

3.1 Definición de espacio de almacenamiento

Configuración de la Base de Datos Escuela

```
-- Crea una base de datos llamada "Escuela"
CREATE DATABASE Escuela
ON (
    -- Define el archivo de datos principal de la base de datos
    NAME = Escuela_Data,
    -- Especifica la ubicación del archivo físico en el sistema
    FILENAME = 'C:\BASES_SQL\Escuela.mdf',
    -- Establece el tamaño inicial del archivo de datos en 10MB
    SIZE = 10MB,
    -- Define el tamaño máximo que puede alcanzar el archivo de datos (100MB)
    MAXSIZE = 100MB,
    -- Indica el incremento en el tamaño del archivo de datos cuando sea necesario (5MB)
    FILEGROWTH = 5MB
)
LOG ON (
    -- Define el archivo de registro de transacciones (log)
    NAME = Escuela_log,
    -- Especifica la ubicación del archivo de log en el sistema
    FILENAME = 'C:\BASES_SQL\Escuela_log.ldf',
    -- Establece el tamaño inicial del archivo de log en 5MB
    SIZE = 5MB,
    -- Define el tamaño máximo que puede alcanzar el archivo de log (50MB)
    MAXSIZE = 50MB,
    -- Indica el incremento en el tamaño del archivo de log cuando sea necesario (1MB)
    FILEGROWTH = 1MB
);
```

Creación de la Base de Datos Escuela

```
-- Selecciona la base de datos "Escuela" para ejecutar las siguientes operaciones
USE Escuela;
GO

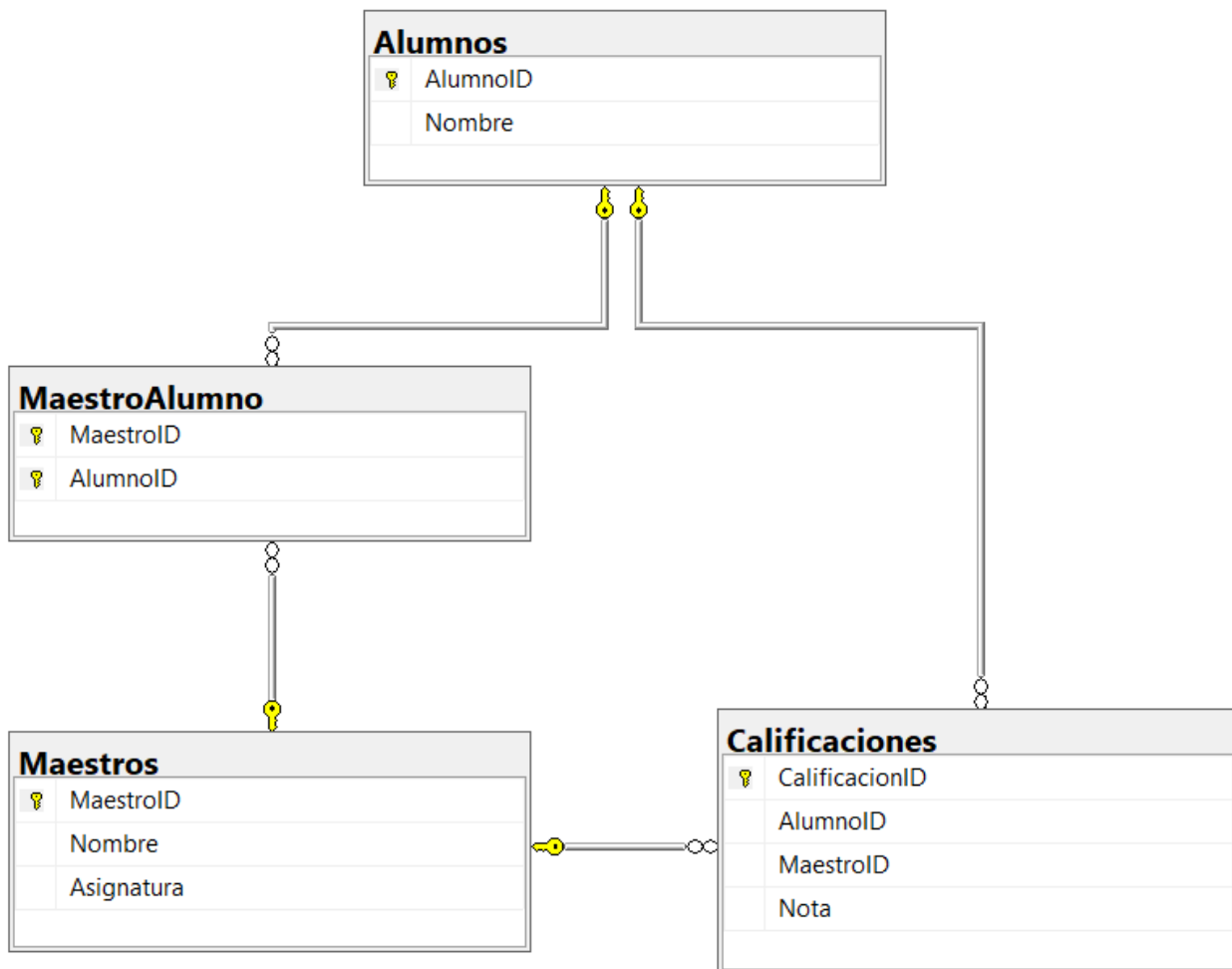
-- Crear la tabla "Maestros" para almacenar los datos de los profesores
CREATE TABLE Maestros(
    -- Identificador único para cada maestro, con incremento automático
    MaestroID INT PRIMARY KEY IDENTITY(1,1),
    -- Nombre del maestro, campo obligatorio
    Nombre NVARCHAR(100) NOT NULL,
    -- Asignatura que imparte el maestro, campo obligatorio
    Asignatura NVARCHAR(100) NOT NULL
);
GO

-- Crear la tabla "Alumnos" para almacenar los datos de los estudiantes
CREATE TABLE Alumnos (
    -- Identificador único para cada alumno, con incremento automático
    AlumnoID INT PRIMARY KEY IDENTITY(1,1),
    -- Nombre del alumno, campo obligatorio
    Nombre NVARCHAR(100) NOT NULL
);
GO

-- Crear la tabla "Calificaciones" para registrar las notas de los alumnos
CREATE TABLE Calificaciones(
    -- Identificador único de la calificación, con incremento automático
    CalificacionID INT PRIMARY KEY IDENTITY(1,1),
    -- Identificador del alumno que recibe la calificación (clave foránea)
    AlumnoID INT,
    -- Identificador del maestro que asigna la calificación (clave foránea)
    MaestroID INT,
    -- Nota obtenida por el alumno, campo obligatorio con formato decimal (máximo 5 dígitos, 2 decimales)
    Nota DECIMAL(5,2) NOT NULL,
    -- Clave foránea que relaciona la tabla "Calificaciones" con la tabla "Alumnos"
    FOREIGN KEY (AlumnoID) REFERENCES Alumnos(AlumnoID),
    -- Clave foránea que relaciona la tabla "Calificaciones" con la tabla "Maestros"
    FOREIGN KEY (MaestroID) REFERENCES Maestros(MaestroID)
);
GO

-- Crear la tabla intermedia "MaestroAlumno" para representar la relación muchos a muchos entre maestros y alumnos
CREATE TABLE MaestroAlumno (
    -- Identificador del maestro, clave foránea
    MaestroID INT,
    -- Identificador del alumno, clave foránea
    AlumnoID INT,
    -- Clave primaria compuesta para evitar duplicados en la relación
    PRIMARY KEY (MaestroID, AlumnoID),
    -- Clave foránea que relaciona la tabla "MaestroAlumno" con la tabla "Maestros"
    FOREIGN KEY (MaestroID) REFERENCES Maestros(MaestroID),
    -- Clave foránea que relaciona la tabla "MaestroAlumno" con la tabla "Alumnos"
    FOREIGN KEY (AlumnoID) REFERENCES Alumnos(AlumnoID)
);
```

Diagrama Entidad-Relación Escuela



Preguntas de Reflexión

PREGUNTAS DE REFLEXIÓN

13/Marzo/2025

1. ¿Cuáles son las diferencias entre el archivo de datos y el archivo de registro?

- Su propósito: El archivo de datos almacena los datos reales de la base de datos y el Archivo de registros, registra todas las transacciones y cambios realizados en la base de datos.
- El Archivo de datos no se utiliza directamente para la recuperación y el archivo de Registros se usa para recuperar la base de datos en caso de fallo.

2. ¿Porqué es importante establecer tamaños máximos y opciones de autocrecimiento?

Establecer tamaños máximos es importante porque optimiza el uso del espacio de almacenamiento, previene errores al ingresar datos demasiado grandes y mejora el rendimiento de consultas e índices.

El autocrecimiento es importante para generar automáticamente claves primarias únicas, reduce errores humanos en la asignación de claves y facilita el diseño y asegura la integridad referencial.