

Laboratorio 2: Creación de tablas y gestión de datos

Objetivo: Crear tablas con restricciones en SQL y gestionar sus datos.

Instrucciones: Este laboratorio lo pueden trabajar en parejas, si lo desean. En tal caso, harían una sola entrega por pareja, indicando quiénes trabajaron juntos en el reporte, pero ambos deben correr el script final en su base de datos *personal*, pues es posible que requiera trabajar sobre ella en subsiguientes laboratorios o *quices*.

Guía de Trabajo

1. Abra la aplicación SQL Server Management Studio. En la ventana de conexión seleccione las opciones 'Database Engine' y 'SQL Server Authentication'. En el campo 'Nombre del servidor' escriba la dirección IP 172.16.202.209.
2. Use la base de datos que tiene por nombre su **número de carné**. Recuerde escribir al inicio de la consulta "use <Nombre de la BD>" y ejecutar este comando.
3. Implemente el esquema relacional simplificado de la BD Universidad (ver [Figura 1](#) al final de esta guía y su correspondiente diagrama ER en la [Figura 2](#)). Note que este esquema es más pequeño que el usado en el laboratorio anterior. Puede escoger el tipo de datos de los atributos según consideren más conveniente y razonable (revise la diferencia entre *char* y *varchar*). Para crear las tablas y sus restricciones, use los comandos DDL (*Data Definition Language*) de SQL y siga estas indicaciones:
 - a. Todas las tablas deben especificar la restricción de **llave primaria**. Tome en cuenta que si la llave primaria es compuesta, la restricción debe colocarse después de la definición de todos los atributos (y no al lado de cada atributo primo, pues sino daría un error por tener más de una llave primaria).
 - b. Las tablas que posean **llaves externas** deben especificar esa restricción.
 - i. La tabla `Grupo` debe tener la restricción **NOT NULL** en la llave externa `CedProf`. Esta restricción corresponde a la participación total entre Grupo e Imparte ([Figura 2](#)).
 - ii. La tabla `Empadronado_en` debe tener la restricción **ON DELETE CASCADE** en su llave externa `CedEstudiante`.
 - iii. La tabla `Grupo` debe tener la restricción **ON DELETE NO ACTION** en su llave externa `SiglaCurso`. (Por omisión en SQL Server, toda restricción es NO ACTION).
 - iv. La tabla `Grupo` debe tener la restricción **ON UPDATE CASCADE** en su llave externa `CedProf`.
 - c. La tabla `Grupo` debe tener la restricción **DEFAULT** para el atributo `Carga`, de manera que por omisión sea 0 (cero).
 - d. La tabla `Lleva` debe tener una restricción **CHECK** para el dominio del atributo `Nota`, de manera que solo se permitan valores entre 0 y 100.

4. **Inserte** al menos **2** tuplas en cada una de las tablas de la [Figura 1](#), mediante comandos DML (*Data Manipulation Language*) de SQL. Tome en cuenta las relaciones que deben existir entre datos de diferentes tablas, según las restricciones impuestas por las llaves externas (integridad referencial). Incluya en el reporte de laboratorio los comandos SQL usados.
5. **Explique** el **comportamiento** de las **restricciones** ON DELETE CASCADE, ON DELETE NO ACTION y ON UPDATE CASCADE creadas en el punto 3(b) sobre llaves externas.
 - a. Haga una consulta sobre la tabla “referenciada” por la llave externa en cuestión, tal que muestre al menos la llave primaria de esta tabla. Incluya en el reporte el comando SQL de la consulta y una captura de pantalla con su resultado.
 - b. Haga una consulta sobre la tabla que contiene la llave externa en cuestión, tal que muestre al menos la llave primaria y la llave externa de esta tabla. Incluya en el reporte el comando SQL de la consulta y una captura de pantalla con su resultado.
 - c. Borre (o modifique, según corresponda) una tupla de la tabla “referenciada” tal que accione la restricción en cuestión (ON DELETE CASCADE, ON DELETE NO ACTION u ON UPDATE CASCADE). Incluya en el reporte de laboratorio el comando SQL ejecutado y una captura de pantalla del resultado mostrado en el panel *Output*.
 - d. Repita los pasos (a) y (b) de tal manera que el efecto del paso (c) sea observable¹. Use las mismas consultas que usó en los pasos (a) y (b) para comparar el estado de las tablas antes y después del comando que acciona la restricción.
 - e. Ofrezca una explicación del comportamiento observado, con base en los estados de las tablas involucradas antes y después de ejecutar el comando que acciona la restricción.
6. Envíen su trabajo a través de la plataforma virtual del curso. En esta ocasión, deben hacer el reporte en algún editor de documentos que permita insertar imágenes y genere un archivo en formato *pdf*. Para la pregunta 5, se espera que presente el comando ejecutado y seguidamente su resultado, para cada una de las partes (a)-(d). Adicionalmente a este *pdf*, deben subir el archivo *.sql* con comentarios que indiquen la sección de esta guía a la que corresponde a cada comando SQL. Verifique que el script se ejecute sin errores.

¹ Puede que necesite agregar más datos a las tablas involucradas para observar el comportamiento esperado.

1. **ESTUDIANTE** (Cédula, Email, NombreP, Apellido1, Apellido2, Sexo, FechaNac, Dirección, Teléfono, Carné, Estado)
2. **PROFESOR** (Cédula, Email, NombreP, Apellido1, Apellido2, Sexo, FechaNac, Dirección, Teléfono, Categoría, FechaNomb, Título, Oficina)
3. **ASISTENTE** (Cédula, NumHoras)
FK(Estudiante)
4. **CURSO** (Sigla, Nombre, Créditos)
5. **GRUPO** (SiglaCurso, NumGrupo, Semestre, Año, CedProf, Carga, CedAsist)
FK(Curso) FK(Profesor) FK(Asistente)
6. **LLEVA** (CedEstudiante, SiglaCurso, NumGrupo, Semestre, Año, Nota)
FK(Estudiante) FK(Gruo)
7. **CARRERA** (Código, Nombre, AñoCreación)
8. **EMPADRONADO_EN** (CedEstudiante, CodCarrera, FechaIngreso, FechaGraduación)
FK(Estudiante) FK(Carrera)
9. **PERTENECE_A** (SiglaCurso, CodCarrera, NivelPlanEstudios)
FK(Curso) FK(Carrera)

Figura 1. Esquema relacional *simplicado* de la BD Universidad.

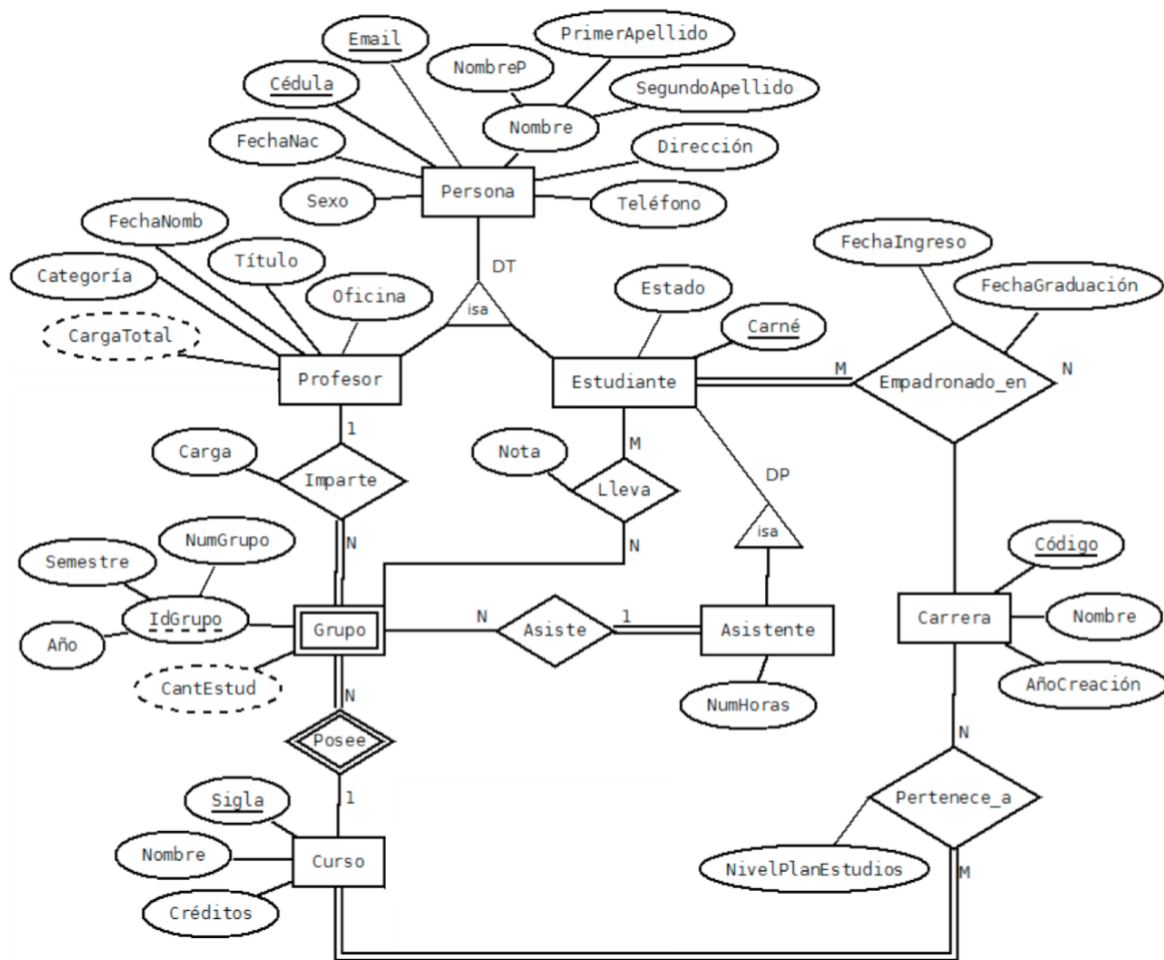


Figura 2. Diagrama ER *simplificado* de la BD Universidad.