

Laboratorio 4: Funciones y procedimientos almacenados

Objetivo: Crear y usar funciones y procedimientos almacenados en SQL Server.

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* en su base de datos *personal*, para trabajar sobre ella en subsiguientes laboratorios.

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' y escriba la dirección IP 172.16.202.209 en el campo 'Nombre del servidor'.
2. Use la base de datos que tiene por nombre su **número de carné**. Recuerde escribir al inicio del *script* "use <Nombre de la BD>" y ejecutar dicho Use la base de datos que tiene por nombre su **número de carné** comando.
3. Con base en el esquema relacional simplificado de la BD Universidad (ver [Figura 1](#) y su correspondiente diagrama ER en la [Figura 2](#)), haga lo siguiente:

- a. [33 pts.] Programe un **procedimiento almacenado** llamado "EmpadronarEstudiante" que empadrona a un estudiante en una carrera. Es decir, debe **insertar** una nueva tupla en la tabla Empadronado_En con base en los siguientes parámetros de entrada: la cédula del estudiante y el código de la carrera. Es posible que necesite parámetros adicionales, dependiendo de los campos que haya definido como *not null* cuando creó la tabla. Invoque el procedimiento y verifique que funciona correctamente. Puede invocar el procedimiento mediante este comando:

```
EXEC EmpadronarEstudiante @cod=codigo, @ced=cedula //en desorden
```

- b. [34 pts.] Programe una **función almacenada** llamada "CreditosPorSemestre" que **consulte** la cantidad de créditos matriculados en un semestre por un estudiante. Los parámetros de entrada son la cédula del estudiante, el semestre y el año (un "semestre" se define como el semestre y el año, por ejemplo, el I semestre del 2020, o el II semestre del 2019). El resultado de la consulta debe ser la cantidad total de créditos que el estudiante matriculó cierto semestre. Invoque la función y verifique que funciona correctamente. Puede invocar la función con este comando:

```
SELECT dbo.CreditosMatriculadosPorSemestre('123456789', 'I', 2020) AS  
'creditos_seestrales'
```

- c. [33 pts.] Programe un **procedimiento almacenado** llamado "ActualizarCreditos" que aumente, en un porcentaje dado, los créditos de los cursos cuyo nombre contenga una hilera dada por parámetro. Los parámetros de entrada son la hilera a buscar (dentro del

nombre del curso) y el porcentaje de aumento. El resultado de invocar este procedimiento debe ser que el creditaje de cada curso que calce con la hilera aumente en un P% (por ejemplo, en un 30%, o en un 100%). Por ejemplo: si la hilera dada por parámetro es “bases”, usted debe aumentar el creditaje de todos los cursos cuyo nombre contenga la palabra “bases”. Si usted definió el atributo créditos (en el Lab2) como un entero, entonces redondee el resultado al entero más cercano. Invoque el procedimiento y verifique que funcione correctamente.

4. Envíe su trabajo a través de la plataforma virtual del curso. (Si trabajaron en pareja, el reporte lo debe subir sólo uno de los dos). Suba el archivo *sql* con comentarios que indiquen la sección de esta guía a la que corresponde a cada función o procedimiento. Recuerde incluir también el comando de invocación de cada función o procedimiento. Verifique que el *script* se ejecute sin errores.

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(Grupo)
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 *simplificado* de la BD Universidad.

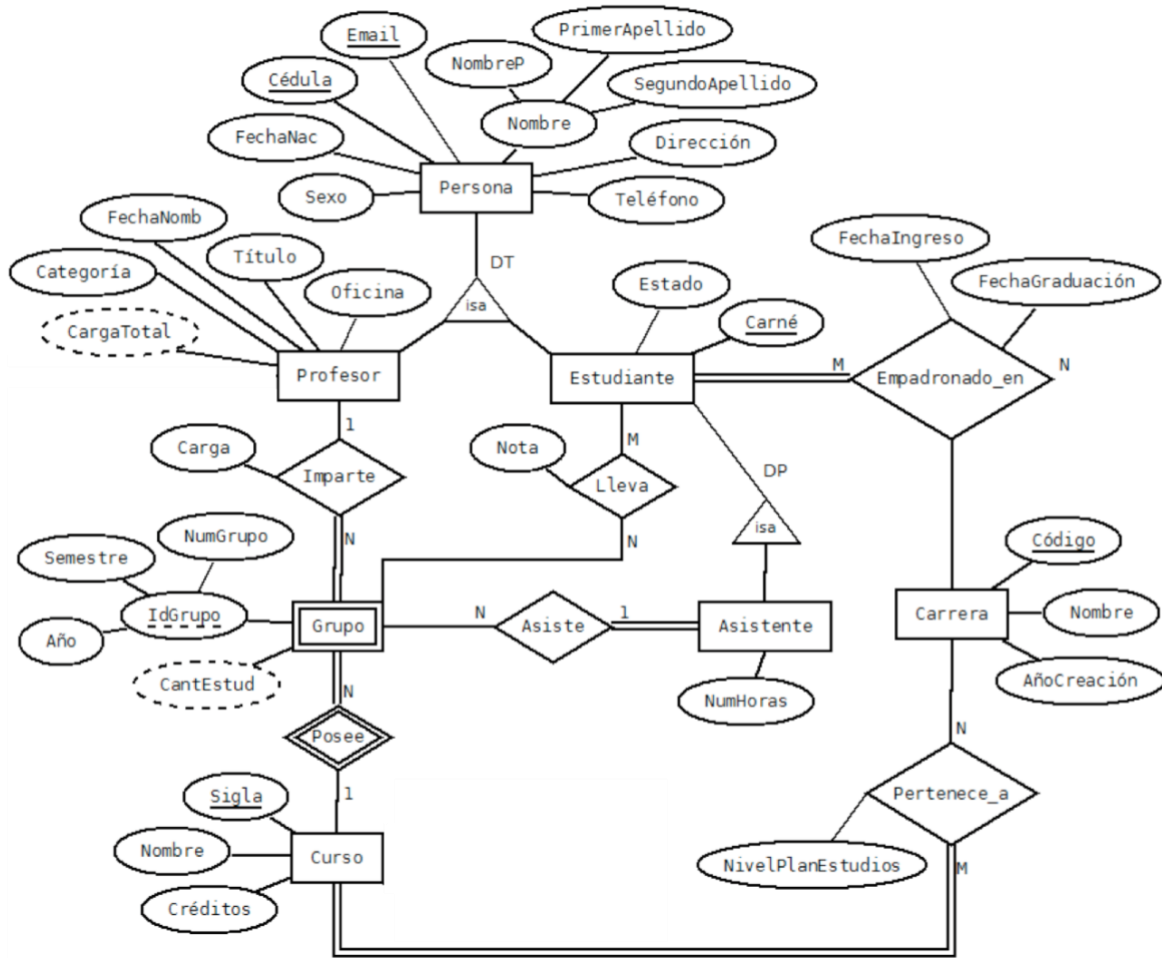


Figura 2. Diagrama ER *simplificado* de la BD Universidad.