

## 1. Estructura de almacenamiento y archivos de datos.

La estructura de almacenamiento de una base de datos en SQL Server se refiere a cómo se organiza y guarda la información en disco. Los componentes principales son:

**Archivos de datos:** Son los archivos físicos donde se almacenan los datos y objetos de la base de datos. Pueden ser de dos tipos:

**Archivo primario (.mdf):** Contiene las tablas, índices y objetos principales de la base de datos.

**Archivos secundarios (.ndf):** Se utilizan para distribuir el almacenamiento de la base de datos en múltiples discos.

**Archivos de registro de transacciones (.ldf):** Guardan un registro de todas las transacciones realizadas en la base de datos para asegurar la recuperación y la consistencia de los datos.

SQL Server gestiona estos archivos para optimizar la lectura, escritura y recuperación de la información.

```
NAME = MiBaseDeDatos_Principal,  
FILENAME = 'C:\SQLData\MiBaseDeDatos_Principal.mdf',  
SIZE = 50MB,  
MAXSIZE = 500MB,  
FILEGROWTH = 10MB
```

## 2. Propiedades y configuraciones de bases de datos.

Las propiedades y configuraciones de una base de datos en SQL Server se refieren a los ajustes que determinan cómo se comporta la base de datos, como el tipo de recuperación, el nivel de aislamiento de las transacciones, el tamaño de los archivos y la configuración de seguridad. Algunas propiedades comunes incluyen:

**Modo de recuperación:** Define cómo se gestionan las transacciones y los respaldos de la base de datos (Full, Simple, Bulk-Logged).

**Tamaño de los archivos:** Especifica el tamaño inicial de los archivos de datos y los archivos de registro.

**Propiedades de colación:** Define el conjunto de reglas que SQL Server utiliza para comparar y ordenar cadenas de texto.

**Autocompletado de archivos:** Controla el comportamiento automático de la expansión de los archivos.

Estas configuraciones permiten personalizar la base de datos para optimizar su rendimiento y seguridad según las necesidades del sistema.

```
1 ALTER DATABASE MiBaseDeDatos
2 MODIFY FILE
3 (
4     NAME = MiBaseDeDatos_Principal,
5     FILEGROWTH = 20MB
6 );
```

### 3. Tipos de recuperación (Simple, Full, Bulk-Logged).

SQL Server ofrece tres **modos de recuperación** para determinar cómo se gestionan los registros de transacciones y las copias de seguridad:

- **Modo de recuperación Simple:** En este modo, los registros de transacciones se eliminan después de que la transacción se ha completado. No es posible realizar copias de seguridad de los registros de transacciones, y las restauraciones solo pueden realizarse hasta el último respaldo completo.
- **Modo de recuperación Completa (Full):** En este modo, se registran todas las transacciones, permitiendo la restauración a un punto exacto en el tiempo. Es ideal para bases de datos que requieren alta disponibilidad y precisión en las restauraciones.
- **Modo de recuperación Bulk-Logged:** Este modo es intermedio, diseñado para cargas masivas de datos. Ofrece un buen rendimiento durante operaciones de carga de grandes volúmenes de datos, pero no permite la recuperación a nivel de transacción.

```
SQLQuery1.sql - LA...PC01\USER 17 (73))* -p X
1 ALTER DATABASE MiBaseDeDatos
2 SET RECOVERY SIMPLE;
```

### 4. Administración de usuarios y roles de seguridad.

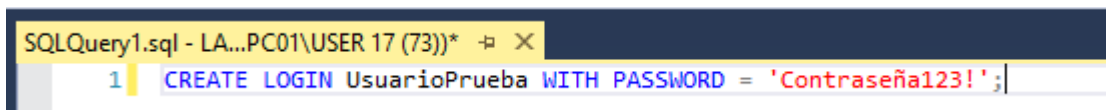
La administración de usuarios y roles en SQL Server se refiere a la creación y gestión de los usuarios que acceden a la base de datos y la asignación de roles que determinan qué acciones pueden realizar.

**Usuarios:** Son las cuentas que se utilizan para acceder a la base de datos.

**Roles:** Son conjuntos de permisos que se asignan a los usuarios para gestionar sus privilegios. Los roles pueden ser predefinidos (por ejemplo, db\_owner, db\_datareader, db\_datawriter) o personalizados.

**Autenticación:** SQL Server soporta dos tipos de autenticación: Autenticación de Windows (integrada) y Autenticación de SQL Server (cuando se gestionan las cuentas dentro del propio servidor de bases de datos).

El control de acceso adecuado asegura que solo los usuarios autorizados puedan interactuar con los datos de manera segura.

A screenshot of a SQL Server query window. The title bar reads 'SQLQuery1.sql - LA...PC01\USER 17 (73))' with standard window controls. The query text area shows a single line of SQL code: '1 CREATE LOGIN UsuarioPrueba WITH PASSWORD = 'Contraseña123!';'. The line number '1' is highlighted in a yellow margin.

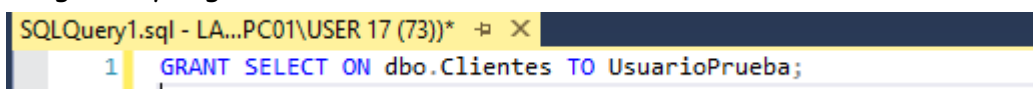
## 5. Asignación de permisos y políticas de acceso.

La asignación de permisos en SQL Server se refiere a la asignación de derechos específicos a usuarios o roles para que puedan realizar operaciones sobre objetos de la base de datos (tablas, vistas, procedimientos, etc.).

**Permisos:** Incluyen acciones como SELECT, INSERT, UPDATE, DELETE, EXECUTE, entre otros.

**Políticas de acceso:** Son reglas definidas para controlar el acceso a la base de datos, como restricciones de IP, limitaciones por hora, y el uso de cifrado o autenticación multifactor.

La correcta asignación de permisos y políticas de acceso es crucial para mantener la integridad y seguridad de los datos.

A screenshot of a SQL Server query window. The title bar reads 'SQLQuery1.sql - LA...PC01\USER 17 (73))' with standard window controls. The query text area shows a single line of SQL code: '1 GRANT SELECT ON dbo.Clientes TO UsuarioPrueba;'. The line number '1' is highlighted in a yellow margin.

## 6. Monitoreo básico con el SQL Server Activity Monitor.

El SQL Server Activity Monitor es una herramienta que permite monitorear el rendimiento y las actividades en tiempo real de un servidor SQL. Proporciona información sobre:

**Procesos activos:** Muestra las consultas y transacciones en ejecución.

**Uso de CPU y memoria:** Monitorea el rendimiento del servidor.

**Esperas de bloqueo:** Identifica los bloqueos que podrían estar afectando el rendimiento.

**Gráficos de rendimiento:** Visualiza el uso de recursos como la I/O de disco, el uso de la CPU, y el almacenamiento.

Es una herramienta útil para detectar cuellos de botella y problemas de rendimiento, ayudando a los administradores a tomar decisiones informadas.

```
SQLQuery1.sql - LA...PC01\USER 17 (73))* -p X
1 SELECT
2     session_id, login_name, status, cpu_time, memory_usage, program_name
3 FROM sys.dm_exec_sessions
4 WHERE is_user_process = 1;
```

## 7. Introducción al uso de SQL Server Agent (tareas automáticas)

El **SQL Server Agent** es un servicio que permite automatizar y programar tareas en SQL Server. Algunas de sus capacidades incluyen:

**Programar trabajos:** Permite ejecutar procedimientos almacenados, scripts SQL o tareas administrativas de forma automatizada y periódica.

**Alertas y notificaciones:** Puede configurar alertas para ciertos eventos (como errores o advertencias) y notificar a los administradores por correo electrónico.

**Mantenimiento:** Facilita tareas como la realización de copias de seguridad, la actualización de estadísticas, la reconstrucción de índices, entre otros.

SQL Server Agent es especialmente útil para la administración de bases de datos en entornos de producción, ya que permite mantener tareas recurrentes de manera eficiente y sin intervención manual.

```
SQLQuery1.sql - LA...PC01\USER 17 (73))*  X
1  USE msdb;
2  GO
3
4  -- Crear el trabajo
5  EXEC sp_add_job
6      @job_name = N'Backup_Automatico_MiBaseDeDatos';
7
8  -- Crear un paso del trabajo
9  EXEC sp_add_jobstep
10     @job_name = N'Backup_Automatico_MiBaseDeDatos',
11     @step_name = N'Copia de seguridad completa',
12     @subsystem = N'TSQL',
13     @command = N'BACKUP DATABASE MiBaseDeDatos
14                 TO DISK = ''C:\SQLBackups\MiBaseDeDatos.bak''
15                 WITH INIT, STATS = 10;',
16     @database_name = N'master';
17
```