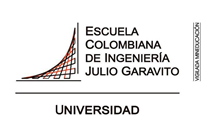
****

**Escuela Colombiana de Ingeniería Julio Garavito**

**Modelos y bases de datos 2024-1**

**Laboratorio IV**

**Cristian Santiago Pedraza Rodríguez**

**Andersson David Sánchez Méndez**

**22 de marzo de 2024**

# 

# MODELOS Y BASES DE DATOS

**Diseño Lógico. Procedimiental. SQL - DDL, DML 2024-1**

# Laboratorio 4/6

## OBJETIVOS

Evaluar el logro de las competencias adquiridas para:

1. Diseñar el modelo conceptual para una organización
2. Diseñar el modelo lógico correspondiente a un modelo conceptual
3. Implementar una base de datos relacional a partir de un diseño conceptual y lógico
4. Implementar los mecanismos procedimentales necesarios para garantizar integridad
5. Poblar la base de datos con información consistente
6. Usar un ambiente de desarrollo de bases de datos (SQL Developer)

## ENTREGA

Publiquen los resultados en un archivo .zip , el nombre de este archivo debe ser la concatenación en orden alfabético de los primeros apellidos de cada uno de los miembros. Deben entregar los archivos: lab04.doc, trueques.asta y trueques.sql Para organizar el archivo .sql incluyan como comentarios los títulos y subtítulos señalados y no olviden dar nombres significativos a cada uno de los elementos de la implementación.

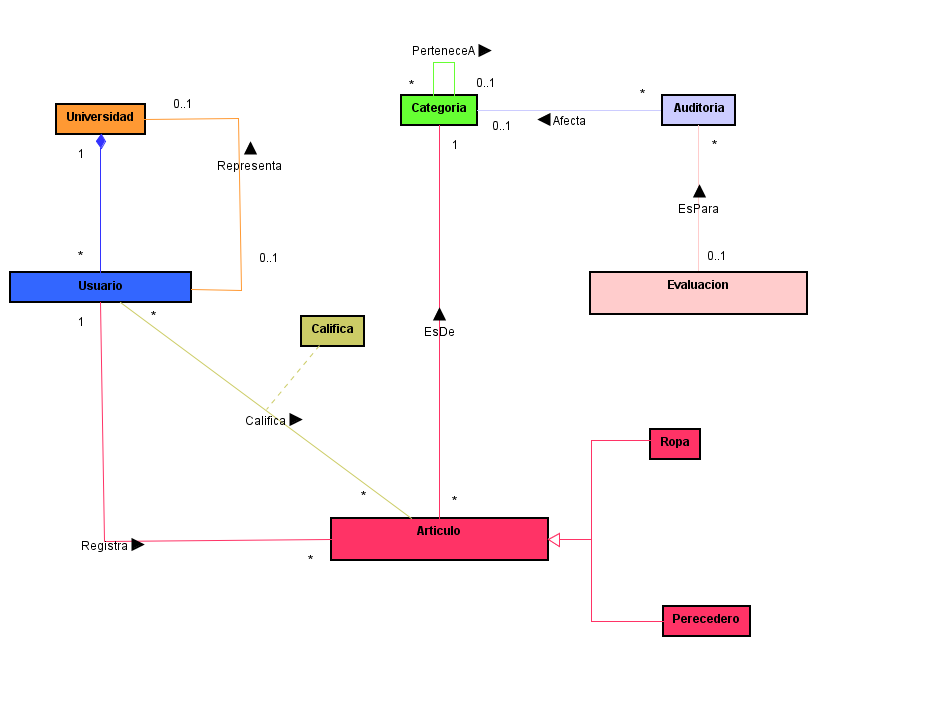
### PUNTO UNO. REFACTORIZACIÓN CICLO.

**Ciclo 1. trueques.**

En este punto vamos a perfeccionar el trabajo realizado: la estructura y las condiciones de integridad básicas del ciclo seleccionado. No olviden considerar los comentarios de la entrega anterior.

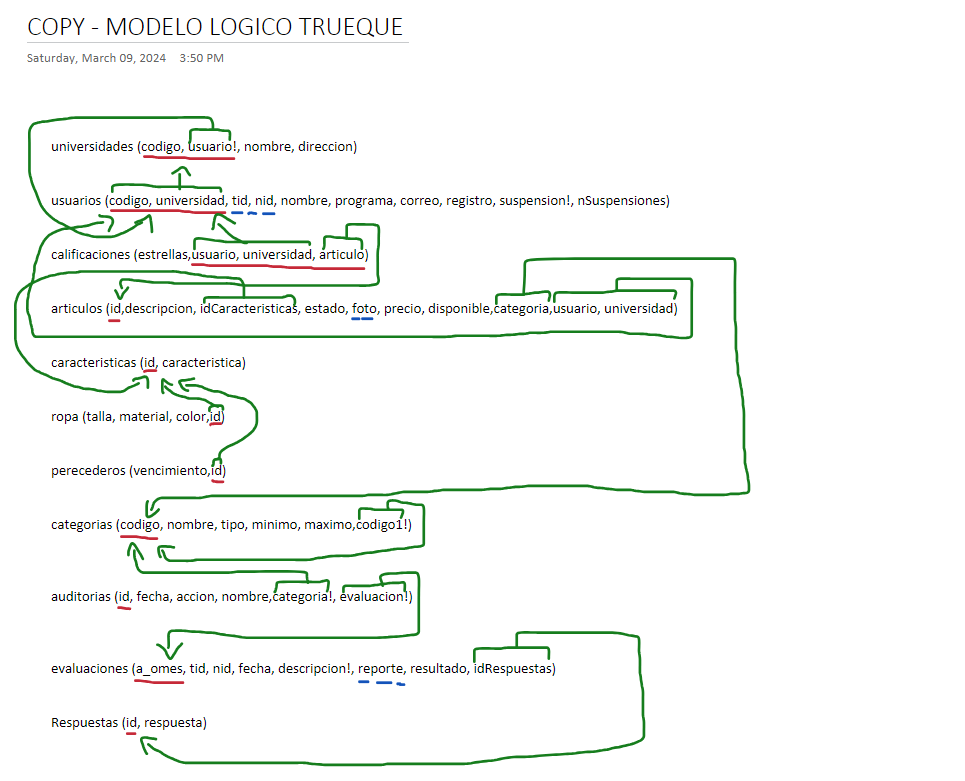
## Modelo conceptual

Revisen su modelo conceptual y perfecciónenlo. ¿Cuáles fueron los cambios realizados? No se realizaron cambios.



## Modelo lógico

Revisen su modelo lógico y perfecciónenlo. ¿Cuáles fueron los cambios realizados?



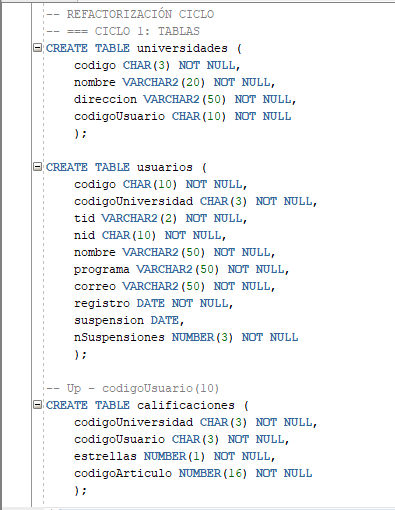
Se agregaron tablas (las de colección), hacer la relación fuerte bien, igual que añadir UK y atributos nulos, y arreglar varias relaciones.

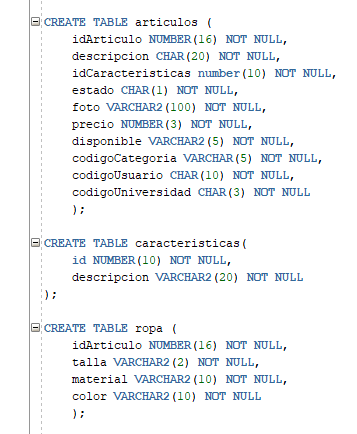
## Construcción

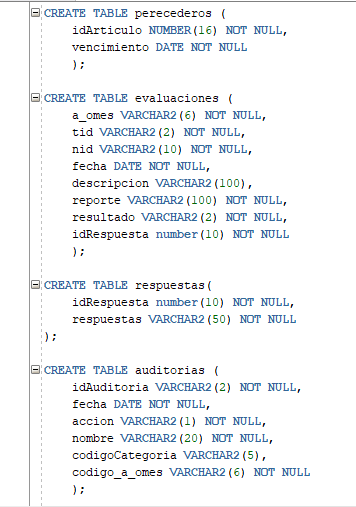
Revisen el código actual del sistema y perfecciónenlo. ¿Cuáles fueron los cambios realizados?

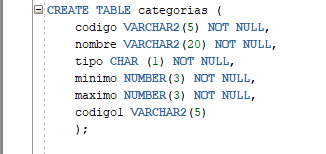
### El código debe: 1) estar en un sólo archivo 2) seguir el estándar de codificación 3) seguir el estándar para nombres de las restricciones

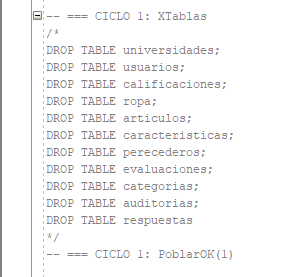
**Como el modelo lógico estaba mal implementado en algunas partes, entonces el cambio en la construcción del código también se notó.**

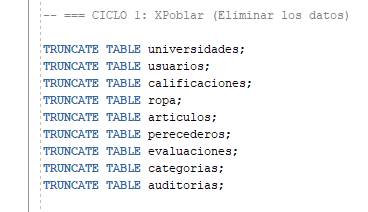
****

****

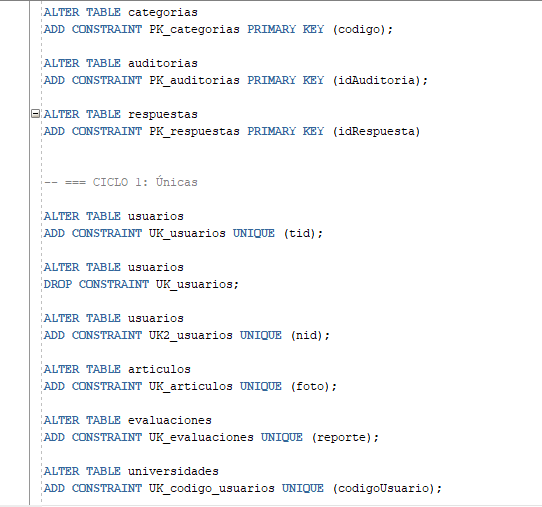
****

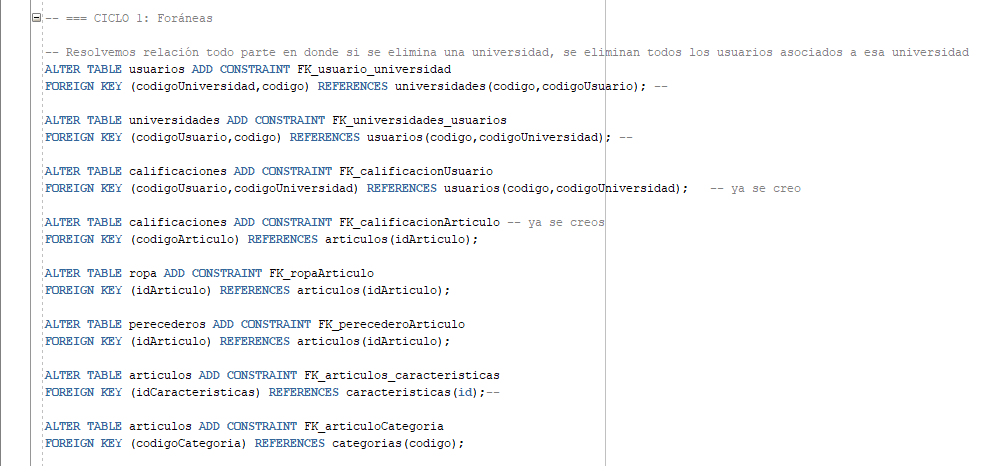
****

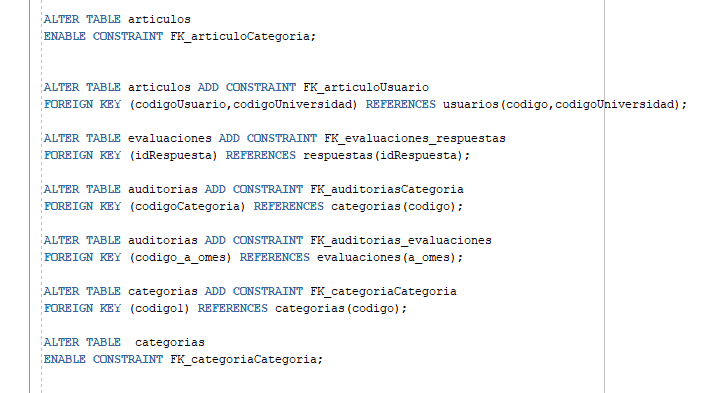
****

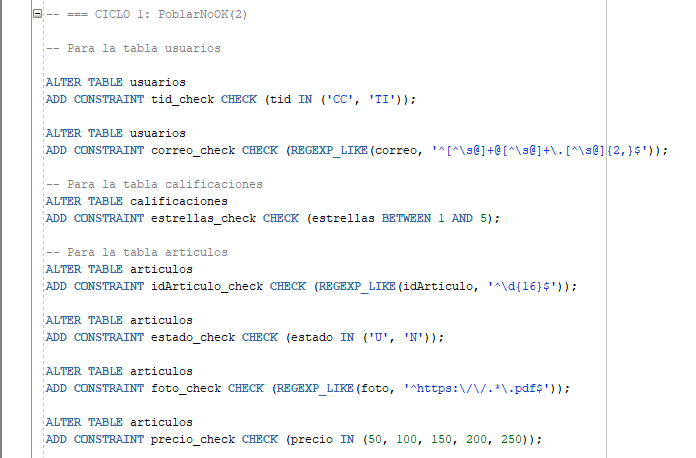
****

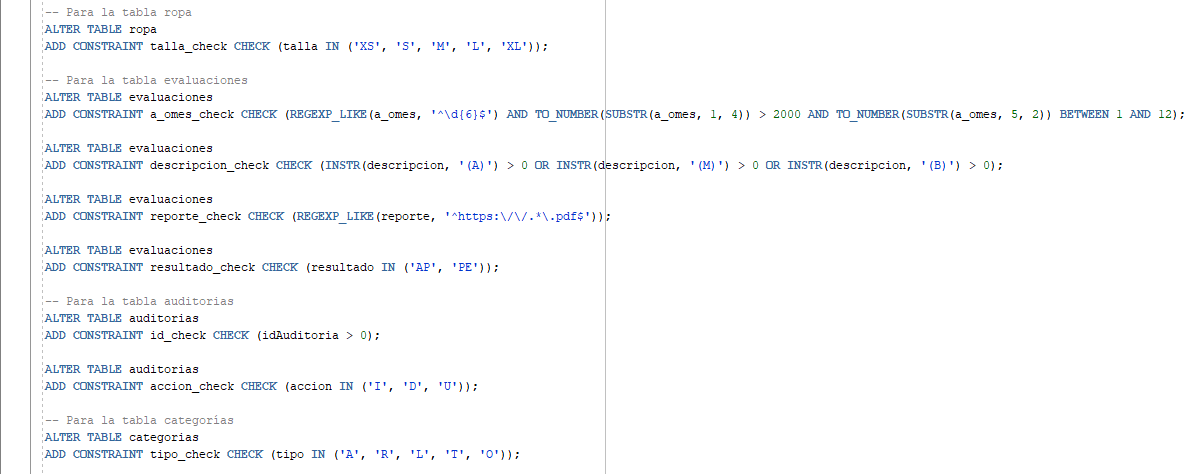
****

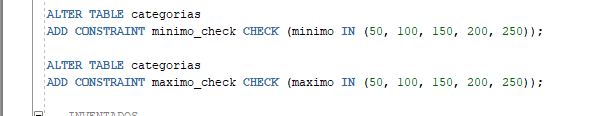
****

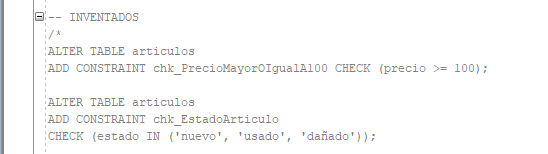
****

****

****

****

****

****

**PUNTO DOS. PREPARANDO CRUDs**

**Ciclo 1. trueques.**

En el laboratorio vamos a iniciar la construcción de dos casos de uso de este ciclo (i)

implementando restricciones (tanto declarativas como procedimentales) y (ii) automatizando acciones. Para cada uno de ellos sigan los pasos propuestos a continuación (A. B.) y documenten las fuentes como se indica.

TRABAJEN CASO POR CASO:

### es decir, implementen completamente un caso de uso antes de pasar al siguiente.

**Caso de uso 1: Mantener categoría + Generar registro de auditoría**

*Actualizar la especificación en astha*

*ADICIONAR*

*Los códigos deben iniciar con una letra mayúscula*

*Si no se indica el nombre se le asigna. ‘Nombre de ‘<codigo> El precio mínimo debe ser menor que el máximo*

*Si no se indica el precio máximo se supone que es el doble del mínimo MODIFICAR*

*Los únicos datos que se pueden modificar son el mínimo y el máximo. Únicamente pueden aumentar. Si se modifica el mínimo, el máximo debe modificarse en el mismo valor.*

*ELIMINAR*

*Únicamente se pueden eliminar los que no tienen artículos asociados.*

*En todos los escenarios se debe generar el registro de autoría.*

**Caso de uso 2: Registrar Evaluación**

*Revisar documentación en astah*

Estos casos de uso se implementaron abajo en parte 2 B en construcción con sus respectivos triggers o acciones referenciales o tuplas, es decir, restricciones declarativas o procedimentales.

## Modelo lógico

Definan el mecanismo apropiado para implementar las reglas de integridad asociadas al caso de uso. Escríbanlas[1](#_gjdgxs) en la descripción del modelo lógico siguiendo el ejemplo presentado en el autoestudio.

Por ejemplo, El nombre debe tener mínimo dos palabras. (CK\_CLIENTES\_NOMBRE). El número se asigna automáticamente TR\_FACTURA\_BI

**Caso de uso 1: Mantener categoría + Generar registro de auditoría**

**ADICIONAR**

* Los códigos deben iniciar con una letra mayúscula - CK\_CATEGORIA\_CODIGO ✓
* Si no se indica el nombre se le asigna. ‘Nombre de ‘<codigo> - TR\_CATEGORIA\_BI ✓
* El precio mínimo debe ser menor que el máximo CK\_CATEGORIA\_PRECIOMINIMO\_PRECIOMAXIMO ✓
* Si no se indica el precio máximo se supone que es el doble del mínimo - TR\_CATEGORIA\_BI ✓

**MODIFICAR**

* Los únicos datos que se pueden modificar son el mínimo y el máximo. Únicamente pueden aumentar. - TR\_CATEGORIA\_BU ✓
* Si se modifica el mínimo, el máximo debe modificarse en el mismo valor. - TR\_CATEGORIA\_AU ✓

**ELIMINAR**

* Únicamente se pueden eliminar los que no tienen artículos asociados. FK\_CATEGORIAS\_ARTICULOS\_ON\_DELETE\_RESTRICT ✓
* En todos los escenarios se debe generar el registro de auditoría TR\_CATEGORIA\_BD X

**Caso de uso 2: Registrar Evaluación**

**ADICIONAR**

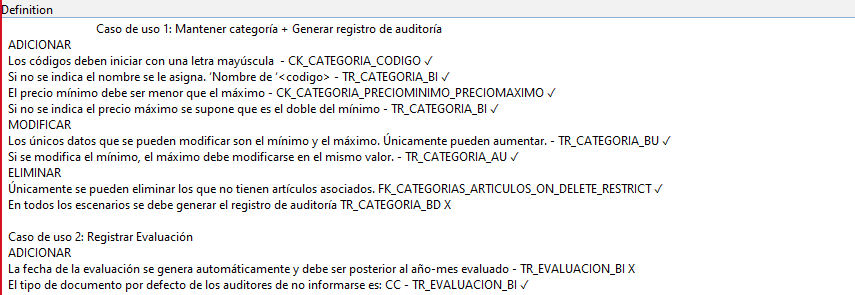
* La fecha de la evaluación se genera automáticamente y debe ser posterior al año-mes evaluado - TR\_EVALUACION\_BI X
* El tipo de documento por defecto de los auditores de no informarse es: CC - TR\_EVALUACION\_BI ✓
* Los registros asociados son los correspondientes al año-mes definido. CK\_EVALUACIONES\_AUDITORIAS X

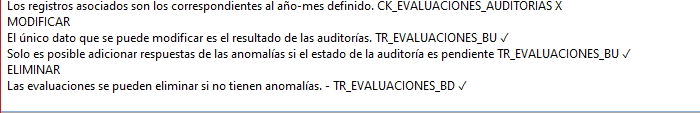
**MODIFICAR**

* El único dato que se puede modificar es el resultado de las auditorías. TR\_EVALUACIONES\_BU ✓
* Solo es posible adicionar respuestas de las anomalías si el estado de la auditoría es pendiente TR\_EVALUACIONES\_BU ✓

**ELIMINAR**

* Las evaluaciones se pueden eliminar si no tienen anomalías. - TR\_EVALUACIONES\_BD✓





## Construcción

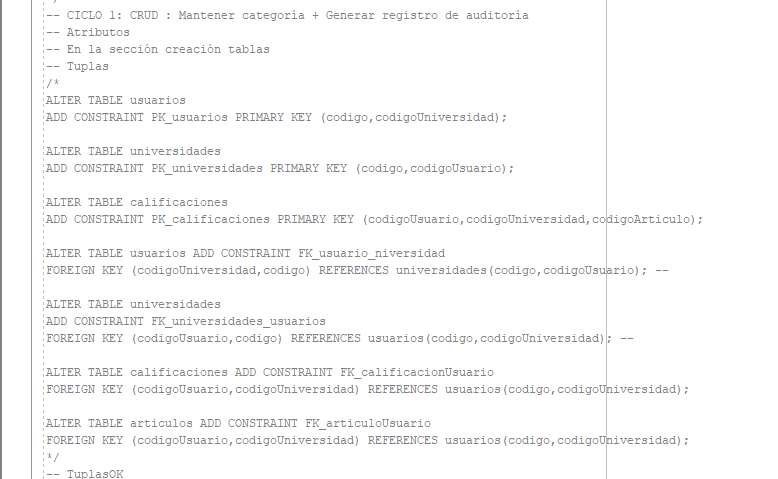
Implementen los componentes definidos en el paso anterior:

Escriba en comentarios siguiendo el siguiente esquema. Incluyan comentarios para explicar la intención de cada caso de NoOK

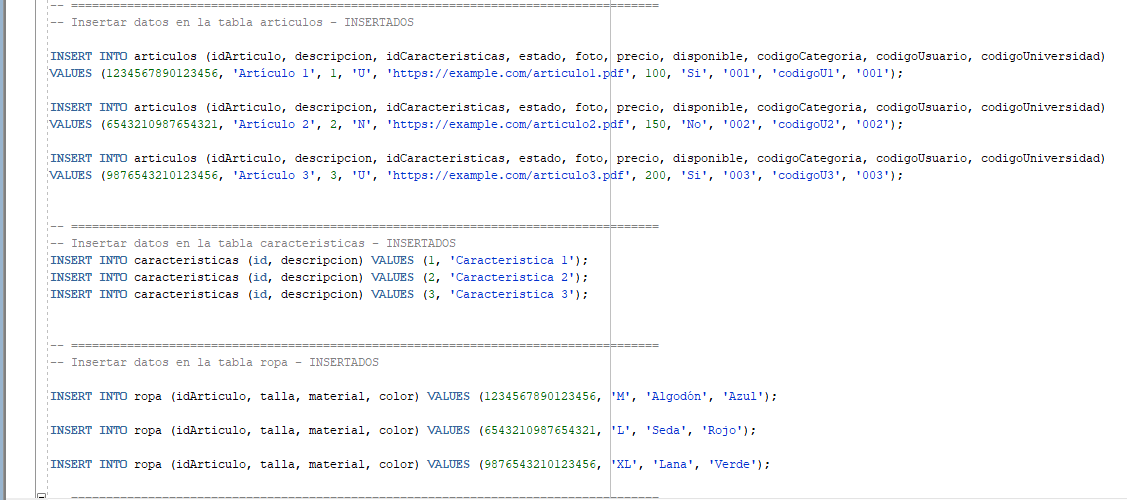
CICLO 1: CRUD :

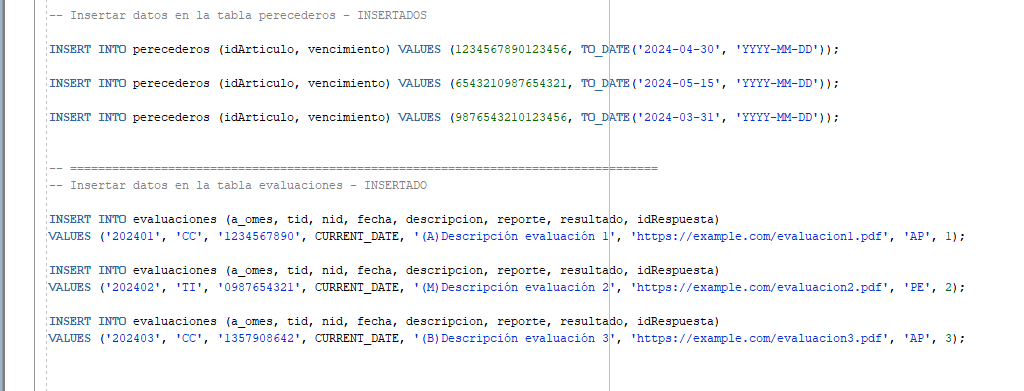
Atributos Tuplas TuplasOK TuplasNoOK Acciones AccionesOK Disparadores DisparadoresOK

DisparadoresNoOK Xdisparadores

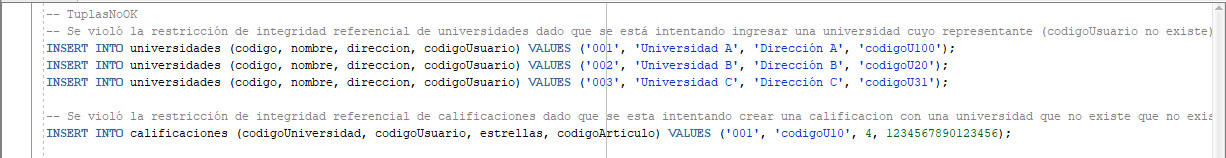


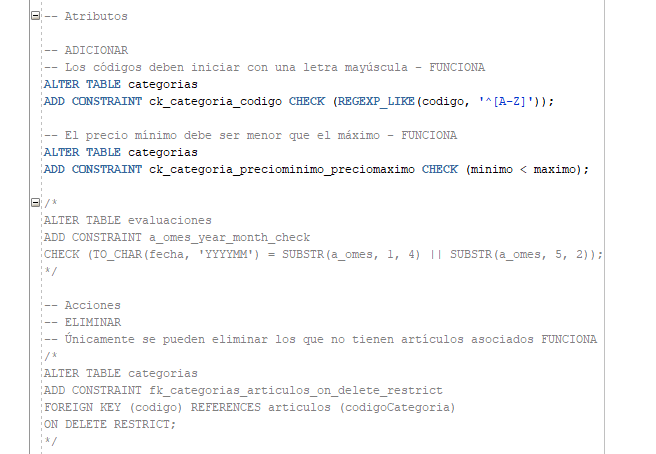


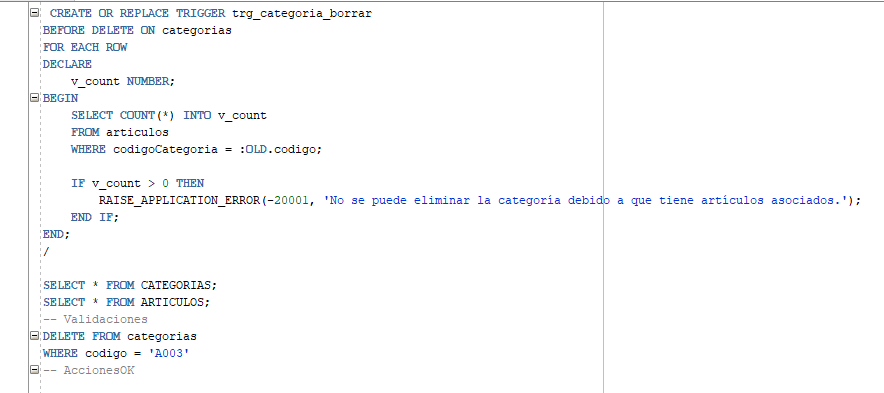


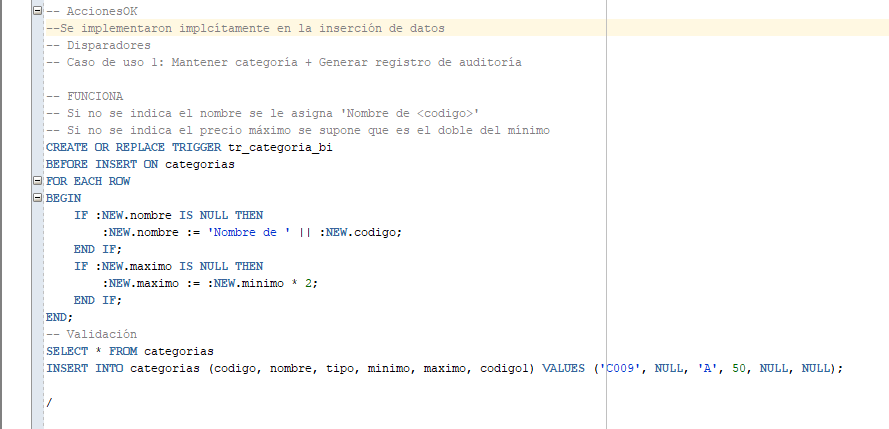


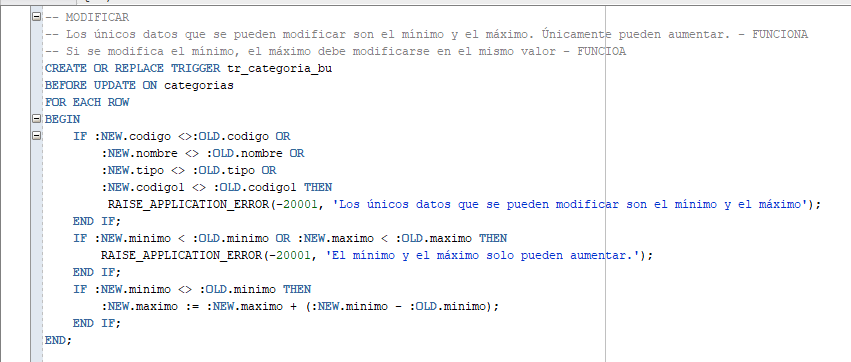


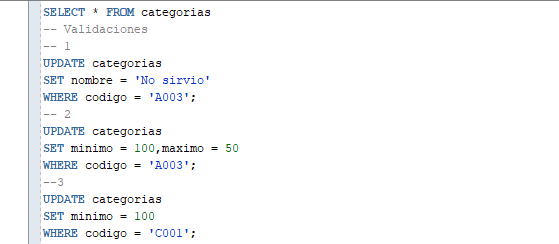


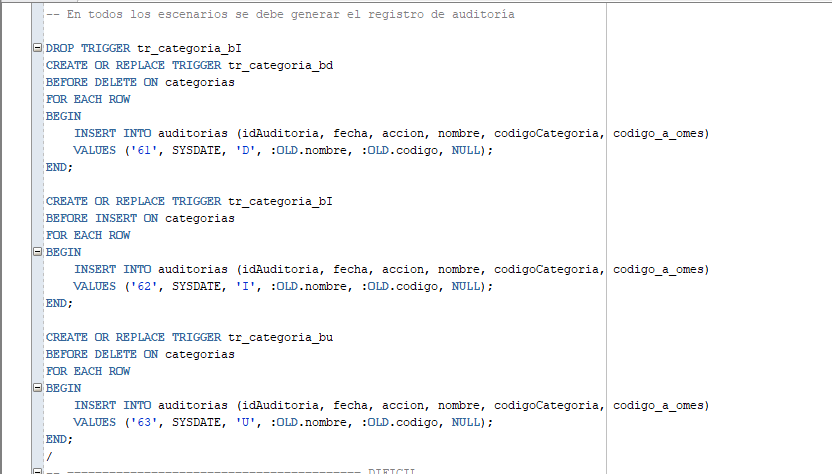


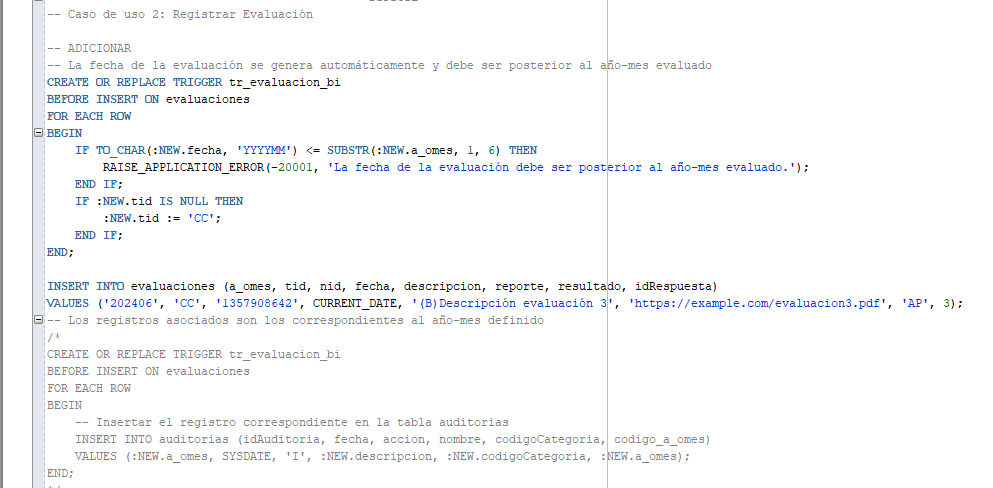


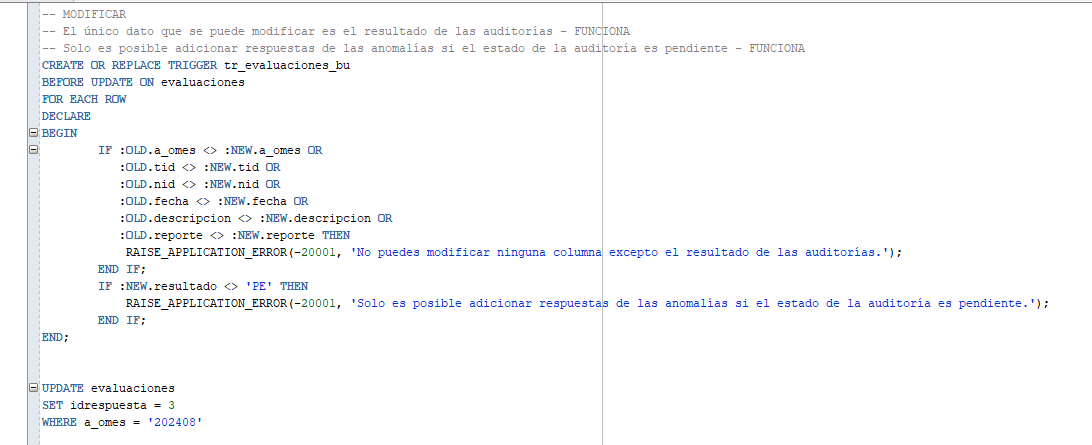


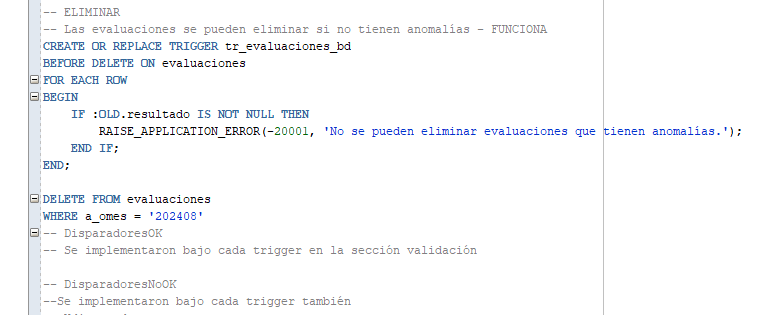


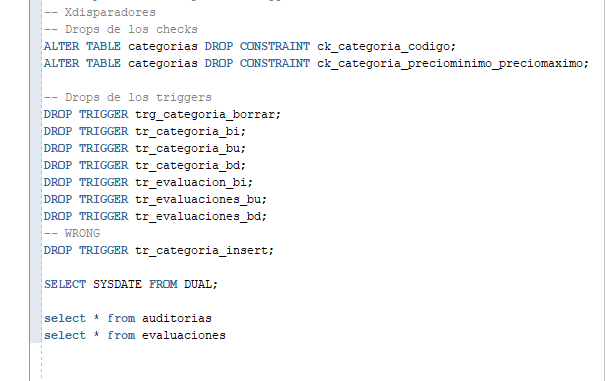












### PUNTO TRES. REFACTORIZACIÓN GENERAL.

Revisen su modelo conceptual general y perfecciónenlo. ¿Cuáles fueron los cambios realizados?

### RETROSPECTIVA

* 1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes? (Horas/Hombre)

Hasta ahora este ha sido el laboratorio al que más se le ha dedicado tiempo, le dedicamos más de 20 horas.

* 1. ¿Cuál es el estado actual del laboratorio? ¿Por qué?

Consideramos que cumplimos con lo establecido, de implementar todo porque hubo trabajo en equipo.

* 1. ¿Cuál consideran fue el mayor logro? ¿Por qué?

Implementar los triggers de los casos de uso, a veces era difícil ver que era lo que faltaba para que disparara el mensaje de error.

* 1. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

Los triggers y algunos checks, se resolvió partiendo ese problema en más pequeños hasta encontrar el error que hacía que no funcionara correctamente.

* 1. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?

Nos apoyábamos mutuamente para la elaboración de este laboratorio, para así con esto, lograr el objetivo de terminarlo. Para mejorar los resultados se necesita de más rapidez a la hora de encontrar la solución al problema en particular.

*1* La descripción debe quedar en astah. Modelo Lógico. Modelo Mini. Descripción.