

Índice

Objetivo de la práctica	2
Modelo de datos	2
Ejercicio 1 : Carga del modelo y creación de usuarios (2 puntos)	2
Ejercicio 2 : COMERCIAL y RRHH (2 puntos)	3
Ejercicio 3 : COMERCIAL y COMERCIAL (2 puntos)	3
Ejercicio 4 : CONTABILIDAD y CONTABILIDAD (2 puntos)	4
Instrucciones de entrega	5

Objetivo de la práctica

Tras la práctica se espera que el alumno se familiarice con:

- La creación de distintos usuarios de base de datos
- La asignación de diferentes privilegios a los usuarios, de acuerdo a sus necesidades
- Las posibilidades de las bases de datos SQL para implementar el nivel 3 del estándar ANSI/SPARC

Se puede descargar [la última versión de la práctica en este enlace](#)

Modelo de datos

Se dispone de un [esquema de base de datos descargable en este link](#), que también se incluye como [adjunto](#) a este PDF. El diagrama del esquema se puede consultar en la figura 1.

Habrán varios usuarios manejando la base de datos:

- INFORMATICA: Permisos completos sobre todas las tablas
- RRHH: Permisos completos sobre los datos de las tablas Oficinas y Empleados. Puede ver los datos de Clientes y cambiar el campo Clientes.CodigoEmpleadoRepVentas.
- COMERCIAL: Permisos completos sobre los datos de las tablas Clientes, Pedidos y DetallePedidos. Podrá ver las tablas Productos, Empleados y Oficinas. Podrá modificar datos de la columna Productos.CantidadEnStock.
- CONTABILIDAD: Permisos completos sobre los datos de la tabla Pagos. Pueden leer de las tablas Clientes, Pedidos y DetallePedidos.
- JARDINERIA: Permisos completos sobre los datos de las tablas Productos y GamasProductos

Ejercicio 1 : Carga del modelo y creación de usuarios (2 puntos)

Crea los usuarios necesarios (con contraseña igual al nombre), y después crea la base de datos en el usuario INFORMATICA.

Posteriormente, asigna los permisos necesarios al resto de usuarios, con la orden GRANT. Se recomienda otorgar el privilegio CREATE VIEW a todos los usuarios, para facilitar el resto de ejercicios.

Los usuarios RRHH, COMERCIAL, CONTABILIDAD y JARDINERIA realizarán sus consultas como si las tablas fueran *suyas*. Para ello, utiliza la sentencia [CREATE SYNONYM](#), sin PUBLIC.

¿Qué se entrega?

Un fichero de nombre **1.permisos.sql** con las órdenes necesarias para asignar los privilegios y crear sinónimos. No es necesario incluir la carga de la base de datos ni la creación de usuarios. Para indicar el usuario con el que realizas las órdenes, incluye un comentario y la orden **CONNECT** de **sqlplus**.

```
-- A partir de aquí soy el usuario JARDINERIA
connect JARDINERIA/JARDINERIA;
```

Listado 1: Cómo indicar qué usuario ejecuta las órdenes

Ejercicio 2 : COMERCIAL y RRHH (2 puntos)

- | | |
|---|---|
| <ol style="list-style-type: none">1. COMERCIAL da de alta al nuevo cliente <i>Pepe-gardens</i>, en Madrid (inventa el resto de datos), sin representante.2. Hace una consulta encontrar a los empleados de mayor rango en Madrid (que no tienen un jefe en Madrid).3. Después asigna el primero de esos empleados como representante de <i>PepeGardens</i>. | <ol style="list-style-type: none">1. Mientras tanto, RRHH ha creado a un nuevo empleado (<i>Manolo Bombo</i>), que será el nuevo jefe en Madrid.2. RRHH desplaza al empleado de mayor rango de Madrid a Barcelona, dejando a <i>Manolo Bombo</i> en su lugar:<ul style="list-style-type: none">■ Asigna todos sus representados a <i>Manolo Bombo</i>■ <i>Manolo Bombo</i> que dependerá del jefe que dependía antes el jefe de Madrid■ Los empleados que dependían del jefe de Madrid dependerán de Manolo Bombo. |
|---|---|

Asegúrate de que no se asigna *PepeGardens* a un empleado de Barcelona, sino que será asignado a *Manolo Bombo*.

¿Qué se entrega?

- Un fichero de nombre **2.comercial.sql** con las órdenes de COMERCIAL.
- Un fichero **2.rrhh.sql** con las órdenes de RRHH.

Para indicar el usuario con el que realizas las órdenes, incluye un comentario y la orden **CONNECT** de **sqlplus**.

No se permite utilizar resultados de **SELECT** anteriores. Si se necesita un dato para una orden, debe extraerse de una **subconsulta**. Para facilitar las subconsultas, pueden crearse vistas libremente.

Ejercicio 3 : COMERCIAL y COMERCIAL (2 puntos)

El usuario **INFORMATICA** modifica la tabla de **PRODUCTOS** para que el stock nunca baje de cero, añadiendo una restricción **CHECK**.

1. *PepeGardens* desea comprar 50 ajedreas.
2. COMERCIAL realiza una consulta para ver cuántas hay disponibles. Si no hay suficientes cancela la transacción
3. Tras ello, crea un pedido para *PepeGardens* con esas 50 ajedreas, eliminando 50 ajedreas del stock.

1. Mientras, otro COMERCIAL va a realizar las mismas operaciones con 140 ajedreas para *Agrojardín*.

En el almacén no hay suficientes ajedreas para los dos pedidos. Asegúrate de que sólo uno de los dos COMERCIAL consigue realizar la venta, y que ninguno le ofrece al cliente algo que después no tiene.

¿Qué se entrega?

- Un fichero de nombre **3.comercial50.sql** con las sentencias del usuario COMERCIAL que vende a *PepeGardens*.
- Un fichero **3.comercial140.sql** con las sentencias del usuario COMERCIAL que vende a *Agrojardín*. Incluye un comentario con el error de **Oracle** al intentar vender más ajedreas de las disponibles.
- Un fichero **3.informatica.sql** con la modificación de la tabla *Productos*.

Ejercicio 4 : CONTABILIDAD y CONTABILIDAD (2 puntos)

PepeGardens realiza finalmente el pedido de 50 ajedreas, y otro de 10 palas, a través de *Manolo Bombo*.

1. Un empleado de *PepeGardens* se conecta por Internet para pagarlo con PayPal.
2. Para recibir un pago, la web consulta primero el saldo del cliente (suma de todos sus pedidos menos suma de todos sus pagos).
3. Solo acepta pagos inferiores al saldo deudor del cliente.

1. Otro empleado se acerca para pagarlo en efectivo.
2. Un empleado de CONTABILIDAD realiza las mismas operaciones que la página web.

Asegúrate de que sólo un empleado de *PepeGardens* consigue pagar.

¿Qué se entrega?

- Un fichero de nombre **4.comercial.sql** con las sentencias del usuario COMERCIAL que creó el pedido de las 10 palas.
- Un fichero de nombre **4.paypal.sql** con las sentencias del usuario CONTABILIDAD que lanza la web para cobrar por PayPal.
- Un fichero de nombre **4.efectivo.sql** con las sentencias del usuario CONTABILIDAD que cobra en efectivo.

Para indicar que la transacción se interrumpe por el valor que devuelva una consulta, incluye un comentario como el siguiente:

```
-- Si el campo XXXX de la consulta anterior es  
-- mayor/menor/igual/distinto que XXXX, hago un ROLLBACK.  
-- En otro caso, continúa la transacción.
```

Listado 2: Cómo indicar fin de la transacción por una condición

Instrucciones de entrega

- El ejercicio se realizará y entregará de manera individual.
 - Solo se admiten trabajos en pareja, si en clase es necesario compartir ordenador.
- Entrega tu trabajo en un fichero ZIP, con todos los ficheros indicados en los enunciados
- Los nombres de los ficheros incluyen un número (para que estén ordenados). No incluyas mayúsculas. No incluyas acentos. No pongas espacios. No añadas palabras, letras ni números a los nombres.
- Sube el documento a [la tarea correspondiente en el aula virtual](#)
- Presta atención al plazo de entrega (con fecha y hora).

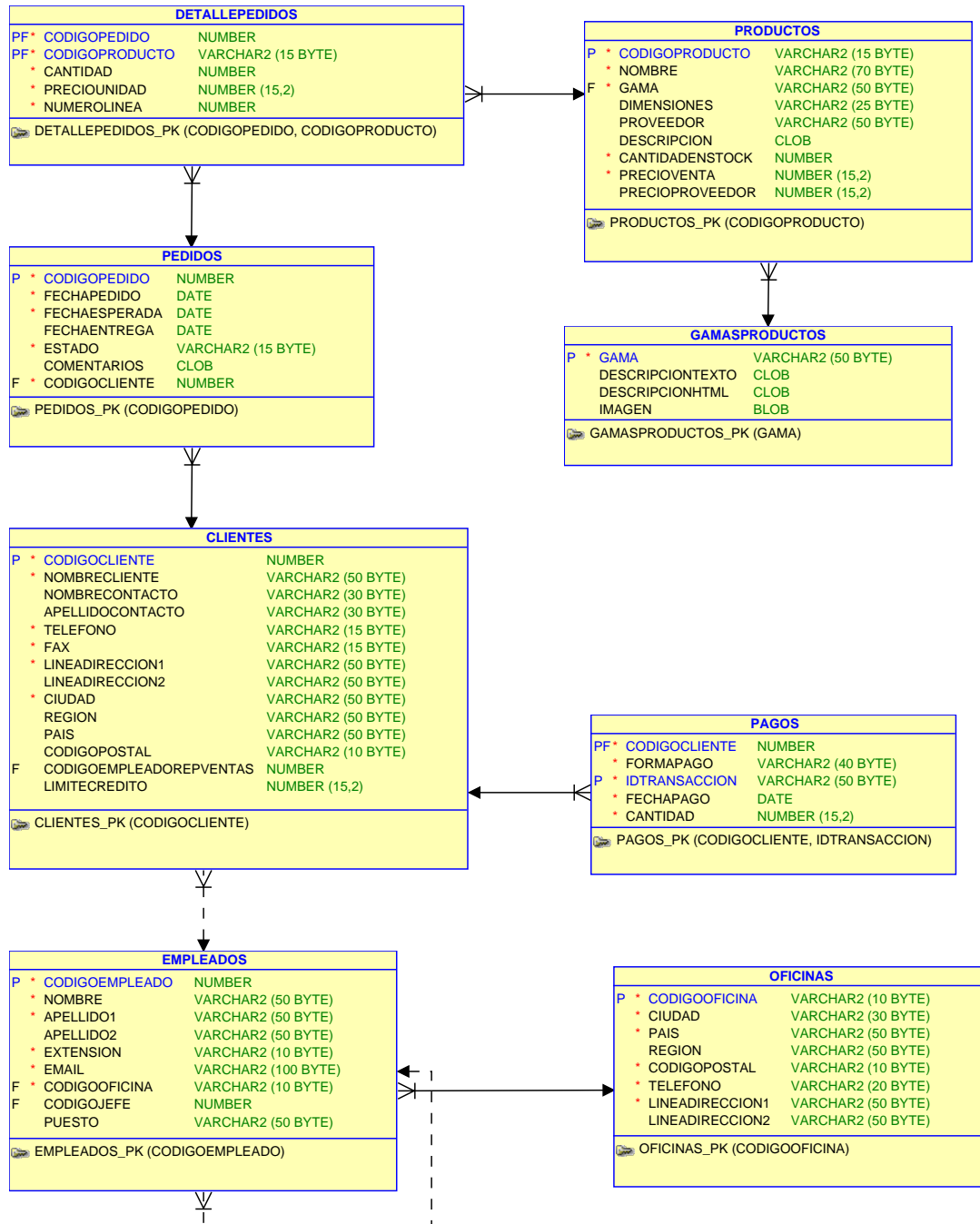


Figura 1: Diagrama de la base de datos