesto que usaremos el configurado por defecto en la base de datos.



El asistente mostrará el siguiente estado:

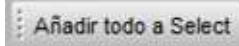


Para seleccionar campos, haremos un clic del ratón sobre ellos manteniendo pulsada la tecla [Ctrl] para realizar una selección múltiple. Luego abriremos el menú contextual con el botón derecho del ratón y los campos todavía seleccionados, y podremos añadirlos a cualquier parte de la consulta (select, from, where, group by, having u order by).

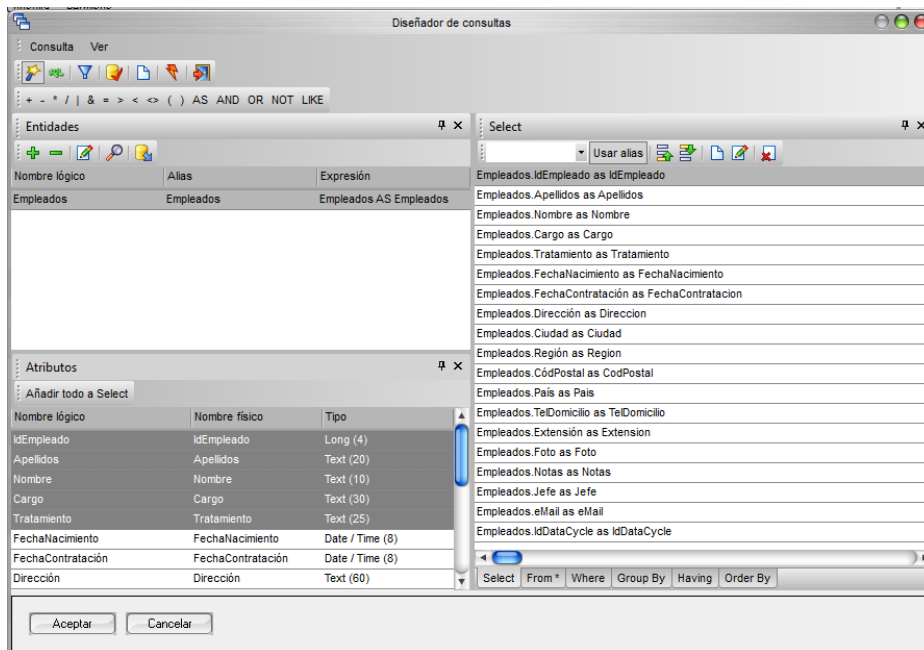


Se puede también hacer cambiar el tamaño de cualquier parte del formulario (entidades, atributos o partes de la consulta). Para ello nos debemos situar en el borde del elemento que queremos ampliar o reducir y arrastrarlo hasta la posición deseada.

En el ejemplo actual, añadiremos todos los campos al **[Select]** de la consulta, pulsando el botón



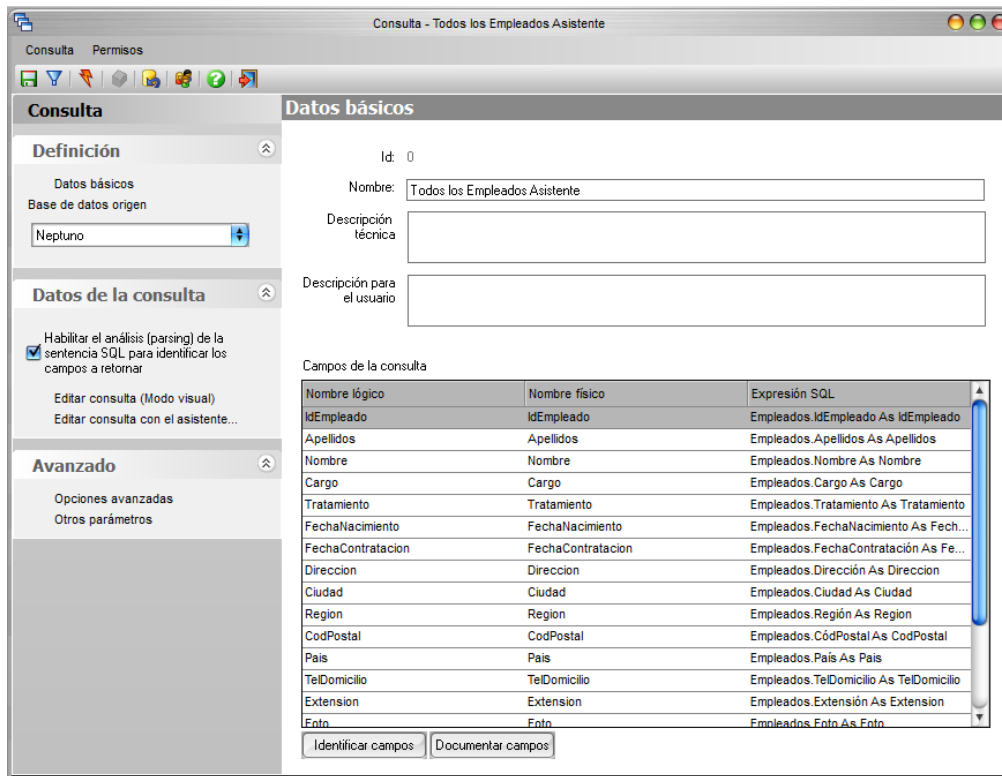
Una vez seleccionados los campos, el asistente mostrará el siguiente estado:






Vamos a Ejecutar la consulta para verificar que no hemos cometido ninguna equivocación. Para ello, pulsamos el botón **Ejecutar**, y aparecerá el “**Visor de Datos**” de DataCycle mostrando los registros obtenidos por la consulta.



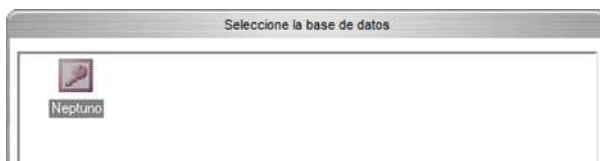
Para cerrar el Visor de Datos pulsamos sobre (salir). Seguidamente regresamos a la ventana de edición de la consulta, y, pulsamos .



Finalmente pulsar sobre el botón  para salvar la consulta y sobre  para salir.

A partir de ahora podremos ejecutarla tantas veces como deseemos sin necesidad de entrar a la ventana de edición de consultas. Simplemente hay que pulsar sobre el botón  de la barra de herramientas principal.

Mediante el menú contextual elegimos la opción  Nueva consulta...



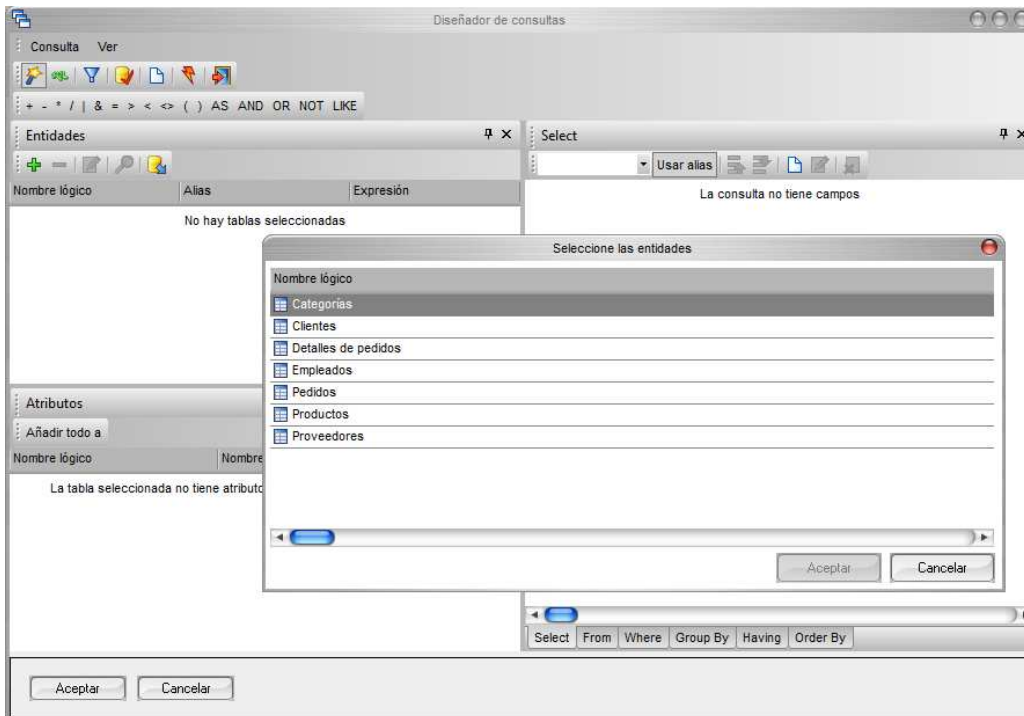
En primer lugar seleccionamos la base de datos origen sobre la que deseamos diseñar la consulta



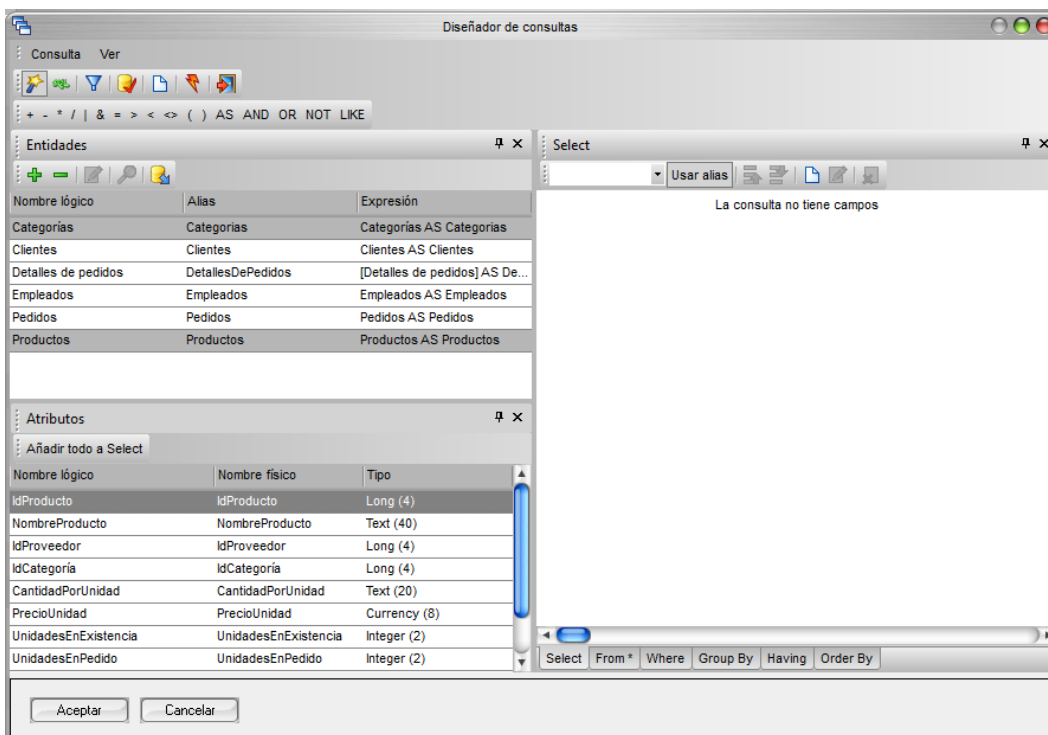
asignamos un título a la nueva consulta: **Detalle de Ventas Asistente.**

Vamos a utilizar el asistente de consultas Visual de DataCycle Reporting para diseñar la consulta que recuperará la información de la base de datos:

Pulsamos sobre **“Editar consulta con el asistente...”** para iniciar el asistente de diseño de consultas.



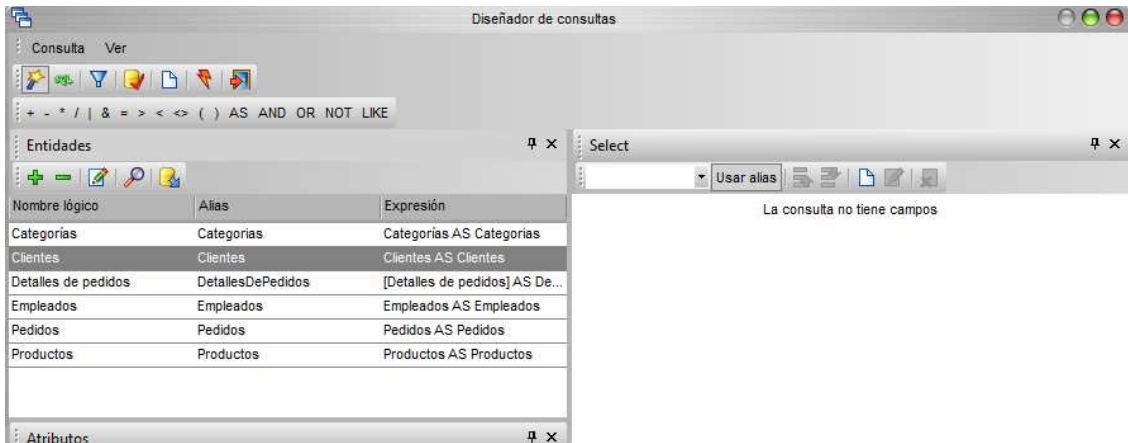
Seleccionamos las entidades (tablas o ficheros) que participarán en la consulta. En este caso seleccionamos con ayuda de la tecla Ctrl las entidades (una por una): **Categorías, Clientes, Empleados, Detalles de pedidos, Pedidos y Productos**, y pulsamos **Aceptar**. Aparecerá una ventana preguntándonos por el alias que aplica a cada entidad. Siempre pulsaremos **Aceptar**. La ventana del diseñador de consultas quedará así:



Queremos que la consulta devuelva una estructura como la siguiente:

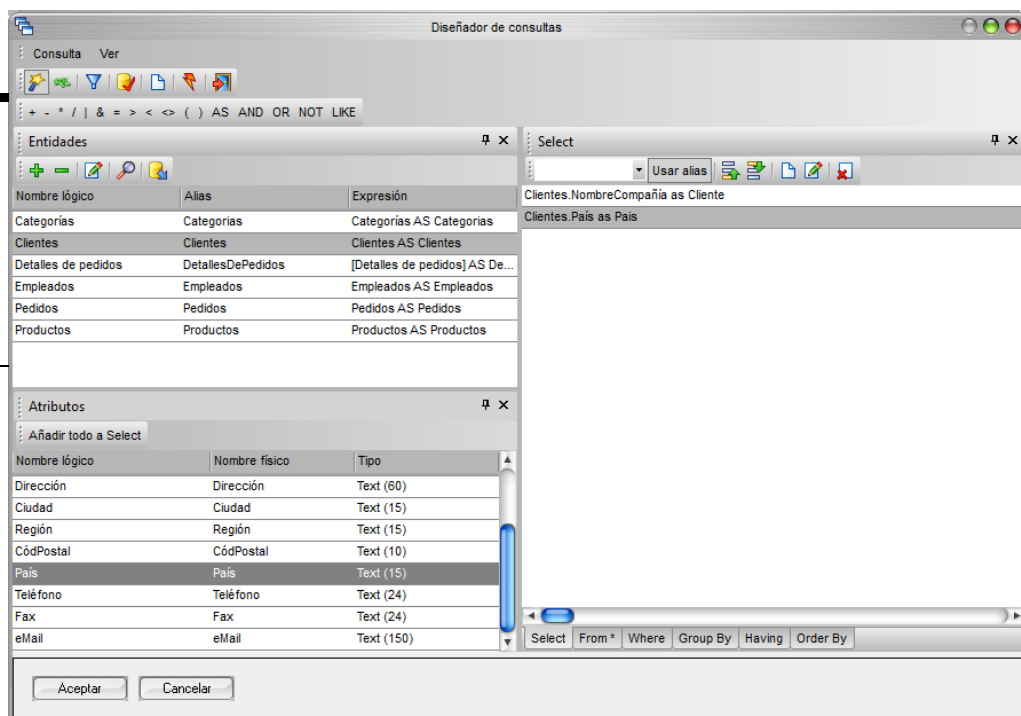
| Cliente | País | Representante | Categoría | Producto | Año | Mes | Ventas | Unidades |
|---------|------|---------------|-----------|----------|-----|-----|--------|----------|
|---------|------|---------------|-----------|----------|-----|-----|--------|----------|


Los dos primeros campos: **Cliente** y **País**, son de la tabla de Clientes. Por lo tanto seleccionaremos la tabla de clientes con el ratón en la lista superior: izquierda (**Entidades**).



Al seleccionar clientes, la lista inferior izquierda (**Atributos**) cambia y muestra los campos de la tabla de clientes. Como necesitamos el nombre del cliente y su país seleccionamos el campo **NombreCompañía** y lo arrastramos hacia la pestaña del select (derecha), le ponemos un alias "Cliente". Hacemos lo mismo con el campo **País**.

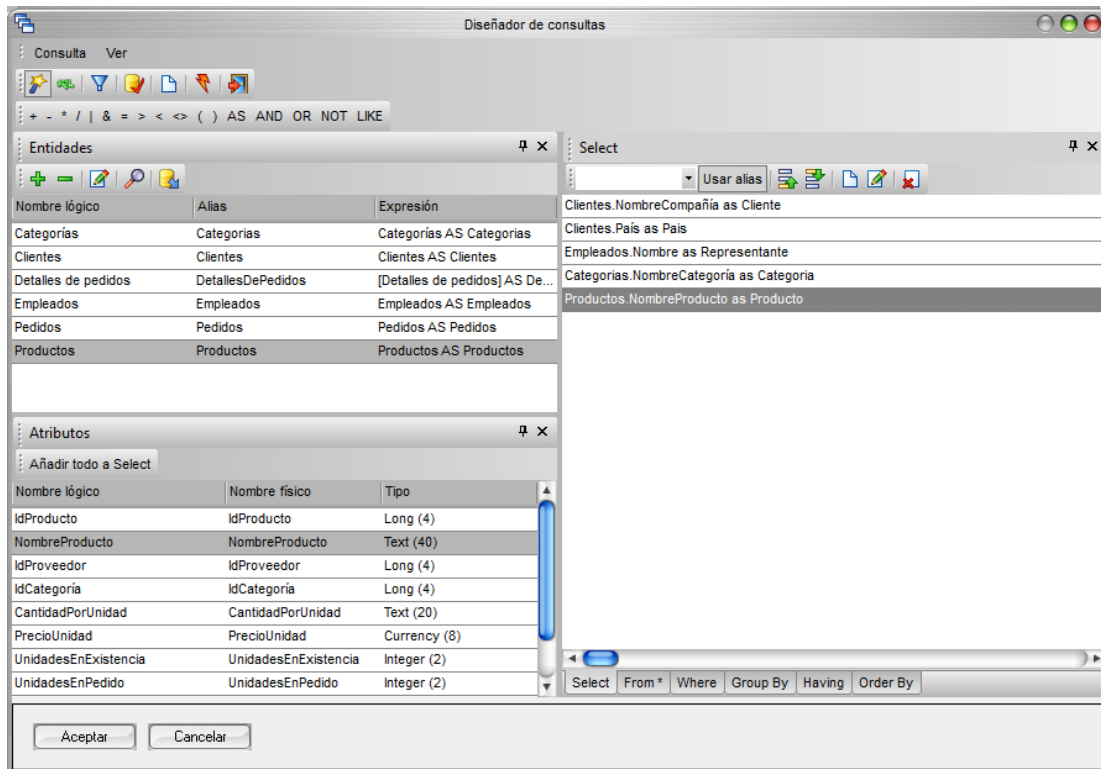
La ventana quedará con los dos campos en la lista de la derecha:



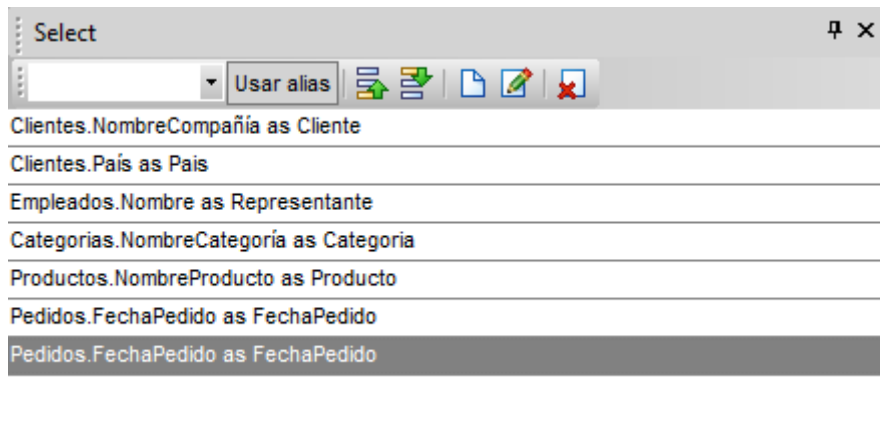
NOTA: Si nos equivocamos de campo, podemos eliminarlo seleccionándolo y pulsando el botón  (Borrar).

El siguiente campo a poner es el **Representante**, es decir el nombre del producto, que se encuentra en la tabla **Empleados**. Seleccionamos la tabla **Empleados** en la lista superior izquierda y elegimos y arrastramos el campo **Nombre**.

Tenemos que hacer la misma operación para añadir la **Categoría** que se encuentra en la tabla **Categorías** y es el campo **NombreCategoría**, y lo mismo para el **Producto** que se encuentra en la tabla de **Productos** y es el campo **NombreProducto**. La consulta quedará así:

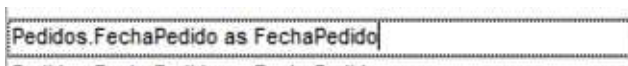


Ahora falta el **Mes** y el **Año**. Los sacaremos de la fecha del pedido. Añadimos **2 veces** el campo **FechaPedido** de la tabla **Pedidos**.



Hemos añadido dos veces la fecha de pedido por que el mes lo sacaremos recogiendo el mes de la fecha del pedido con ayuda de una función, y con el año haremos lo mismo.

Hacemos un doble-clic sobre el primer campo **Pedidos.FechaPedido** de la lista de la derecha (pestaña **Select**) y nos permite modificarlo:

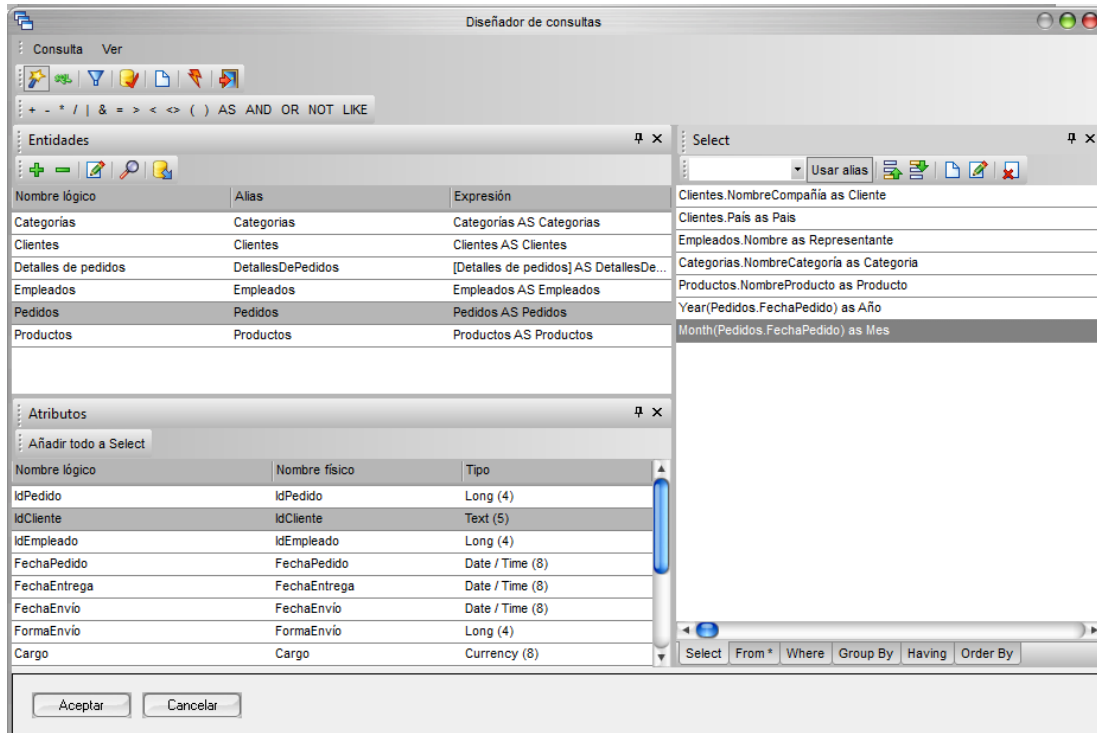


Modificamos el texto para que sea:

Year(Pedidos.FechaPedido) as Anyo

Y haremos algo parecido con el segundo **Pedidos.FechaPedido**. Hacemos un doble-clic sobre **Pedidos.FechaPedido** y cambiamos el texto para que sea:

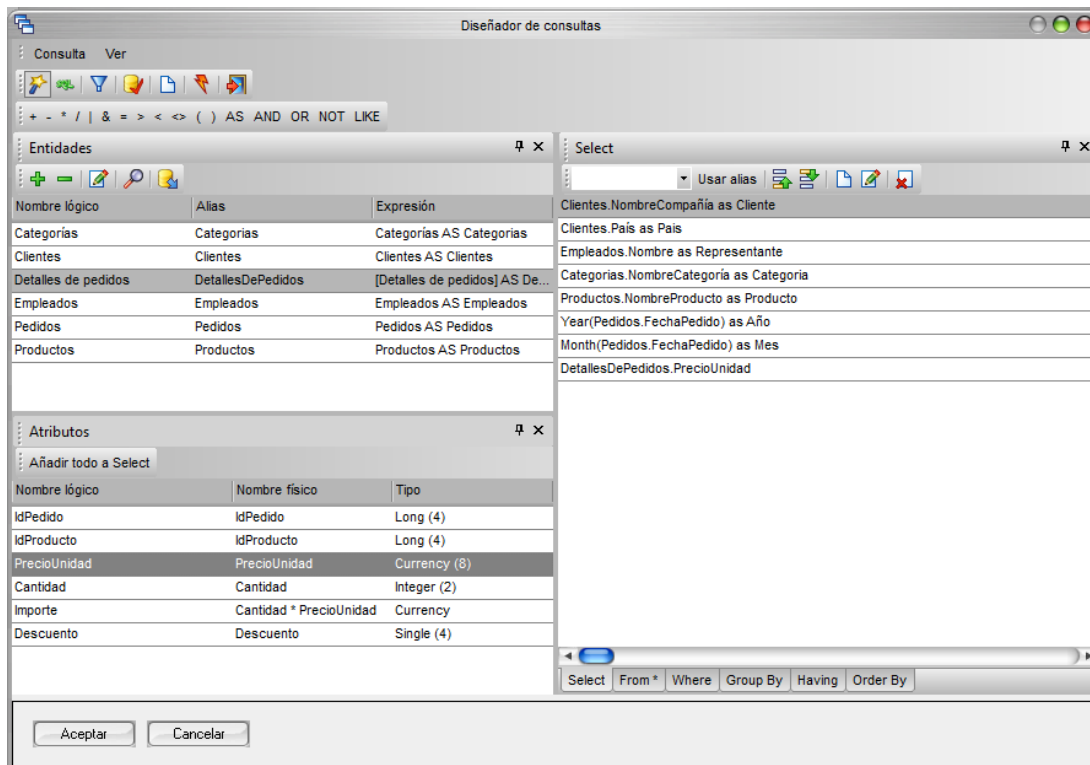
Month(Pedidos.FechaPedido) as Mes



Pese a que la entidad **Detalles de pedidos** ya incorpora un campo calculado que nos devuelve el importe del pedido, a modo de ejemplo crearemos en la consulta un campo que realice el mismo cálculo.

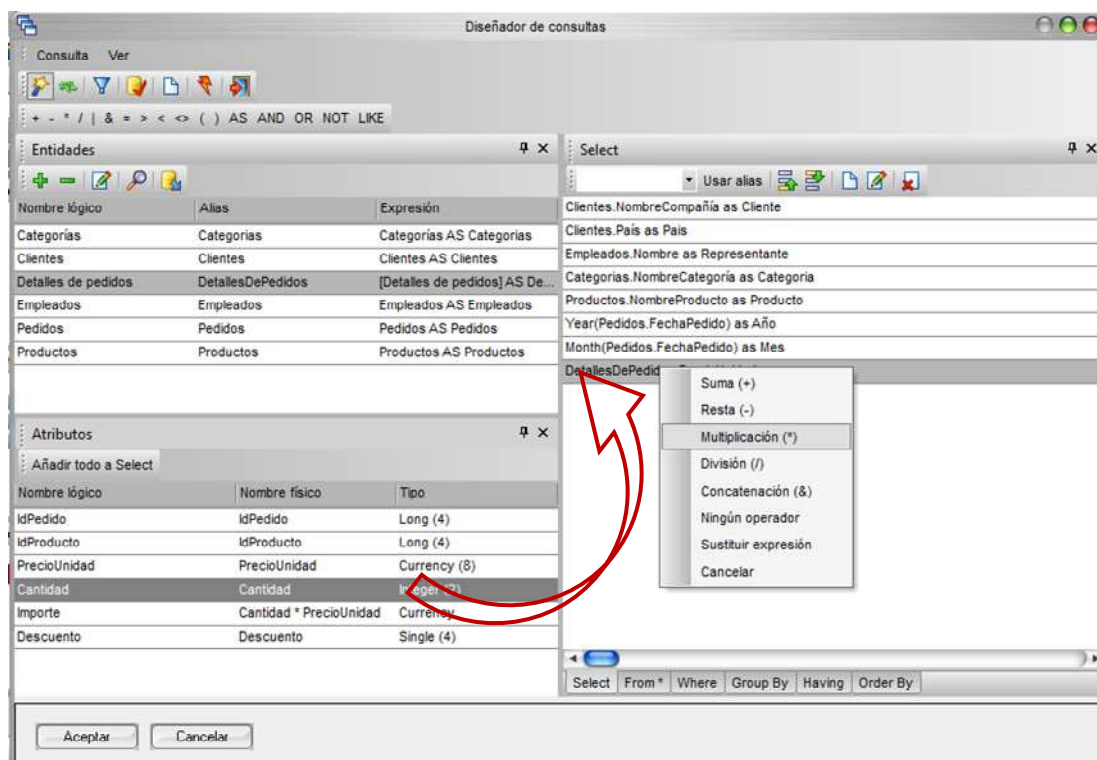
Primero debemos desactivar la opción de Alias (nombre lógico que se da a un campo de una consulta para entender mejor su significado y usarlo en el resto de la consulta) para poder usar el campo en una operación. Para ello pulsaremos sobre el botón de **Usar alias** situado en la ventana de **Select** para inhabilitar la opción de Alias. El botón quedará sin estar remarcado (**Usar alias**).

Ahora, seleccionamos la tabla **Detalles de Pedidos** y elegiremos el **PrecioUnidad**. De modo que la lista de campos seleccionados quedará así:

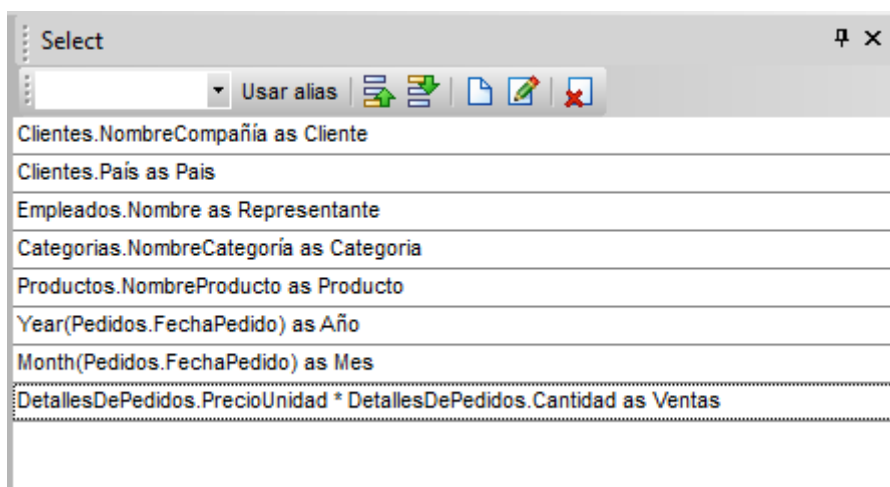


Pero hemos seleccionado el campo [Detalles de pedidos].PrecioUnidad cuando en realidad queremos el importe. El importe se define como la multiplicación entre el **PrecioUnidad** y la **Cantidad**. Vamos a indicar esta operación.

Ahora, debemos seleccionar el campo **Cantidad** de la lista de la izquierda, arrastrarlo y dejarlo caer sobre el campo [Detalles de pedidos].PrecioUnidad de la lista de la derecha:

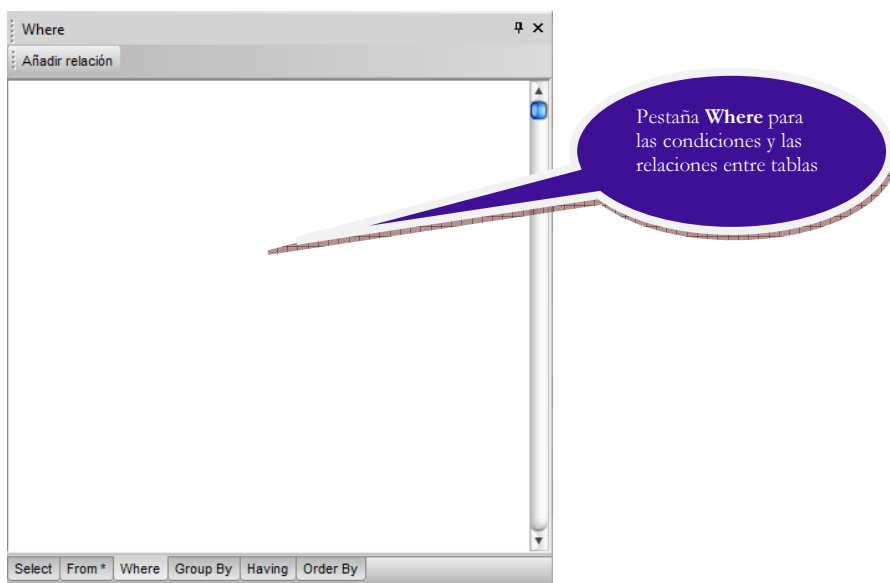


Justo al soltar el botón del ratón aparece un menú indicando que operación queremos efectuar, elegimos **Multiplicación (*)**, y le ponemos como alias Ventas. La lista de campos seleccionados quedará así:



Por último, añadiremos un campo más al Select, el campo **DetallesDePedidos.Cantidad**, pues nos informará del número de unidades de cada pedido.

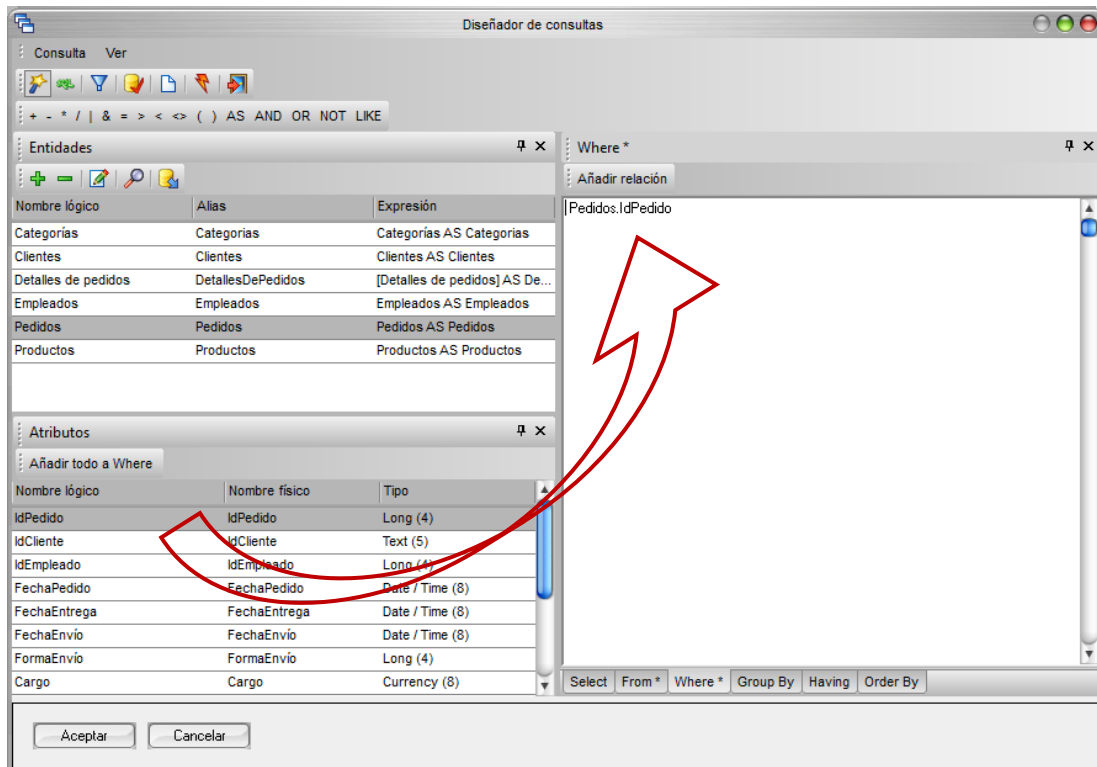
Ya tenemos la lista de campos que queremos. Ahora sólo falta indicar las uniones entre los ficheros en la pestaña **Where**. Pulsamos esa pestaña.



Tenemos que unir: **Detalles de pedidos** con **Pedidos**, **Categorías** con **Productos**, **Cientes** con **Pedidos**, **Empleados** con **Pedidos**, y **Productos** con **Detalles de pedidos**.

La primera condición de unión que vamos a definir es que el código de pedido del detalle de pedidos es igual al código de pedido del Pedido. Para ello vamos empezar arrastrando el campo

IdPedido de la tabla **Pedidos** que debe de aparecer en la lista inferior izquierda, hacia el cuadro de la derecha de **Condiciones generales** (asegúrese primero que en la lista superior izquierda tenga seleccionada la tabla **Detalles de pedidos**).



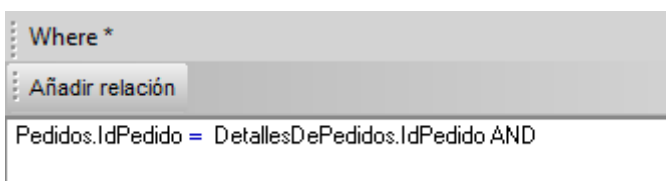
Como el texto de las condiciones se puede editar, hacemos un clic en el cuadro **Condiciones generales** y tecleamos el símbolo igual = después de *Pedidos.IdPedido*.



Ahora buscamos el campo **IdPedido** de la tabla **Detalles de Pedidos** y lo arrastramos de igual modo al cuadro de condiciones generales, para que quede:



Hacemos un clic sobre el cuadro de condiciones y tecleamos al final la palabra **AND** para enlazar con otras condiciones:



Y hacemos las mismas operaciones con:

- `Productos.IdProducto=DetallesDePedidos.IdProducto`
- `Categorias.IdCategoría=Productos.IdCategoría`
- `Cientes.IdCliente=Pedidos.IdCliente`
- `Empleados.IdEmpleado=Pedidos.IdEmpleado`


Para que al final quede así:

Where *

Añadir relación

Pedidos.IdPedido = DetallesDePedidos.IdPedido **AND**
DetallesDePedidos.IdProducto = Productos.IdProducto **AND**
Productos.IdCategoría = Categorias.IdCategoría **AND**
Pedidos.IdCliente = Cientes.IdCliente **AND**
Pedidos.IdEmpleado = Empleados.IdEmpleado

Observe **con atención** el uso de la palabra **AND** que sirve para juntar condiciones. Por lo tanto al final de todo no hav un **AND**.

En este punto podemos probar de ejecutar la consulta pulsando el botón .

Consulta Ver

Añadir relación

Pedidos.IdPedido = DetallesDePedidos.IdPedido **AND**
DetallesDePedidos.IdProducto = Productos.IdProducto **AND**
Productos.IdCategoría = Categorias.IdCategoría **AND**
Pedidos.IdCliente = Cientes.IdCliente **AND**
Pedidos.IdEmpleado = Empleados.IdEmpleado

El botón **Ejecutar** efectúa la consulta y nos muestra el resultado.

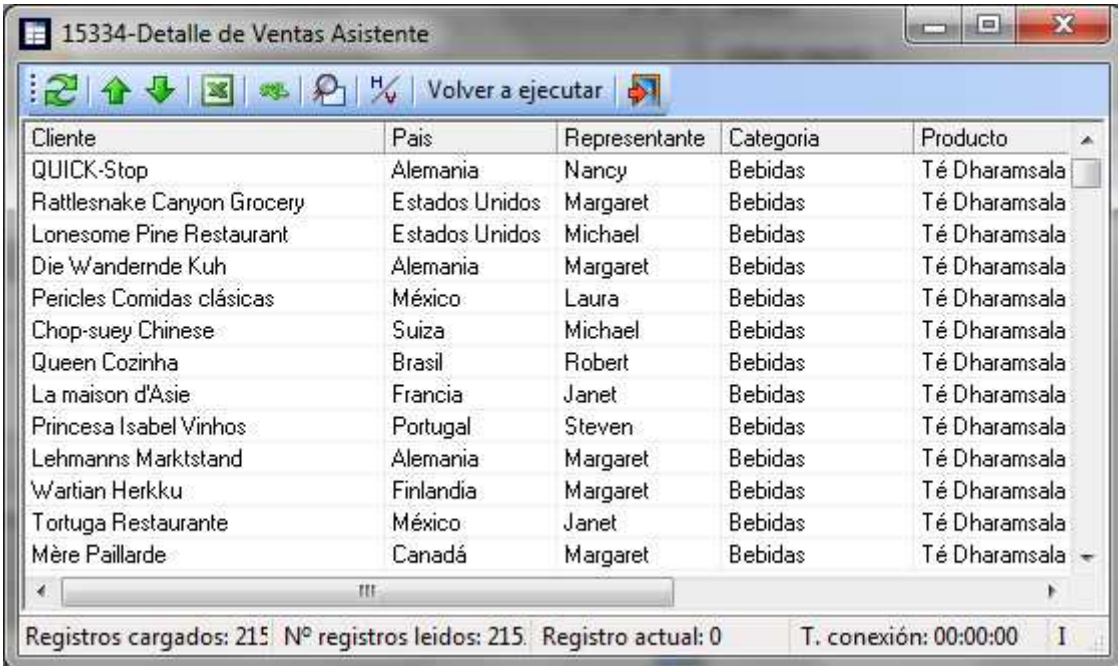
| Nombre lógico | Alias | Expresión |
|---------------------|-------------------|--------------------------------|
| Categorias | Categorias | Categorias AS Categorias |
| Cientes | Cientes | Cientes AS Cientes |
| Detalles de pedidos | DetallesDePedidos | [Detalles de pedidos] AS De... |
| Empleados | Empleados | Empleados AS Empleados |
| Pedidos | Pedidos | Pedidos AS Pedidos |
| Productos | Productos | Productos AS Productos |

| Nombre lógico | Nombre físico | Tipo |
|-------------------|-------------------|-----------------|
| IdEmpleado | IdEmpleado | Long (4) |
| Apellidos | Apellidos | Text (20) |
| Nombre | Nombre | Text (10) |
| Cargo | Cargo | Text (30) |
| Tratamiento | Tratamiento | Text (25) |
| FechaNacimiento | FechaNacimiento | Date / Time (8) |
| FechaContratación | FechaContratación | Date / Time (8) |
| Dirección | Dirección | Text (60) |



Select From * Where * Group By Having Order By

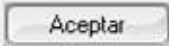


Aceptar Cancelar

El resultado aparece como:



| Cliente | Pais | Representante | Categoria | Producto |
|----------------------------|----------------|---------------|-----------|---------------|
| QUICK-Stop | Alemania | Nancy | Bebidas | Té Dharamsala |
| Rattlesnake Canyon Grocery | Estados Unidos | Margaret | Bebidas | Té Dharamsala |
| Lonesome Pine Restaurant | Estados Unidos | Michael | Bebidas | Té Dharamsala |
| Die Wandernde Kuh | Alemania | Margaret | Bebidas | Té Dharamsala |
| Pericles Comidas clásicas | México | Laura | Bebidas | Té Dharamsala |
| Chop-suey Chinese | Suiza | Michael | Bebidas | Té Dharamsala |
| Queen Cozinha | Brasil | Robert | Bebidas | Té Dharamsala |
| La maison d'Asie | Francia | Janet | Bebidas | Té Dharamsala |
| Princesa Isabel Vinhos | Portugal | Steven | Bebidas | Té Dharamsala |
| Lehmans Marktstand | Alemania | Margaret | Bebidas | Té Dharamsala |
| Wartian Herkku | Finlandia | Margaret | Bebidas | Té Dharamsala |
| Tortuga Restaurante | México | Janet | Bebidas | Té Dharamsala |
| Mère Paillard | Canadá | Margaret | Bebidas | Té Dharamsala |

Cerramos esta ventana con el icono  de la barra de herramientas. Seguidamente pulsamos  en el diseñador de consultas (este se cerrará) y regresamos a la ventana de edición de la consulta.

Pulsamos sobre  y se nos mostrará la ventana principal de consultas. Finalmente, pulsaremos el botón  para grabar la consulta y sobre  para salir de la ventana de edición de consulta.