

## Semana 9 - EFT

# Modelamiento de Bases de Datos (PRY2204)

Formato de respuesta

<b>Nombre estudiante:</b>	Bastián Valdivia
<b>Asignatura:</b>	Modelamiento de base de datos
<b>Profesor:</b>	Rodrigo Opazo

# Descripción de la actividad

En esta novena semana, realizarás la Evaluación Final Transversal (EFT), donde a partir de un caso planteado, deberás entregar el desarrollo de una base de datos partiendo con el Modelo Conceptual hasta finalizar con la implementación de un modelo físico para brindar solución a los requerimientos del cliente. El producto de esta evaluación se dividirá en dos partes, y para cumplir con los requisitos mencionados, deberás realizar los siguientes pasos:

En la parte I, tendrás que generar tres archivos y un enlace:

- Un documento Word con las capturas de los modelos desarrollado en todas sus etapas.
- Un archivo zip con los modelos de datos desarrollados en la herramienta Oracle SQL Developer Data Modeler.
- Un script con extensión .SQL, conteniendo la implementación funcional desarrollado en la herramienta Oracle SQL Developer de la creación y poblamiento de las tablas, y la recuperación de datos del caso práctico entregado.
- Un enlace al repositorio de GitHub que contendrá los archivos anteriormente indicados.

En la Parte II, deberás presentar la implementación de tu solución, al caso planteado, mediante la grabación de un video (con una duración máxima de 5 minutos), explicando el desarrollo de tu proceso en cada una de las etapas de diseño de la base de datos, cómo funciona tu implementación, características principales de lo realizado y demostración del script entregado. Debe aparecer el IDE (Oracle SQL Developer Data Modeler y Oracle SQL Developer abierto) y una ventana donde seas visible.

## Instrucciones específicas

Para llevar a cabo la actividad de la semana, considera las siguientes indicaciones:

- Deberás usar la herramienta Oracle SQL Developer Data Modeler para construir el Modelo Entidad Relación (MER).
- Deberás usar la herramienta Oracle SQL Developer Data Modeler para construir el Modelo Relacional Normalizado (MR).
- Deberás utilizar el IDE Oracle SQL Developer y el motor DBMS Base de Datos Oracle 19C Express Edition (o versiones superiores) o utilizar Oracle Cloud para la creación de las tablas del Modelo Relacional (MR).
- Utilizando SQL Developer, conéctate a la base de datos como usuario SYS o SYSTEM y ejecuta el script [PRY2204\\_EFT\\_S9\\_Script\\_crea\\_usuario.sql](#) que crea el usuario PRY2204\_S9. Si estás utilizando Oracle Cloud, realiza este paso como usuario ADMIN.
- Crea una nueva conexión a la base de datos llamada PRY2204\_S9, con el usuario creado en el punto anterior.



### Importante

Los resultados que se proporcionan en cada caso son referenciales para que puedas ver el formato en el cual se requiere la información y corresponden a un ejemplo que el script puede generar en cada caso.

A continuación, te presentaremos el contexto de negocio que deberás analizar en detalle:

## CONTEXTO CASO: Cristalería Andina S.A.

Cristalería Andina S.A. es una empresa chilena dedicada a la fabricación de envases de vidrio (botellas, frascos y vasos) para las industrias de alimentos, bebidas, vinos, cosmética y farmacia. Opera bajo un esquema 24/7 con tres turnos rotativos y mantiene dos plantas productivas en Chile, ubicadas en distintas regiones y comunas, lo que obliga a ordenar su información territorial para efectos operativos y regulatorios.

Cada planta está compuesta por hornos y líneas IS (formado), además de equipos de temple, inspección y empaque. Las máquinas se clasifican por tipo de máquina y cada equipo posee un código único que se usa en trazabilidad, seguridad y mantenimiento. Los equipos pueden estar activos o inactivos según su condición.

La dotación se organiza con empleados que tienen uno de estos tres perfiles clave:

- Jefe de turno: coordina líneas y personas, define la carga de trabajo y autoriza paradas.
- Operario: ejecuta tareas en caliente o frío (moldeado, inspección, empaque) y debe contar con certificaciones específicas.
- Técnico de mantenimiento: atiende mantenciones programadas y correctivas, con especialidad (eléctrica/mecánica/instrumentación) y nivel de certificación.

La operación diaria exige planificar y registrar asignaciones de turno: qué empleado trabaja en qué máquina, en qué turno y en qué fecha, junto con el rol desempeñado (p. ej., Moldeador, Inspector). Paralelamente, se deben programar y cerrar órdenes de mantenimiento, dejando evidencia del técnico responsable, la fecha programada y la fecha efectiva de ejecución.

Hoy la empresa gestiona estos datos en planillas separadas, lo que genera duplicidad, asignaciones superpuestas, códigos de máquina inconsistentes y pérdida de trazabilidad sobre quién operó qué equipo y cuándo. Para resolverlo, se requiere una base de datos relacional normalizada que mejore la planificación, cumplimiento normativo, disponibilidad de equipos y calidad de datos para la toma de decisiones.

## **PARTE I: DESARROLLO E IMPLEMENTACIÓN**

Debido a que el equipo de desarrollo IT anterior, dejó varias actividades incompletas, el proyecto se dividirá en cuatro fases.

- En la **Fase 1** se te entregarán los requisitos de negocio para que puedas construir un Modelo Entidad Relación Extendido (notación Barker).
- En la **Fase 2**, se te solicitará transformar el Modelo Entidad Relación Extendido a un Modelo Relacional Normalizado.
- En la **Fase 3**, deberás construir el script DDL a partir del desarrollo previo.
- Finalmente, en la **Fase 4**, deberás recuperar los datos usando sentencias SELECT.

### **Fase 1: Modelo Entidad Relación Extendido (MER-E) Normalizado**

Diseña el Modelo Entidad Relación Extendido Normalizado basado en los siguientes requerimientos:

- Se debe registrar la infraestructura productiva: plantas con su nombre, dirección y comuna. Cada comuna pertenece a una región.
- Se desea almacenar información de todo el personal de la cristalería, que se clasifica en jefes de turno, operarios y técnicos de mantenimiento. En todos los casos se los identificará con un código numérico, y además se registrarán: RUT, nombres y apellidos, fecha de contratación, sueldo base, estado activo (S/N), planta en la que trabajan, AFP y sistema de salud. Por defecto el estado activo de los empleados es S.
- Para los jefes de turno, además, se debe registrar el área de responsabilidad y el máximo de operarios que pueden coordinar en un turno.
- Para los operarios, se debe registrar la categoría de proceso en la que trabajan (por ejemplo: caliente, frío, inspección), una certificación si aplica, y las horas estándar por turno. Por defecto, las horas de los turnos es de 8 horas.
- Para los técnicos de mantenición, se debe registrar la especialidad (por ejemplo: eléctrica, mecánica, instrumentación), el nivel de certificación (si tuviese) y el tiempo de respuesta estándar.

- Un empleado puede tener asociado un jefe directo dentro de la organización; si corresponde, se debe almacenar esa relación jerárquica.
- Se deben registrar los tipos de máquina que utiliza la cristalería y, para cada máquina, se registra el número de máquina, nombre, estado activo (valores S/N), la planta donde se encuentra y su tipo de máquina. Cabe recalcar que el código interno único de las máquinas está compuesto por el número de máquina y la planta donde se encuentra.
- Se deben definir los turnos de trabajo (por ejemplo: Mañana, Tarde, Noche) indicando su hora de inicio y hora de término. Considera para las horas de inicio y término un tipo de dato CHAR que almacene la hora en formato HH:MM
- Se deben registrar órdenes de mantenimiento de maquinaria indicando la máquina intervenida, el técnico responsable, la fecha programada, la fecha de ejecución (si ya fue realizada) y una descripción del trabajo.
- Se requiere planificar diariamente la dotación mediante asignaciones de turno, donde para cada fecha se indique el empleado, el turno, la máquina en la que trabaja y el rol desempeñado (si aplica).
- Un empleado no puede estar asignado a más de un turno en la misma fecha.

### Figura 1

#### Asignación de Turnos en Cristalería

**Módulo:** Planificación diaria de personal y máquinas

**Usuario principal:** Jefe de Turno

**Fecha:** \_\_\_\_\_

#### Campos a completar:

- **Empleado:** \_\_\_\_\_
- **Rol en Turno:** \_\_\_\_\_
- **Máquina:** \_\_\_\_\_
- **Turno:** ( ) Mañana ( ) Tarde ( ) Noche
- **Fecha:** \_\_\_\_ / \_\_\_\_ / \_\_\_\_

*Nota.* La imagen muestra un formulario para registrar en el sistema qué empleado trabajará en qué máquina y turno en una fecha específica.

## Fase 2: Modelo Relacional (MR) Normalizado

Una vez construido el Modelo Entidad Relación Extendido (MER-E) Normalizado, transforma a Modelo Relacional (MR) Normalizado recordando que:

- El nombre de los tablas, columnas y constraints son representativos.
- Las tablas pueden tener clave primaria simple o compuesta.
- Las tablas pueden tener clave foránea simple o compuesta.

## Fase 3: Creación y poblamiento del esquema

Para la construcción del script de creación de las tablas del Modelo Relacional (MR) Normalizado debes considerar lo siguiente:

- Crea las tablas del modelo desarrollado en la Fase 2 en orden secuencial siguiendo la jerarquía de dependencia, es decir, desde las tablas fuertes a las más débiles. **Utiliza el IDE: Oracle SQL Developer.**
- Crea los valores por defecto y las restricciones de Clave Primaria (PK), Clave Foránea (FK), Clave Única (UN) y Check (CK), de las tablas de acuerdo al diagrama relacional y al análisis del modelo.
- Considera que todas las restricciones (CONSTRAINTS) deben tener un nombre representativo según las tablas en las que pertenecen: PK, FK, CK, UN.
- Asigna los tipos de datos y precisiones de las columnas de las tablas, de acuerdo al modelo desarrollado, para posteriormente no tener problemas con el poblamiento de la base de datos.

Al crear las tablas debes considerar:

- La clave primaria de la tabla COMUNA debe ser de tipo IDENTITY con incremento de 5 en 5 y que parte en 1050.
- Cuando una orden de mantención registra fecha de ejecución, ésta debe ser igual o posterior a la fecha programada.
- Los nombres de región, sistema de salud, AFP, tipo de máquina y turno no se repiten en sus respectivos catálogos.

Al poblar las tablas, ten en cuenta:

- La tabla REGION, el id\_region debe iniciar en 21, y se debe incrementar en 1 (usa objeto secuencia)

Considera que tu script se ejecutará en forma secuencial, es decir, debes ordenarlo según las restricciones de integridad y dependencia. Solo debes poblar las siguientes tablas: REGION, COMUNA, PLANTA, TURNO

## Figura 2

*Datos para considerar en el poblamiento de las tablas*

Tabla REGION

	ID_REGION	NOMBRE_REGION
1	21	Región de Valparaíso
2	22	Región Metropolitana

Tabla COMUNA

	ID_COMUNA	NOMBRE_COMUNA	REGION_ID
1	1050	Quilpué	21
2	1055	Maipú	22

Tabla PLANTA

	ID_PLANTA	NOMBRE_PLANTA	DIRECCION	COMUNA_ID
1	45	Planta Oriente	Camino Industrial 1234	1050
2	46	Planta Costa	Av. Vidrieras 890	1055

Tabla TURNO

	ID_TURNO	NOMBRE_TURNO	HORA_INICIO	HORA_FIN
1	M0715	Mañana	07:00	15:00
2	N2307	Noche	23:00	07:00
3	T1523	Tarde	15:00	23:00

*Nota.* Las imágenes muestran los datos a considerar en el poblamiento de las tablas. Oracle. (s.f.). SQL Developer. (24.3.1). [Software]. <https://www.oracle.com/cl/database/sqldeveloper/>

#### Fase 4: Recuperación de Datos

Posteriormente al poblamiento de los datos, debes recuperar toda la información generando informes con el formato indicado usando la sentencia SELECT de manera adecuada.

##### INFORME 1:

Listar los turnos cuyo horario de inicio sea posterior a “20:00”. Estas filas deben ser ordenadas de manera descendente por la hora de inicio.

##### Figura 3

*Registro de turnos con horario de inicio mayor a 20:00.*

TURNO	ENTRADA	SALIDA
1 N2307 - Noche	23:00	07:00

*Nota. La tabla muestra un listado con TURNO, ENTRADA, SALIDA.*

##### INFORME 2:

Adicional a la información anterior, se necesita obtener los turnos diurnos que tienen horario de inicio entre “06:00” y “14:59” (ambos incluidos en el rango). Se requiere entregar el informe ordenado por hora de inicio de manera ascendente.

##### Figura 4

*Registro de turnos diurnos.*

TURNO	ENTRADA	SALIDA
1 Mañana (M0715)	07:00	15:00

*Nota. La tabla muestra un listado de los turnos diurnos.*

## **PARTE II: PRESENTACIÓN DE IMPLEMENTACIÓN**

Cuando hayas terminado de construir las 4 fases, deberás presentar la implementación de tu solución, mediante la grabación de un video (con una duración máxima de 5 minutos), explicando:

- Desarrollo de tu proceso en cada una de las etapas de diseño de la base de datos.
- Cómo funciona tu implementación.
- Características principales de lo realizado.
- Demostración de ejecución de script desarrollado.

Debe aparecer el IDE (Oracle SQL Developer Data Modeler y Oracle SQL Developer abierto) y una ventana donde seas visible.

- Adjunta el enlace de tu video al momento de subir tu actividad.

Puedes descargar el instructivo de Kaltura en el siguiente enlace:

[https://ava.duoc.cl/bbcswebdav/xid-3577107\\_1](https://ava.duoc.cl/bbcswebdav/xid-3577107_1)

## Instrucciones de entrega

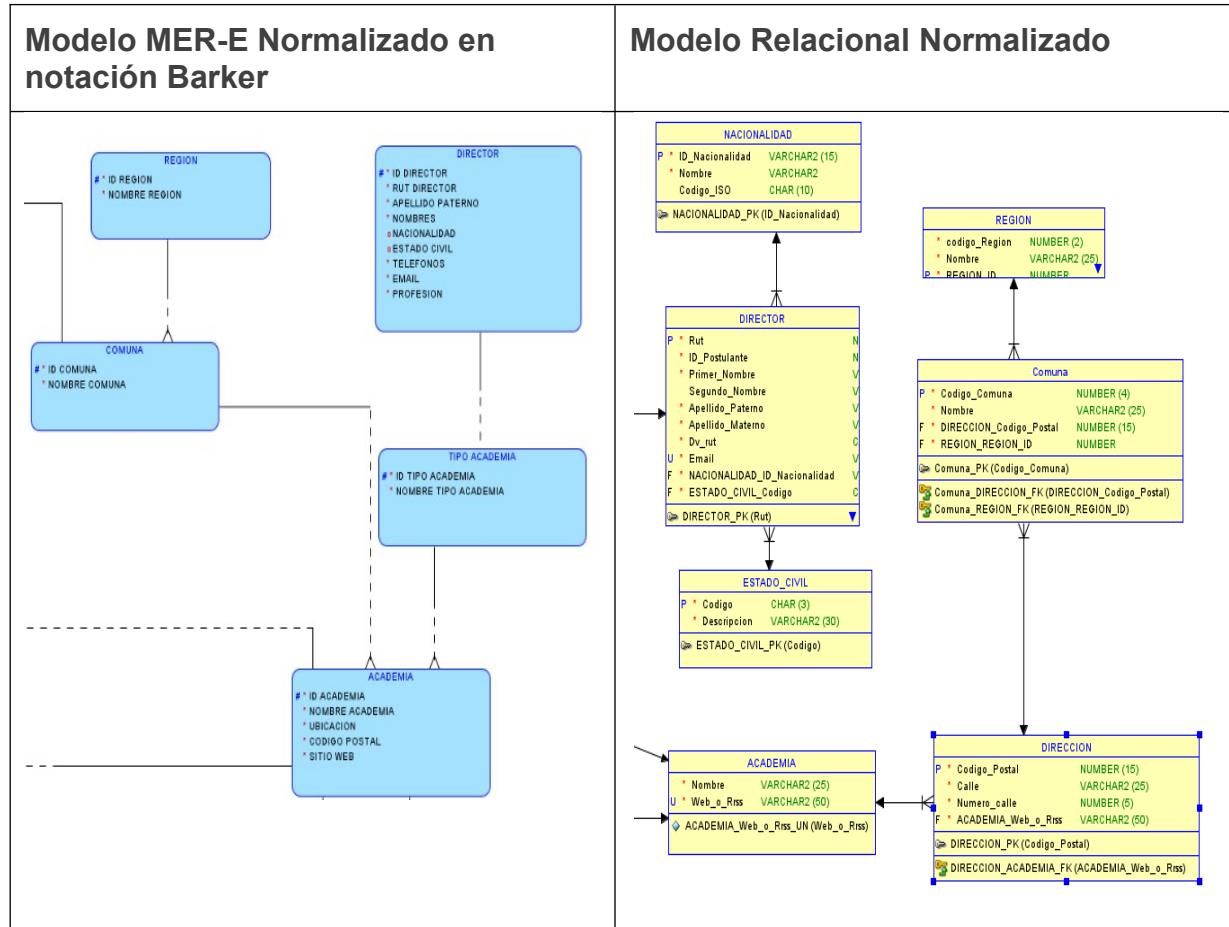
**Paso 1:** Para llevar a cabo este proceso, tendrás que utilizar la herramienta Oracle SQL Data Modeler, disponible de descarga a través del siguiente enlace:

<https://www.oracle.com/database/sqldeveloper/technologies/sql-data-modeler/download/>

Cuando tengas lista tu solución, deberás adjuntar en este documento las capturas de:

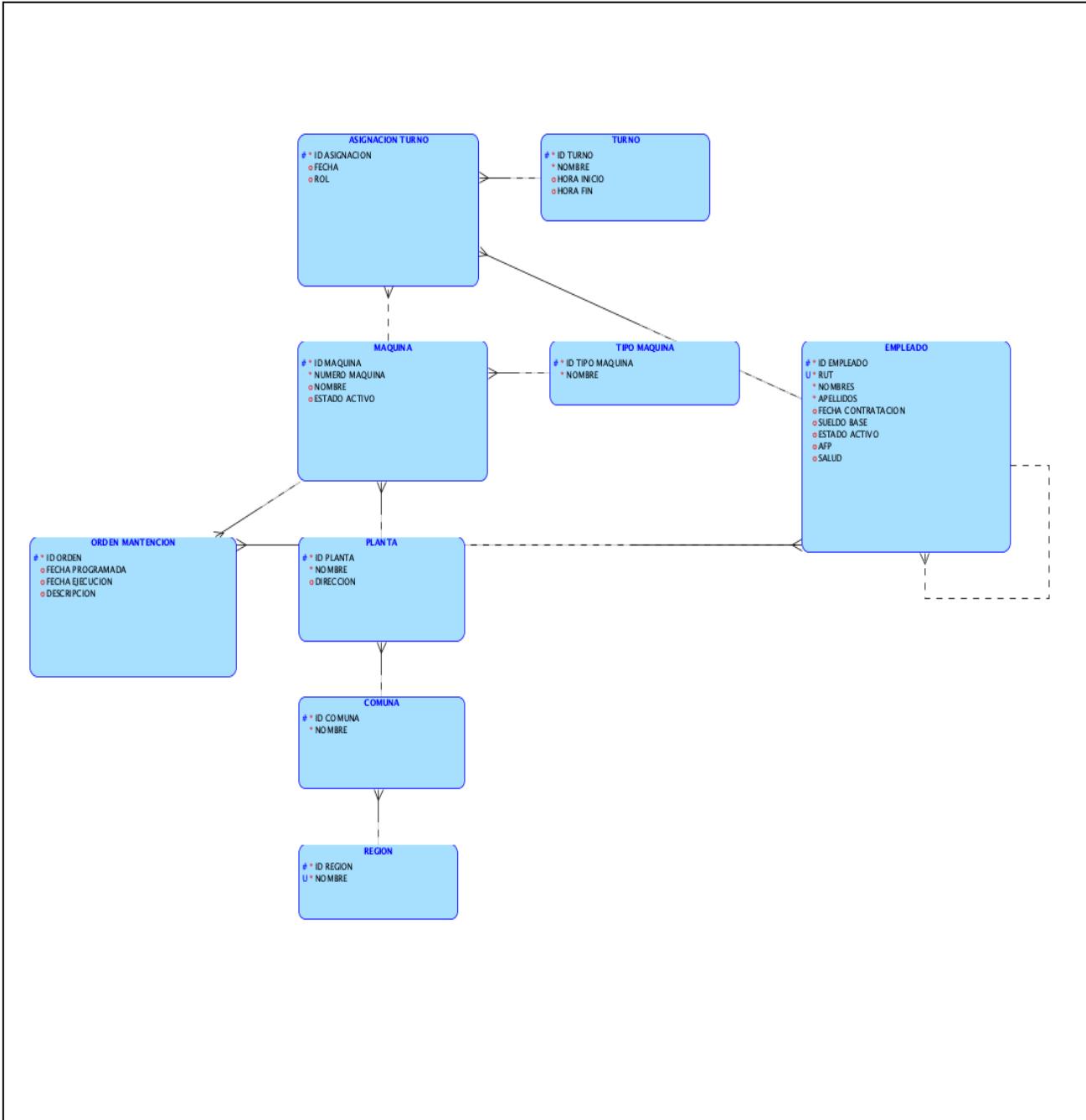
- Modelo Entidad-Relación-Extendido (MER-E) Normalizado solicitado en Fase 1 en notación Barker.
- Modelo Relacional (MR) Normalizado solicitado en Fase 2.

### Ejemplos:

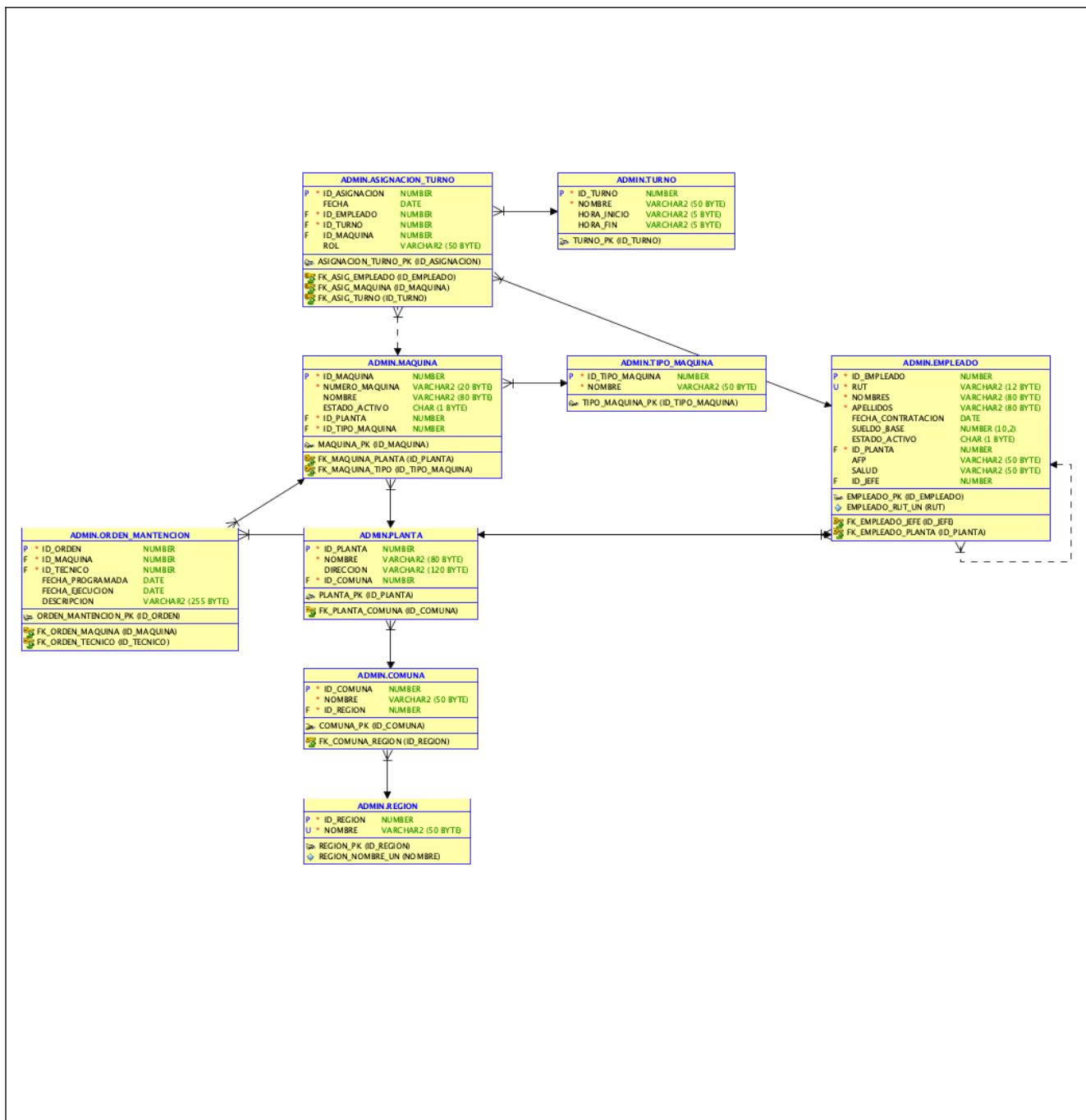


Adjunta tus evidencias en esta parte:

**Modelo Entidad Relación-Extendido Normalizado (MER-E) en notación Barker:**



## Modelo Relacional (MR) Normalizado:



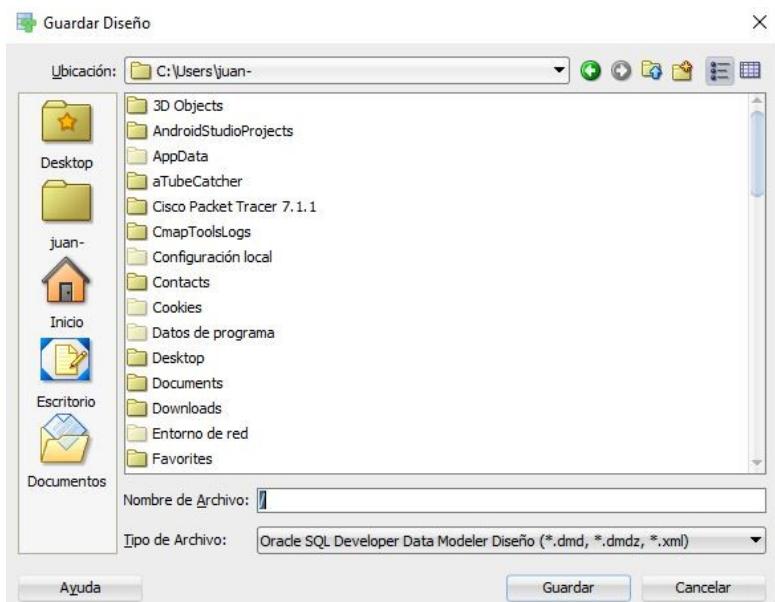
## Paso 2:

Además, tendrás que descargar el resultado ([los modelos MER-E y MR desarrollados](#)) y generar un archivo DMD y una subcarpeta.

Para ello, tendrás que hacer clic en la opción “Guardar como” del menú “Archivo”, lo que desplegará el submenú que se ilustra en la siguiente imagen:

**Figura 5**

*Cómo guardar Diseño en un archivo .dmd*



*Nota.* Ejemplo de diseño guardado como archivo .dmd con sus respectivas subcarpetas.

Oracle. (s.f.). *Oracle SQL Developer Data Modeler* [Software]. Oracle.

<https://www.oracle.com/cl/database/sqldeveloper/technologies/sql-data-modeler/>

**Figura 6**

*Ejemplo de archivos de diseño generado con SQL Developer Data Modeler*

Nombre	Tipo
Modelo_Base	Carpeta de archivos
Modelo_Base.dmd	Archivo DMD

**Figura 7**

*Contenido estándar de la subcarpeta generada del ejemplo (Modelo\_Base)*

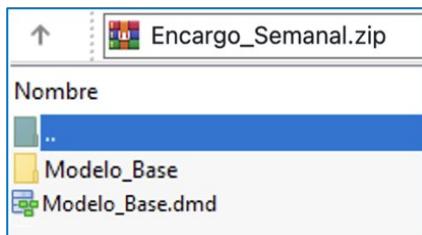
Nombre	Fecha de modificación	Tipo
businessinfo	03/10/2024 15:55	Carpeta de archivos
datatypes	03/10/2024 15:55	Carpeta de archivos
files	03/10/2024 15:55	Carpeta de archivos
logical	03/10/2024 15:55	Carpeta de archivos
mapping	03/10/2024 15:55	Carpeta de archivos
pm	03/10/2024 15:55	Carpeta de archivos
rdbms	03/10/2024 15:55	Carpeta de archivos
rel	03/10/2024 15:55	Carpeta de archivos
ChangeRequests.local	03/10/2024 15:55	Archivo LOCAL
dl_settings.xml	03/10/2024 15:55	xmlfile

*Nota.* La figura muestra el contenido detallado de la carpeta de recursos asociada a un modelo de Oracle Data Modeler. Oracle. (s.f.). *Oracle SQL Developer Data Modeler [Software]*. Oracle. <https://www.oracle.com/cl/database/sqldeveloper/technologies/sql-data-modeler/>

**Paso 3:** Una vez generado el archivo .dmd y su subcarpeta correspondiente, todo este contenido debe comprimirse en un solo archivo ZIP o RAR.

**Figura 8**

*Contenido del archivo comprimido*

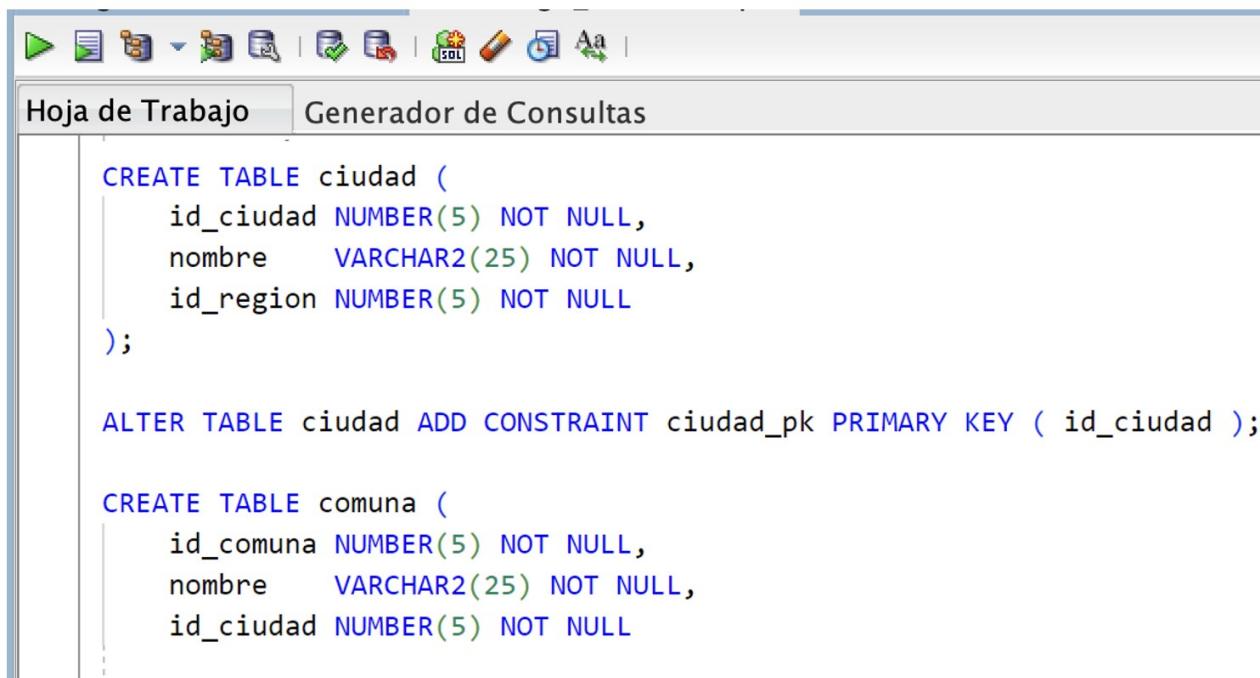


*Nota.* La figura muestra la estructura interna del archivo comprimido Encargo\_Semanal.zip. Oracle. (s.f.). *Oracle SQL Developer Data Modeler [Software]*. Oracle. <https://www.oracle.com/cl/database/sqldeveloper/technologies/sql-data-modeler/>

**Paso 4:** Posteriormente, deberás guardar el Script (de la implementación desarrollada) en un archivo .SQL:

**Figura 9**

*Ejemplo Script (SQL)*



The screenshot shows the Oracle SQL Developer interface. The title bar says "Hoja de Trabajo" and "Generador de Consultas". The main area contains the following SQL code:

```
CREATE TABLE ciudad (
    id_ciudad NUMBER(5) NOT NULL,
    nombre VARCHAR2(25) NOT NULL,
    id_region NUMBER(5) NOT NULL
);

ALTER TABLE ciudad ADD CONSTRAINT ciudad_pk PRIMARY KEY ( id_ciudad );

CREATE TABLE comuna (
    id_comuna NUMBER(5) NOT NULL,
    nombre VARCHAR2(25) NOT NULL,
    id_ciudad NUMBER(5) NOT NULL
```

No

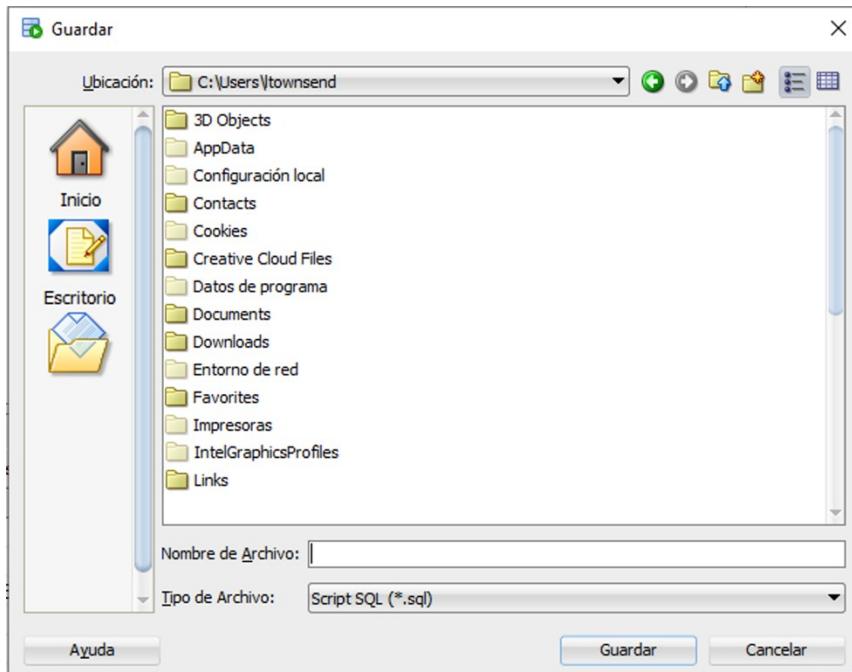
ta. La imagen muestra la definición de dos tablas en SQL: ciudad y comuna. La tabla ciudad contiene los campos id\_ciudad, nombre e id\_region, donde id\_ciudad se define como clave primaria mediante el comando ALTER TABLE ciudad ADD CONSTRAINT ciudad\_pk PRIMARY KEY (id\_ciudad);. Oracle. (s.f.). SQL Developer.

<https://www.oracle.com/cl/database/sqldeveloper/>

Para ello, tendrás que hacer clic en la opción “Guardar como” del menú “Archivo”, lo que desplegará el submenú que se ilustra en la siguiente imagen y seleccionar Tipo de Archivo “Script SQL (\*.sql)”:

**Figura 10**

*Cómo guardar script en un archivo .sql*



*Nota.* Ejemplo de script guardado como archivo .sql. Oracle. (s.f.). *Oracle SQL Developer [Software]*. Oracle. <https://www.oracle.com/database/sqldeveloper/technologies/download/>

**Figura 11**

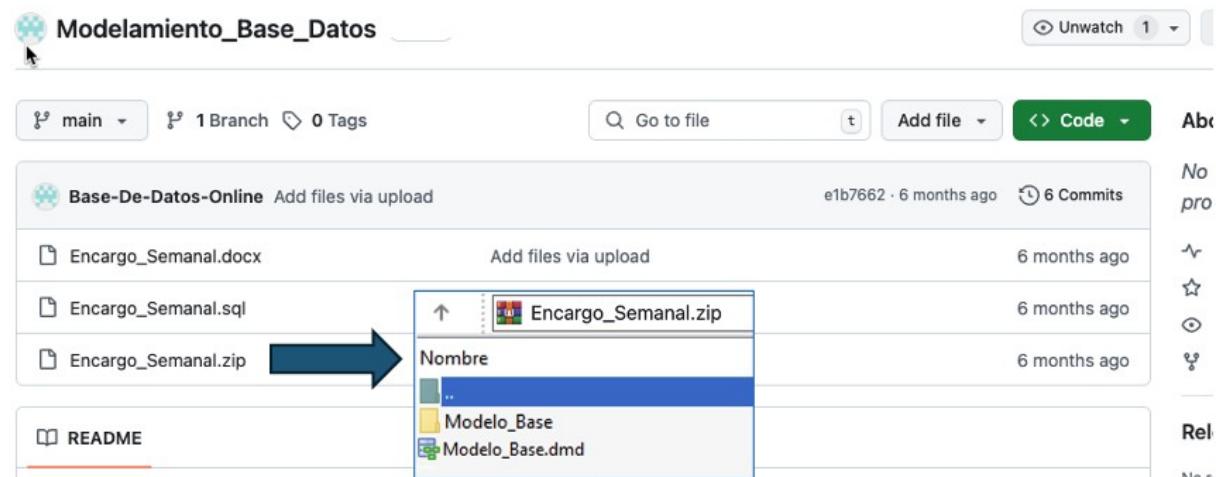
*Ejemplo de script .sql generado con SQL Developer*

Nombre	Tipo	Tamaño
Encargo_Semanal.sql	Archivo SQL	15 KB

**Paso 5:** Una vez generado todos los respaldos (este documento Word, archivo zip con dmd y subcarpeta, archivo SQL) deberás subirlos al repositorio GitHub (sin comprimir). Si no has creado tu cuenta aún, puedes hacerlo a través del siguiente enlace: <https://github.com/>

**Figura 12**

### Repository en GitHub

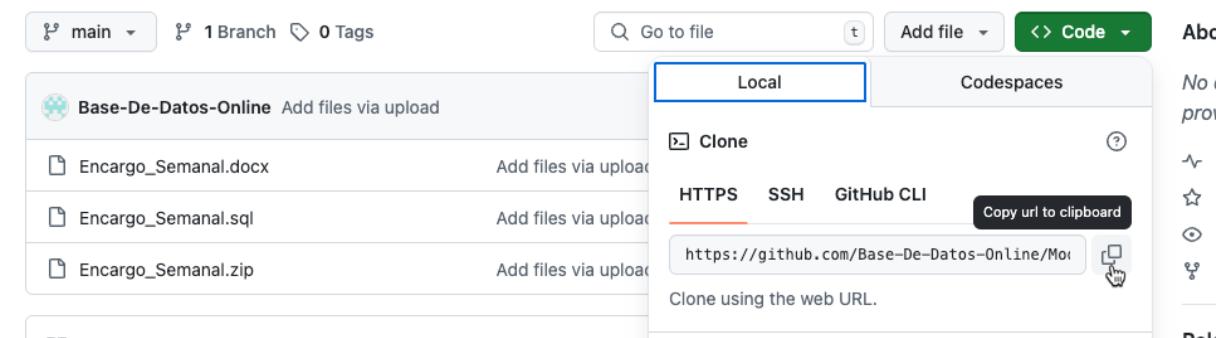


*Nota.* Ejemplo genérico de archivos cargados en el repositorio GitHub. GitHub (s.f.). GitHub. <https://github.com/>

**Paso 6:** Una vez cargados los archivos, desde el repositorio, deberás generar un enlace de tu proyecto:

**Figura 13**

### Enlace de proyecto GitHub



*Nota.* Ejemplo genérico de donde se extrae un enlace en GitHub. GitHub (s.f.). GitHub. <https://github.com/>

**Paso 7:** Finalmente, deberás subir al AVA este documento Word sin comprimir (con las capturas), el archivo .SQL (con la implementación), el enlace de tu repositorio GitHub y el enlace de tu video de la implementación de tu solución en la sección “Entrega”.

#### Figura 14

*Visualización de entrega en el AVA*

Entrega

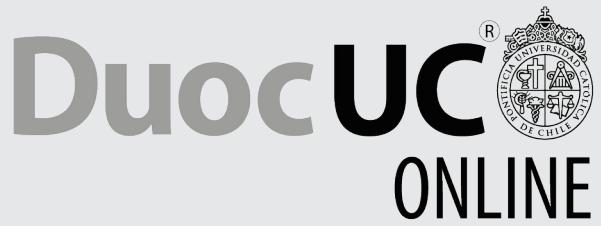
The screenshot shows a digital submission interface. At the top, there are two blue links: "Enlace a repositorio GitHub" and "Enlace a video de presentación". Below these are two file attachments:

- Encargo\_Semanal.docx**: A Microsoft Word document. Its preview window shows a single page with the text "Encargo Semanal". The toolbar below it includes icons for zoom, search, and print.
- Encargo\_Semanal.sql**: An SQL script. Its preview window shows the following code:

```
CREATE TABLE banco (
    cod_banco NUMBER(2) NOT NULL,
    nombre     VARCHAR2(25) NOT NULL
);

ALTER TABLE banco ADD CONSTRAINT banco_pk PRIMARY KEY ( cod_banco );

CREATE TABLE boleta (
```



Reservados todos los derechos Fundación Instituto Profesional Duoc UC. No se permite copiar, reproducir, reeditar, descargar, publicar, emitir, difundir, de forma total o parcial la presente obra, ni su incorporación a un sistema informático, ni su transmisión en cualquier forma o por cualquier medio (electrónico, mecánico, fotocopia, grabación u otros) sin autorización previa y por escrito de Fundación Instituto Profesional Duoc UC. La infracción de dichos derechos puede constituir un delito contra la propiedad intelectual.