**MODELOS Y BASES DE DATOS**

**Modelo físico**

**2019-02**

**Laboratorio 6/6 [ : ) ]**

# OBJETIVOS

Evaluar el logro de las competencias adquiridas para:

1. Realizar el diseño físico de una base de datos
2. Implementar una base de datos relacional a partir de un diseño físico
3. Definir e implementar índices y vistas para optimizar el funcionamiento
4. Definir e implementar los componentes transaccionales de la base de datos
5. Implementar los mecanismos para empaquetar los servicios de la base de datos
6. Usar el ambiente 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: afterRide.doc afterRide.asta y afterRide.sql

Para organizar el archivo .sql incluyan como comentarios los títulos señalados. No olviden dar nombres significativos a cada uno de los elementos de la implementación.

**CICLO UNO. afterRide.**

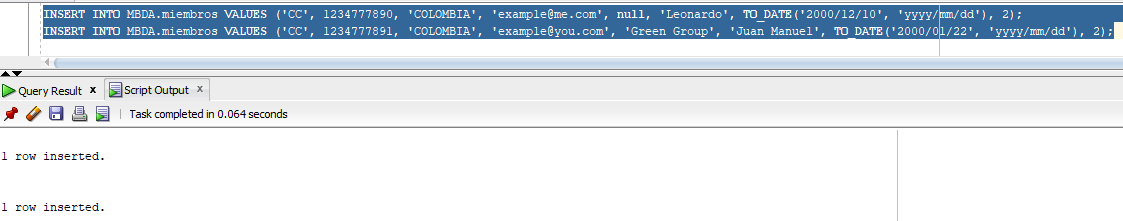
**Extendiendo. Contenidos.**

Los datos de algunos contenidos los vamos a importar de un archivo público llamado mbda.MIEMBROS

1. Consulte la información que actualmente está en el archivo

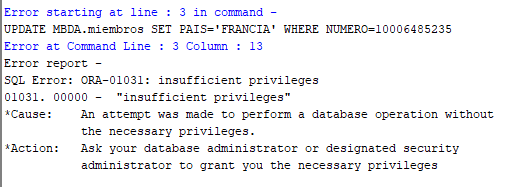
**R/:** Hay una tabla llamada MIEMBRO que tiene los atributos: tipo, número, país, correo, razon, nombres, nacimiento, categoria.

1. Inclúyanse como ciclistas. **HECHO**

**R/: **

1. Traten de modificarse o borrarse. ¿qué pasa?

**R/:**

****

1. Escriban la instrucción necesaria para lograr ese comportamiento. ¿quién la debería escribir?

**R/:** La debería escribir el administrador de la base de datos o cualquier persona con un rol

1. Escriban las instrucciones necesarias para importar los datos de esa tabla a su base de datos.

**R/: Se tienen que generar varios INSERT debido a las restricciones de cada una de las tablas de las que se obtiene los datos**

**MIEMBRO:**

INSERT INTO Miembro

SELECT ROWNUM, tipo, numero, pais, correo FROM MBDA.miembros

WHERE pais IS NOT NULL

AND numero >= 1111111111

AND LENGTH(correo) < 31

AND LENGTH(pais) < 16;

1. Para esta nueva funcionalidad, adicionen este nuevo caso de uso a funciones

COMO Administrador

QUIERO importar miembros de una fuente externa PARA PODER contar con esta información

**Modelo físico. Componentes.**

1. Diseñe e implemente los paquetes correspondiente a los siguientes CRUD del sistema: **Diseño hecho**
   1. Registros (PC\_REGISTROS)
   2. Carreras (PC\_CARRERAS)

En los paquetes deben incluir los subprogramas necesarios para atender los escenarios de los casos de uso de funciones y los casos de uso de consultas. **HECHO**

**CRUDE (Especificación)**

**CRUDI (Implementación)**

**--**

**XCRUD**

**–**

1. Prueben los paquetes construidos con los casos más significativos: 5 éxito y 3 de fracaso. **HECHO**

**CRUDOK**

**CRUDNoOK**

**Modelo físico. Seguridad.**

1. Diseñe e implemente los paquetes que ofrezcan las operaciones válidas para cada uno de los usuarios del sistema. **HECHO**
   1. Administrador (PA\_ADMINISTRADOR)
   2. Participante (PA\_PARTICIPANTE)

**ActoresE**

**ActoresI**

1. Asumiendo que uno de sus compañeros del curso (no del equipo) es el administrador,otorguen los permisos correspondientes. Prueben la ejecución desde esa cuenta.**HECHO**
2. Creen el rol de participantes, otorguen los permisos correspondientes a ese rol.Asumiendo que dos de sus compañeros de curso (no del equipo) son participantes, asígnenles ese rol. Prueben la ejecución desde una de esas cuentas.**HECHO**

**Seguridad (Autorizaciones)**

**–-**

**XSeguridad**

1. Prueben el esquema de seguridad con los casos más significativos: 5 éxito y 3 de fracaso. **HECHO**

**SeguridadOK**

**SeguridadNoOK**

[AYUDA: Consulta la instrucción CREATE ROLE rol]

[AYUDA: Consulte la instrucción GRANT rol TO usuario]

**Pruebas SOLO FALTA ESTO**

Las pruebas de aceptación son historias de varios pasos (10 aprox) que cuentan un uso posible del sistema.

Las siguientes son algunas reglas de construcción:

1. se diseñan con base en los casos de uso de funciones y consultas
2. para cada paso se presenta una descripción en lenguaje natural seguida de la instrucciónSQL correspondiente.
3. las instrucciones SQL son únicamente llamados a métodos de los paquetes de actores.
4. se ilustran acciones de éxito, consultas y la protección sobre acciones no permitidas.5) el éxito de las acciones se confirma con un paso siguiente.

Es necesario que la historia tenga un buen argumento e ilustre las mejores zonas de implementación del sistemas.

**A.** Diseñen e implementen una prueba de aceptación. **Pruebas**

**RETROSPECTIVA**

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

**R/:** 15 horas/hombre.

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

**R/:** Únicamente faltó la parte de pruebas, pues no entendimos bien el concepto y no supimos implementarlo de forma adecuada.

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

**R/:** Poder importar los datos a nuestra base de datos, ya que al inicio nos generó bastantes problemas.

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

**R/:** Crear los paquetes, ya que no sabíamos la sintaxis exacta, un compañero nos ayudo a solucionarlo.

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

**R/:** Usamos la práctica de programación a pares y hablamos menos cuando estábamos desarrollando el laboratorio.