

1. VISTAS

➤ TAREA Vistas.1: Trabajando con vistas.

1. En la BD SOCIOS crea una vista de nombre **v_inf_profesor** que contenga todos los datos de los profesores de la tabla EMPLEADO y también la especialidad de cada uno.
2. Modifica la vista **v_inf_profesor** para que almacene los datos de los profesores que están en la tabla de empleados, pero sin la especialidad.
3. Modifica la vista **v_inf_profesor** para que ahora su código esté cifrado. Comprueba con el procedimiento almacenado del sistema `sp_helptext` qué ocurre cuando quieres consultar el texto de la vista.
4. Elimina la vista **v_inf_profesor** creada anteriormente.
5. En la BD EMPLEADOS crea una vista de nombre **v_vendedores_oeste** que muestre los nombres de los vendedores de la región OESTE. Realiza la instrucción que muestra los datos de la vista.
6. En la BD EMPLEADOS crea la vista **v_pedidos_clientes_del_vend102** que contenga solamente los pedidos remitidos por clientes asignados al vendedor 102.
7. En la BD EMPLEADOS definir una vista de nombre **v_cliente_y_su_vendedor** sobre la tabla CLIENTE que incluya únicamente los nombres de los clientes y los vendedores a los que están asignados.
8. En la BD EMPLEADOS crea una vista de nombre **v_oficinas_ventas_altas** que devuelva todos los datos de las oficinas con ventas superiores a 500.000.
9. Utilizando transacciones para no realizar los cambios en la bd, ejecuta la consulta que inserte una nueva oficina a través de la vista **v_oficinas_ventas_altas** con los siguientes datos:

OFICINA	CIUDAD	REGIÓN	DIR	OBJETIVO	VENTAS
77	NEGREIRA	NOROESTE	105	300.000	100.000

¿Te permite hacer la inserción? Si es así comprueba si se ha añadido la fila y deshaz la transacción. Si no te lo permite ¿sabrías explicar por qué?

10. Modifica la vista **v_oficinas_ventas_altas** añadiéndole la cláusula `WITH CHECK OPTION`. A continuación intenta, usando transacciones, insertar la misma fila de antes en la tabla OFICINA pero a través de la vista.
¿Te permite hacer la inserción? Si es así comprueba si se ha añadido la fila y deshaz la transacción. Si no te lo permite ¿sabrías explicar por qué?
Quítale la cláusula `WITH CHECK OPTION` a la vista.

11. Crea una tabla de nombre OFI_PRUEBA que contenga los mismos datos que la tabla OFICINA. Ahora crea una vista de nombre **v_oficinas_este** que contengan todos los datos de las oficinas de la región este de la tabla OFI_PRUEBA, pero los campos en la vista se llamarán:

CODIGO	NOMBRE	ZONA	DIRECTOR	VENTAS_ESPERADAS	VENTAS_ACTUALES
--------	--------	------	----------	------------------	-----------------

Una vez creada la vista y comprobado su contenido, intenta borrar la tabla OFI_PRUEBA. ¿Te lo permite el servidor?

En caso de que te lo haya permitido, crea la tabla OFI_PRUEBA de nuevo y modifica la definición de la vista de tal manera que cuando se intente borrar la tabla sobre la que se ha definido la vista no nos lo permita.

Intenta ahora borrar la tabla OFI_PRUEBA de nuevo y comprueba qué ocurre ¿tiene sentido?

2. GESTIÓN DE USUARIOS EN SQL Server

➤ TAREA Usuarios.1: Trabajando con usuarios y esquemas.

1. Conéctate al servidor de tu máquina virtual.
2. En la BD EMPLEADOS crea tres usuarios de nombres **tarea2_usu1**, **tarea2_usu2** y **tarea2_usu3**. Recuerda que cada usuario debe tener un inicio de sesión asociado (créalos con el mismo nombre que el usuario, contraseña la misma que el nombre del login y bd por defecto EMPLEADOS). No indiques esquema por defecto a los usuarios.
3. Crea en la BD EMPLEADOS el esquema **Pruebas_tu_nombre** (en mi caso sería *Pruebas_Monica*) con propietario **tarea2_usu1**. En la misma instrucción crea una tabla de nombre **Fabricante** con 2 campos:
 - Identificador char(3) clave primaria
 - Nombre char(20) clave candidata
4. Establece 3 conexiones nuevas, una por cada inicio de sesión que has creado.
5. Realiza la consulta que devuelva toda la información de la tabla **Fabricante** en cada una de las diferentes sesiones. ¿Qué resultado obtienes? ¿Entiendes por qué con el usuario **tarea2_usu1** puedes consultar la tabla y con los demás no?
6. Cambia el usuario propietario del esquema **Pruebas_tu_nombre** por **tarea2_usu3**. Realiza la consulta que devuelva toda la información de la tabla **Fabricante** para los usuarios **tarea2_usu1** y **tarea2_usu3**. ¿Qué usuario/s pueden consultar ahora la tabla **Fabricante**?
7. Crea en la BD EMPLEADOS el esquema **Pruebas_tu_nombre2** (en mi caso sería *Pruebas_Monica2*) con propietario **tarea2_usu2**.
8. Dentro del esquema **Pruebas_tu_nombre2** crea una tabla **cliente2** con la misma estructura y contenido que la tabla **cliente** de la BD EMPLEADOS.
9. Realiza la consulta que devuelva toda la información de la tabla **cliente2** desde las conexiones de los 3 usuarios que has creado. ¿Qué usuario/s pueden consultar la tabla **cliente2**?
10. Mueve la tabla **cliente2** al primer esquema creado **Pruebas_tu_nombre**. Realiza la consulta que devuelva toda la información de la tabla **cliente2** desde las conexiones de los 3 usuarios que has creado. Realiza la consulta adecuada para obtener el contenido de **cliente2**. ¿Qué usuario/s pueden consultar la tabla **cliente2**?
11. Intenta eliminar el esquema **Pruebas_tu_nombre**. ¿Te lo permite? ¿Entiendes por qué?
12. Intenta eliminar el esquema **Pruebas_tu_nombre2**. ¿Te lo permite? ¿Entiendes por qué?

➤ **TAREA Usuarios.2: Gestión de usuarios, permisos y roles (ejercicio de tipo examen).**

1. Necesitamos que con nuestra bd puedan trabajar, por el momento, 3 usuarios con los siguientes datos:
 - **Jefe_proyecto**, bd por defecto EMPLEADOS y contraseña abc
 - **Programador**, bd por defecto EMPLEADOS y contraseña abc
 - **Usu_ingenuo**, bd por defecto EMPLEADOS y contraseña abc
2. El jefe de proyecto deberá poder trabajar con la bd con los mismos permisos que si hubiese sido él quién creó la bd.
3. El programador deberá poder leer y también escribir en cualquier objeto que existe en la bd, pero también en cualquiera de los que se vaya a crear, sin necesidad de darle los permisos sobre cada objeto.
4. El usuario ingenuo sólo podrá consultar alguna información, pero nunca directamente sobre las tablas. Debemos crear las estructuras que permitirán al usuario consultar la información sin acceder directamente a las tablas. Además estas estructuras deberán crearse organizadas todas juntas en una nueva estructura que a partir de aquí llamaré **estructura_usu_ingenuo**.
DATOS QUE PODRÁ CONSULTAR EL USUARIO INGENUO:
 - Nombre y edad del vendedor y el nombre de la oficina de aquellos vendedores que trabajan en oficinas de la región este. Las columnas resultantes se llamarán VENDEDOR, AÑOS, OFICINA. Para darle el nombre a las columnas no podrás hacerlo usando alias en el SELECT.
 - Todos los datos de las oficinas de la región ESTE. Crea esta estructura de tal manera que a través de ella no se puedan realizar operaciones que vulneren las condiciones del where.
5. El usuario ingenuo deberá poder consultar cualquiera de las estructuras que has creado en la estructura_usu_ingenuo y todas las que vayas a crear dentro de ella en un futuro, sin que tengamos que darle el permiso correspondiente sobre cada objeto.
6. Crea en el esquema dbo una copia de las tablas PEDIDO y PRODUCTO de nombres PEDIDO2 y PRODUCTO2 (no uses la instrucción CREATE TABLE).
7. Crea un esquema de nombre esquema_pedidos y mueve las 2 tablas nuevas PEDIDO2 y PRODUCTO2 a este esquema.
8. Realiza la instrucción que permite que cualquier usuario existente en la bd EMPLEADOS o cualquiera que se cree en un futuro, pueda consultar y eliminar filas de la tabla PEDIDO2.
9. Crea el inicio de sesión (*sólo el inicio de sesión*) inicio_sesion_01 con contraseña abc y bd por defecto EMPLEADOS. La primera vez que se conecte el inicio de sesión inicio_sesion_01 al servidor deberá cambiar la contraseña.
¿Puede conectarse inicio_sesion_01 al servidor? ¿Por qué?
10. Usando T-SQL consigue que el inicio de sesión inicio_sesion_01 pueda conectarse a EMPLEADOS sin crearle usuario en la bd EMPLEADOS.
11. ¿Puede el inicio de sesión inicio_sesion_01 consultar la tabla PEDIDO2? ¿Por qué?
12. ¿Puede el inicio de sesión inicio_sesion_01 consultar la tabla PRODUCTO2? ¿Por qué?
13. Crea una vista en el esquema esquema_pedidos que se llame v_productos_mas100stock y que devuelva la descripción, el precio y las existencias de los productos de PRODUCTO2 que superan las 100 unidades en stock. Los campos de la vista se llamarán PRODUCTO, IMPORTE y UNIDADES_EXISTENTES y no le puedes dar el nombre a los campos en el SELECT de la vista.
14. Modifica la vista v_productos_mas100stock para que su código esté cifrado.

15. Modifica la vista v_productos_mas100stock para que no se puedan insertar, modificar ni borrar productos a través de ella con stock menor o igual a 100 existencias.
16. Modifica la v_productos_mas100stock para que no se pueda ni eliminar, ni ejecutar instrucciones ALTER sobre la tabla PRODUCTO2 del esquema esquema_pedidos. Es decir, debes vincular la vista al esquema esquema_pedidos.
17. Intenta eliminar la tabla PRODUCTO2 del esquema esquema_pedidos. ¿Te lo permite? ¿por qué?
18. Usando el procedimiento almacenado adecuado consulta el texto de la vista v_productos_mas100stock. ¿Te muestra el texto? ¿Por qué?
19. Escribe la instrucción que cambia el propietario del esquema esquema_pedidos al usuario programador.
20. Suponiendo que en tu equipo existe un usuario de Windows de nombre LoginSQL qué instrucción o instrucciones realizarías para que se pueda conectar al servidor SQL Server.
21. Crea un usuario en la BD EMPLEADOS de nombre inicio_sesion_02 y asociado a un inicio de sesión con el mismo nombre y con contraseña abc
22. Crea un usuario en la BD EMPLEADOS de nombre inicio_sesion_03 y asociado a un inicio de sesión con el mismo nombre y con contraseña abc
23. Como sa dale el permiso de SELECT sobre REPVENTAS a inicio_sesion_02 de tal manera que pueda conceder ese permiso a otros.
24. Quítale a inicio_sesion_02 el permiso de concederlo a otros. Ejecuta una instrucción de consulta sobre REPVENTAS siendo inicio_sesion_02. ¿Puedes hacerlo? ¿Por qué?
25. Como inicio_sesion_02 concédele el permiso de SELECT sobre REPVENTAS a inicio_sesion_03. ¿Puedes hacerlo? ¿Por qué?
26. Quítale a inicio_sesion_02 el permiso de consulta sobre REPVENTAS. Ejecuta una instrucción de consulta sobre REPVENTAS siendo inicio_sesion_02. ¿Puedes hacerlo? ¿Por qué?
27. Como sa dale al usuario inicio_sesion_03 el permiso de borrar cualquier fila de cualquier objeto de la bd.
28. Quítale el permiso anterior y comprueba que ahora no puedes lanzar una instrucción de borrado sobre la tabla OFICINA. (Usa transacciones en el borrado por si te equivocas).
29. El director de la empresa quiere poder realizar las mismas funciones en nuestra instancia de bd que sa. Realiza las instrucciones de T-SQL necesarias para que se pueda conectar (se conectará con nombre director y contraseña director) y para que realice cualquier función en cualquier bd como haría el sa.
30. Conectado como director y desde la bd master y sin usar la instrucción USE, realiza una consulta a la tabla ACTIVIDAD de la bd SOCIOS.

3. COPIAS DE SEGURIDAD en SQL Server

➤ **TAREA Copias.1: Usando T-SQL realiza lo que se te pide en los siguientes apartados.**

1. Crea una carpeta en la unidad D de nombre COPIAS_SEGURIDAD.
2. Pon el modelo de recuperación de la bd SOCIOS a Simple.
3. Realiza una copia de seguridad completa de la bd SOCIOS y guárdala en la carpeta que acabas de crear en un fichero con el nombre SOCIOS.bak. El nombre de la copia será COPIA_COMPLETA_SOCIOS.
4. Comprueba que se ha creado el fichero COPIAS_SEGURIDAD\SOCIOS.bak
5. Ahora crea una tabla copia de la tabla AULA que se llame AULA_COPIA.
6. Realiza una copia diferencial en la carpeta que has creado en un fichero de nombre SOCIOS_DIFF.bak. El nombre de la copia será COPIA_DIFERENCIAL_SOCIOS.
7. Comprueba que se ha creado el fichero COPIAS_SEGURIDAD\SOCIOS_DIFF.bak
8. Restaura la copia completa de la base de datos. Como después vas a restaurar una copia de BD diferencial, recuerda usar la opción NORECOVERY.
9. Puedes acceder a la base de datos SOCIOS ¿Entiendes por qué?
10. Restaura la copia diferencial de la base de datos.
11. ¿Está AULA_COPIA? ¿Entiendes por qué?

➤ **TAREA Copias.2**

En este ejercicio vas a utilizar el Management Studio de SQL Server para crear dispositivos de copia de seguridad que contengan copias de seguridad de la base de datos y del registro de transacciones.

Objetivos: Después de realizar esta práctica, deberás ser capaz de:

- Crear un archivo de copia de seguridad permanente.
- Hacer una copia de seguridad de una base de datos.
- Hacer una copia de seguridad de un registro de transacciones.
- Hacer una copia de seguridad diferencial.

2.1. Creación de un dispositivo de copia de seguridad mediante el Management Studio.

En este procedimiento vas a utilizar el Management Studio de SQL Server para crear dos dispositivos de copia de seguridad.

■ **PASOS:**

1. Abre el Management Studio de SQL Server.
2. En el explorador de objetos sitúate en la carpeta **Objetos de servidor**.
3. En el menú contextual de la carpeta **Dispositivos de Copia de seguridad**, haz clic en la opción **Nuevo dispositivo de copia de seguridad**.
4. Con el Explorador de Windows, crea una carpeta denominada Backup en el directorio raíz de la unidad D:
5. Utiliza la información de la tabla siguiente para crear dos archivos de copia de seguridad permanentes con el Management Studio de SQL Server.

NOMBRE	NOMBRE DEL ARCHIVO EN DISCO
Nw1	D:\Backup\Nw1.bak
Nwlog	D:\Backup\Nwlog.bak

2.2. Confirmación de la creación de los nuevos dispositivos

En este procedimiento vas a utilizar el Management Studio de SQL Server para confirmar que se han creado Nw1 y Nwlog. *(Puede que ya estés viendo los dispositivos que acabas de crear, pero a veces es necesario Actualizar).*

■ PASOS:

1. En el explorador de objetos de SSMS sitúate en la carpeta **Objetos de servidor**.
2. Haz clic con el botón derecho del ratón en la carpeta **Dispositivos de Copia de seguridad** y, después, haz clic en **Actualizar**.
3. Confirma que se han creado los dispositivos Nw1 y Nwlog.

2.3. Realización de la copia de seguridad de una BD mediante el Management Studio.

En este procedimiento vas a realizar una copia de seguridad completa de la base de datos **Northwind** en el archivo de copia de seguridad Nw1 del disco.

■ PASOS:

1. En el explorador de objetos, haz clic con el botón secundario del ratón en **Bases de datos** ->**Northwind** y, a continuación, haz clic en **Propiedades**.
2. En la ventana **Opciones**, escoge como modelo de recuperación el **Simple (Simple)** y, a continuación, haz clic en **Aceptar**.
3. En el explorador de objetos, haz clic con el botón derecho del ratón en **Administración** y en el menú contextual que obtienes selecciona la opción **Copia de seguridad**.
4. En la ventana **Copia de seguridad** de SQL Server, en la opción *General*, rellena las opciones con la información de la tabla siguiente.

CAMPO	VALOR
Base de datos	Northwind
Nombre	NorthwindCompleta
Tipo de copia de seguridad	Base de datos: completa
Descripción	Primera copia de seguridad de Northwind
Destino	Nw1

5. En la ventana **Copia de seguridad** de SQL Server, en la opción *Opciones*, en el apartado *Sobrescribir medios* escoge "Sobrescribir todos los conjuntos de copia de seguridad existentes".
6. Haz clic en **Aceptar** para realizar la copia de seguridad.

2.4. Visualización del contenido de un dispositivo de copia de seguridad

En este procedimiento vas a utilizar el Management Studio de SQL Server para ver el contenido del archivo de copia de seguridad Nw1 y asegurarte de que contiene una copia de seguridad completa de la base de datos.

■ PASOS:

1. En el Management Studio de SQL Server.
2. En el explorador de objetos sitúate en la carpeta **Objetos de servidor**.
3. Dentro de la carpeta **Dispositivos de Copia de seguridad** haz doble clic en el dispositivo Nw1.
4. Selecciona la opción **Contenido de los medios**.
5. Confirma que el archivo de copia de seguridad contiene la copia de seguridad completa de la base de datos que creaste anteriormente.

2.5. Hacer una copia de un registro de transacciones mediante el Management Studio.

En este procedimiento vas a utilizar el Management Studio de SQL Server para hacer la copia de seguridad del registro de transacciones de la base de datos **Northwind** en el archivo de copia de seguridad Nwlog.

■ PASOS:

1. En el explorador de objetos, haz clic con el botón secundario del ratón en **Bases de datos** ->**Northwind** y, a continuación, haz clic en **Propiedades**.
2. En la ventana **Opciones**, escoge como modelo de recuperación el **Completo (Completa)** y, a continuación, haz clic en **Aceptar**.
3. En el explorador de objetos, haz clic con el botón derecho del ratón en **Administración** y en el menú contextual que obtienes selecciona la opción **Copia de seguridad**.
4. En la ventana **Copia de seguridad** de SQL Server, en la opción *General*, rellena las opciones con la información de la tabla siguiente:

CAMPO	VALOR
Base de datos	Northwind
Nombre	NwindLog
Tipo de copia de seguridad	Registro de transacciones
Descripción	Registro de transacciones de Northwind
Destino	Nwlog

5. En la ventana **Copia de seguridad** de SQL Server, en la opción Opciones, en el apartado *Sobrescribir medios* escoge "Sobrescribir todos los conjuntos de copia de seguridad existentes".
6. Haz clic en **Aceptar** para realizar la copia de seguridad. No lo permite ¿por qué? ¿Qué deberíamos hacer para que nos lo permitiese. Haz los cambios necesarios para que se pueda ejecutar la copia de log.
7. Después de terminar la copia de seguridad, revisa el contenido del archivo de copia de seguridad Nwlog con el Management Studio de SQL Server.

2.6. Hacer una copia de seguridad diferencial

En este procedimiento vas a hacer una copia de seguridad diferencial de la base de datos **Northwind** y vas a anexarla al archivo de copia de seguridad.

■ PASOS:

1. En el explorador de objetos, haz clic con el botón derecho del ratón en **Administración** y en el menú contextual que obtienes selecciona la opción **Copia de seguridad**.
2. En la ventana **Copia de seguridad** de SQL Server, rellena las opciones con la información de la tabla siguiente:

CAMPO	VALOR
Base de datos	Northwind
Nombre	Nwind diferencial
Tipo de copia de seguridad	Base de datos: diferencial
Descripción	Cambios desde la última copia de seguridad completa de la base de datos de Northwind
Destino	D:\Backup\Nwdiff.bak

3. En la ventana **Copia de seguridad** de SQL Server, en la opción Opciones, en el apartado *Sobrescribir medios* escoge “Anexar al conjunto de copia de seguridad existente”.
4. Haz clic en **Aceptar** para realizar la copia de seguridad.
5. Después de terminar la operación de la copia de seguridad, haz clic con el botón secundario del ratón en la carpeta **Administración** y selecciona la **Copia de seguridad...** En la ventana que aparece selecciona la BD **Northwind**.
6. En la parte inferior de la ventana, en la lista **Destino**, haz clic en el botón Contenido.

➤ **TAREA Copias.3: Resuelve los siguientes apartados.**

1. Si no pudieses hacer un BACKUP de la BD SOCIOS ¿qué harías para copiar los ficheros de la BD a otro servidor, por ejemplo en una máquina virtual, y poder adjuntarlos?
Inténtalo copiando los ficheros directamente de la carpeta DATA a otra ubicación, ¿te lo permite? Consigue adjuntar (**attach**) la bd SOCIOS a un servidor de la máquina virtual en la que habías instalado dos instancias de SQL Server. Indica todos los pasos e instrucciones T-SQL que has realizado para conseguirlo.
2. Haz lo mismo que en el apartado 1 para la base de datos EMPLEADOS pero utilizando la opción **Separar (detach)** de la opción tareas del menú contextual de la base de datos.
3. ¿Qué base de datos crees que deberíamos copiar si quisiésemos guardar los inicios de sesión del servidor?

AGENTE SQL SERVER (SQL Server Agent)

Antes de comenzar con la **tarea 4** vamos a aprender algunos conceptos importantes de la programación de tareas en SQL Server.

Para programar cualquier tipo de trabajo debe estar arrancado el servicio **Agente SQL Server**. Si en nuestro servidor programamos tareas, debemos tener en cuenta que el servicio Agente SQL Server debe estar configurado como inicio automático.

Los componentes del Agente SQL Server que permiten la automatización son:

- las **alertas**,
- los **trabajos** y
- los **operadores**.

Los trabajos y las alertas se definen separadamente, y pueden ejecutarse o activarse de forma independiente. Pueden combinarse trabajos y alertas para lograr una funcionalidad adicional. Por ejemplo, si un trabajo no se ejecuta correctamente debido a un error del sistema, una alerta definida para responder a ese número de error podría ejecutar otro trabajo que resolviera el problema.

■ TRABAJOS:

Un **trabajo** es una serie de pasos (*acciones*) que realiza el Agente SQL Server.

Existen varias maneras de ejecutar trabajos:

- Según una o más programaciones.
- Como respuesta a una o varias alertas.
- Ejecutando el procedimiento almacenado **sp_start_job**.

Un paso de trabajo, por ejemplo, puede consistir en la ejecución de una instrucción Transact-SQL o un comando del sistema.

■ PROGRAMACIONES:

Una **programación** específica cuándo se ejecuta un trabajo. Se puede ejecutar más de un trabajo en la misma programación y se puede aplicar más de una programación al mismo trabajo. Una programación puede definir las condiciones siguientes del momento en el que se ejecuta un trabajo:

- Cuando se inicia el Agente SQL Server.
- Cuando el uso de la CPU del equipo se encuentre en un nivel que se haya definido como inactivo.
- Una vez, a una hora y una fecha específicas.
- Periódicamente.

■ ALERTAS:

Una **alerta** es una respuesta automática a un evento específico. Por ejemplo, un evento puede ser el inicio de un trabajo.

Una alerta puede realizar las acciones siguientes:

- Notificar a uno o varios operadores
- Ejecutar un trabajo

■ OPERADORES:

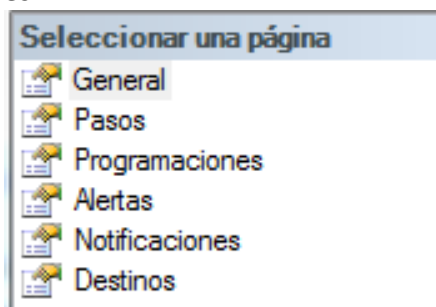
Los operadores definen información de contacto para las personas responsables del mantenimiento de una o varias instancias de SQL Server.

En algunas compañías, las responsabilidades de operador están asignadas a una sola persona. En compañías con varios servidores, muchas personas comparten las responsabilidades de operador. SQL Server puede notificar a los operadores de alertas mediante una o varias de las opciones siguientes:

- Correo electrónico
- Buscapersonas (por correo electrónico)
- **NET SEND**

■ CREACIÓN DE TRABAJOS CON SSMS:

Cuando creas un nuevo trabajo (con la opción del menú contextual de la carpeta Trabajos del Agente SQL Server), aparecen varias opciones.



- **General:** Indica un nombre que describa el trabajo, en categoría Mantenimiento de BBDD y una breve descripción del trabajo.
- **Pasos:** Crea un nuevo paso. En la ventana que aparece tienes dos opciones: *General* y *Avanzado*. En la opción *General* debes indicar un nombre para el paso (cada trabajo va a constar de un único paso en esta tarea). En tipo escoge Script T-SQL y escoge la BD donde quieres ejecutar la sentencia y el comando, es decir el código T-SQL que se ejecutará en ese paso del trabajo. Si, por ejemplo, estás realizando el trabajo de la copia de seguridad completa, en el comando escribirías la instrucción BACKUP DATABASE adecuada.

En la opción *Avanzado* puedes especificar alguna característica más como el número de reintentos o la acción a realizar en caso de producirse un error.

- **Programaciones:** En esta opción es donde determinamos la frecuencia y duración del trabajo.
- **Alertas:** Podemos asociar alertas a nuestro trabajo.
- **Notificaciones:** Acciones a realizar cuando el trabajo termine.
- **Destinos:** Aquí especificamos en qué servidores se va a lanzar el trabajo.

➤ TAREA Copias.4: Programar copias de seguridad

Si en nuestra estrategia de copias decidimos hacer una copia completa de nuestra bd AdventureWorks cada sábado a la 1:00 de la madrugada, una diferencial el resto de días de la semana a las 00:00 de la noche y una de log a las 6:00, 12:00 y 18:00, tenemos que programarlas para no tener que lanzar nosotros mismos el proceso en cada momento, sino que se ejecute automáticamente en el servidor. Debemos crear 3 trabajos nuevos desde el servidor, uno por cada tipo de copia de seguridad: completa, diferencial y de log.

➤ TAREA Copias.5: Ejercicio de tipo examen

En una empresa de nombre **INFOR_FCT S.L** tienen por estrategia de copias de seguridad de su bd **CONTABILIDAD** la siguiente:

Domingo	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
00:00 Copia Full	00:00 Copia Diff	00:00 Copia Diff	00:00 Copia Diff	00:00 Copia Diff	00:00 Copia Diff	00:00 Copia Diff
<ul style="list-style-type: none"> - Las copias completas se hacen en el fichero copias_full.bak en el directorio D:\Copias\backup. Cada copia completa sustituye a la copia del domingo anterior. - Las copias diferenciales se hacen en el fichero copias_diff.bak en el directorio D:\Copias\backup. Cada copia diferencial sustituye a la copia del día anterior. - Cualquier copia del registro de transacciones deberá realizarse en copias_log.bak en la carpeta H:\Copias\backup_log - Cualquier otro tipo de copia deberá realizarse en el dispositivo permanente de copias de seguridad OTRAS_COPIAS que debes suponer que ya está creado. 						

5.1. El martes a las 12:00hh llega un inspector de hacienda y solicita una copia de la bd **CONTABILIDAD** en el momento actual, pero el DBA no quiere que esta copia afecte al modelo de recuperación actual. ¿Qué instrucción de copia debe realizarse?

5.2. El viernes a las 15:37 tú que estás cursando en esta empresa la FCT formateas por error el disco donde la empresa guarda los datos (no el registro de transacciones) de la bd **CONTABILIDAD**. Tu tutor de FCT en la empresa te dice que pongas la bd a funcionar.

¿Qué instrucciones realizarás para que la bd quede operativa y con la menor pérdida de información posible? Debes suponer que el modo de recuperación de la bd **CONTABILIDAD** es Full.