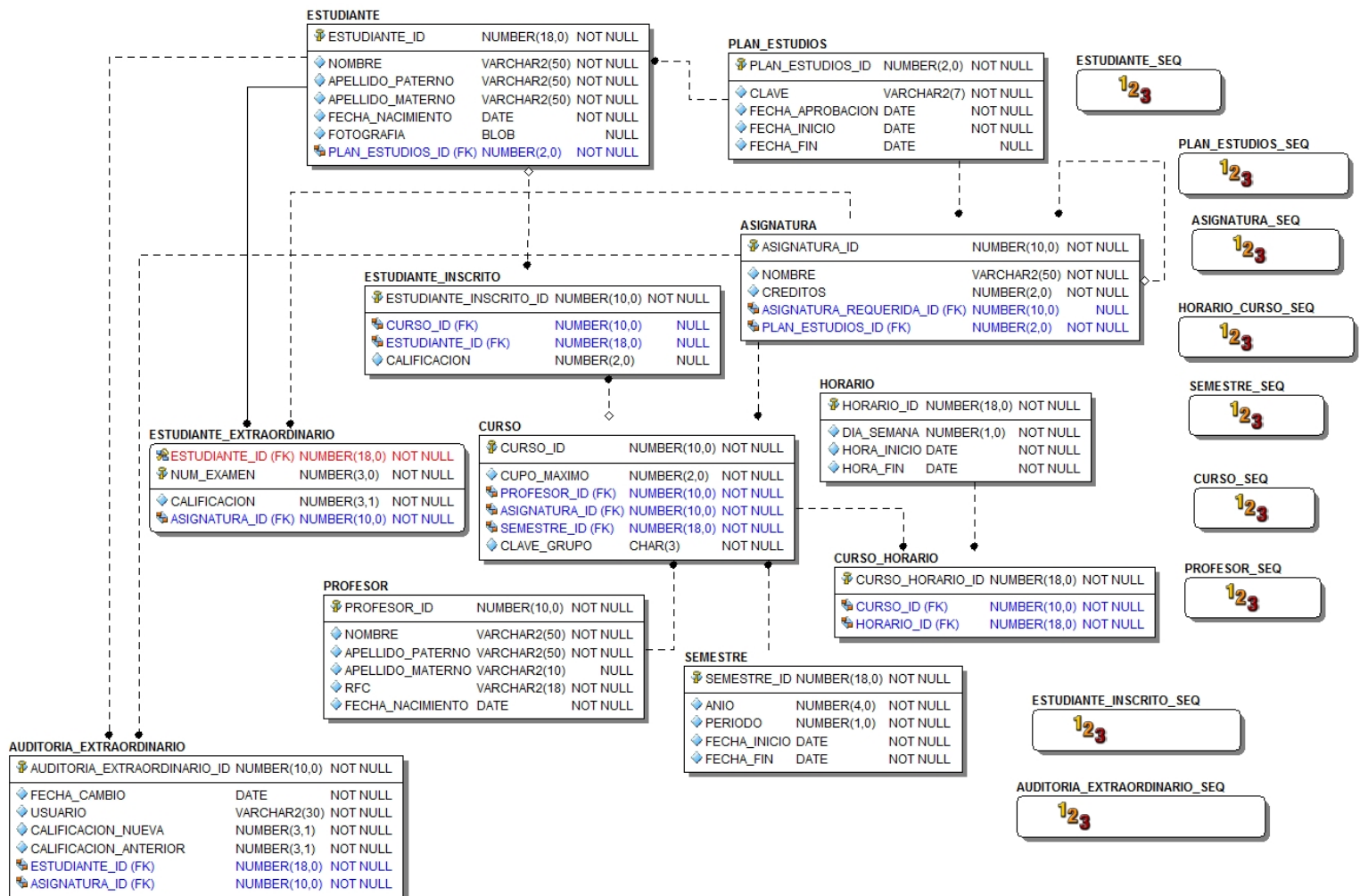


TEMA 10.  
PROGRAMACIÓN DE SCRIPTS EN SQL

Antes de realizar los ejercicios de este documento realizar las siguientes acciones:

1. Abrir una terminal, levantar el *listener* y la *instancia* de la BD.
2. Obtener los archivos SQL que se encuentran en la carpeta compartida BD correspondiente al tema 10. Abrir los archivos para revisar su contenido.
3. Abrir una nueva terminal y cambiarse al directorio donde se encuentran los archivos.
4. Entrar a `sqlplus`, no es necesario autenticarse. Es decir, ejecutar `sqlplus /nolog`
5. Ejecutar el archivo `s-01-ejercicios-tema10-main.sql`. El script solicitará los datos de un nuevo usuario:  
`sql> start s-01-ejercicios-tema10-main.sql`

Los scripts anteriores generarán nuevamente el esquema de control escolar revisado en el tema 9 con algunas tablas adicionales:



## 1.1. TRIGGERS.

### 1.1.1. Ejercicio 1

Generar un trigger llamado `tr_valida_extraordinarios`. El trigger deberá validar las siguientes reglas de negocio:

- Un estudiante no puede registrar mas de 10 extraordinarios, de intentarlo, el trigger deberá enviar un mensaje de error.
- No se permiten eliminar registros de exámenes extraordinarios anteriores.
- En caso de aplicar una actualización de la calificación de un examen se deberá guardar un registro de auditoria en `auditoria_extraordinario`

Revisar y ejecutar el archivo `s-04-extraordinarios-prueba.sql` para verificar los resultados.

## 1.2. PROCEDIMIENTOS ALMACENADOS.

### 1.2.1. Ejercicio 2

Realice un procedimiento llamado `p_inscribe_estudiante`. El procedimiento deberá aceptar los siguientes parámetros:

```
estudiante_id
semestre_id
asignatura_id
clave_grupo
curso_id (out)
resultado (out)
```

El procedimiento deberá validar si el estudiante indicado en el parámetro puede inscribirse en el grupo, asignatura y semestre indicados por los parámetros.

El sistema puede mandar los siguientes mensajes cuando no sea posible inscribir al alumno:

- El estudiante no se puede inscribir porque ya ha cursado la materia con promedio  $\geq 6.0$  en el semestre indicado o en algún otro semestre (error -1)
- El estudiante no se puede inscribir porque ya está inscrito en el semestre actual. (error -2)
- El estudiante no puede ser inscrito porque ya no hay cupo en el grupo solicitado (error -3). En este caso, el procedimiento deberá mostrar una lista de los grupos que si tienen cupo (hacer uso de un cursor).

En caso de no existir errores de validación el procedimiento deberá inscribir al alumno en el grupo solicitado.

Las variables de salida `curso_id` y `resultado` deberán contener los siguientes valores:

- `curso_id` contendrá el identificador en caso de haber sido inscrito.
- `Resultado` contendrá el valor 0 si la inscripción fue exitosa, o el valor negativo indicado anteriormente en caso de error.

Ejecutar el archivo `s-05-inscribe_estudiante-prueba.sql` para verificar los resultados.

### 1.2.2. Ejercicio 3

Se desea construir un procedimiento almacenado el cual se ejecuta al final de cada semestre para obtener un reporte que muestre para cada asignatura el grupo que ha tenido el mayor número de inscritos. El reporte deberá contener las siguientes columnas:

Semestre_id	Nombre de la asignatura	nombre de la asignatura antecedente (si existe)	número de alumnos inscritos

- El resultado del reporte se deberá guardar en una tabla llamada `resultado_reporte`. Cabe destacar que el procedimiento debe verificar si la tabla existe. En caso negativo, se deberá crear. En caso de existir, se deberán eliminar todos los registros del semestre solicitado e insertar los nuevos valores.
- Al final, el procedimiento deberá mostrar el contenido de la tabla `resultado_reporte`. El procedimiento recibe como parámetro el identificador del semestre a consultar y un parámetro `num_registros` de salida en el que se debe almacenar el número total de registros. No olvidar otorgar permisos para poder crear procedimientos al usuario del caso de estudio control escolar.
- Ejecutar el archivo `s-06-ejecuta_reporte_inscritos-prueba.sql` para verificar los resultados.