

PRACTICA COMPLEMENTARIA 10
OPERADORES SQL, JOINS

El reporte se entrega en equipos formado por máximo 2 integrantes.

1.1. OBJETIVO:

Entender y reafirmar los conceptos aprendidos en clase, asociados con el diseño de sentencias SQL tipo SELECT aplicadas en la explotación de datos para un caso de estudio, iniciando desde el proceso de diseño, implementación, hasta la recuperación de los datos. El enfoque de esta práctica comprende 2 principales conceptos: Operadores SQL y uso de los distintos tipos de Joins.

1.2. DEFINICIÓN DEL CASO DE ESTUDIO.

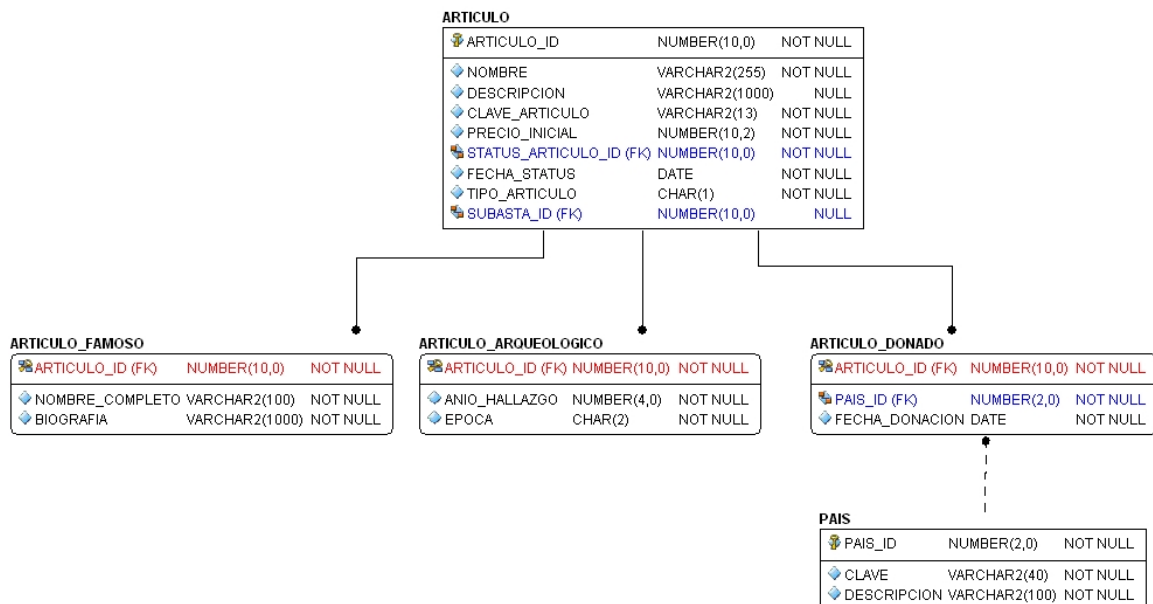
Subasta Mexicana S.A. (SUBMEX) es una empresa que se dedica a realizar subastas a nivel nacional de artículos de interés social. La empresa decide implementar una base de datos relacional para administrar la información de sus subastas, clientes y artículos. Los requerimientos se muestran a continuación:

- SUBMEX cuenta con un catálogo de artículos que pueden participar en un proceso de subasta. Los artículos se dividen en 3 grupos:
 - Artículos pertenecientes a personalidades o famosos.
 - Artículos arqueológicos encontrados por antropólogos
 - Artículos donados por países asociados.
- Cada vez que llega un nuevo artículo, este se debe registrar en la base de datos:
 - Nombre del artículo, descripción, número único de identificación, formado por 13 caracteres, precio inicial de venta.
 - Para los artículos pertenecientes a personalidades, se guarda el nombre completo del famoso al que le perteneció, y una pequeña biografía del famoso.
 - Para los artículos arqueológicos, se guarda el año en el que se realizó el hallazgo indicando si se encontró DC (Después de Cristo) o AC (Antes de Cristo).
 - Para el caso de los artículos donados por los países asociados, se almacena el país que lo donó, y la fecha de donación.
- Los artículos cuentan con un ciclo de vida:
 - REGISTRADO, cuando el artículo se registra en la base de datos.
 - EN SUBASTA, cuando el artículo está en proceso de subasta, la cual puede durar varios días.
 - VENDIDO, cuando el artículo es vendido a un cliente, este ya no se presenta para ser subastado.
 - ENTREGADO. Cuando el artículo es entregado al cliente.
- Se requiere almacenar este status, la fecha de status y manejar su histórico. Se solicita que el histórico se actualice de forma automática al realizar un cambio en el status actual del producto.
- Cuando se decide realizar una nueva subasta, se deberá registrar la siguiente información en la base de datos:
 - Nombre, fecha en la que se inicia la subasta, fecha fin, lugar (campo abierto), y la lista de artículos que participarán en dicha subasta. Los artículos se seleccionan del catálogo, solo aquellos con status = REGISTRADO. En el momento que se registra una subasta con sus artículos, su status cambia a EN SUBASTA.
 - Para realizar una subasta, SUBMEX envía correos electrónicos a sus clientes. La empresa cuenta con un catálogo de clientes con los siguientes datos: nombre, apellido paterno, apellido materno, fecha de nacimiento, email, ocupación, domicilio (ciudad y entidad o estado).
 - Cabe mencionar que un cliente puede comprar varios artículos durante la subasta, y el artículo solo lo compra un cliente.
 - No siempre se les invita a todos ya que cada subasta cuenta con un número limitado de invitados dependiendo el lugar.
 - Cuando el artículo es vendido durante la subasta, el status del artículo se cambia a VENDIDO, y se registran los siguientes datos:
 - Precio de venta, indicando el cliente al que se le vendió el artículo, el cual corresponde con el precio más alto ofertado.
 - Para realizar el cobro de los artículos vendidos SUBMEX registra los datos de las tarjetas de crédito de sus clientes: Número de tarjeta, mes y año de expiración, así como su tipo (VISA, MASTER CARD y AMERICAN EXPRESS). Un cliente puede registrar varias tarjetas de crédito.
 - Finalmente, el sistema deberá generar facturas para el cliente. Se registra la fecha de elaboración, y se asocia a la factura la tarjeta con la cual se realizará el cargo, así como los artículos que integran a dicha factura.

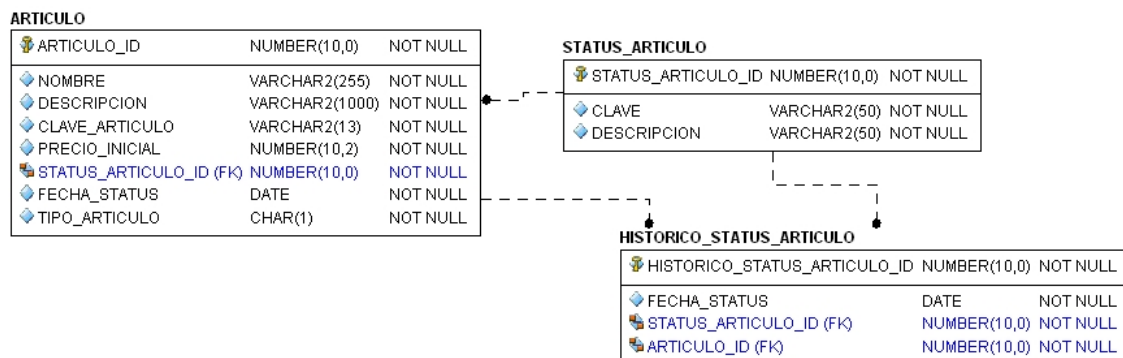
1.3. MODELADO DEL CASO DE ESTUDIO.

Los siguientes puntos muestran los pasos a seguir para construir el modelo relacional.

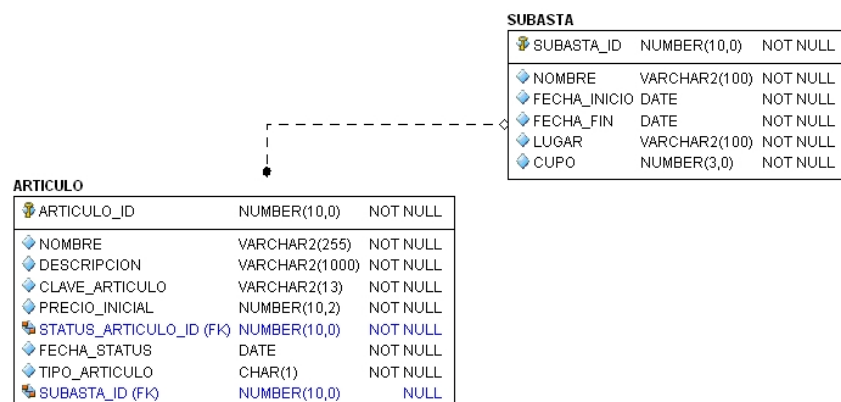
- Modelado de los artículos: Observar en el diagrama la estrategia para modelar los artículos. Se emplea el concepto de subtipos (3 tipos), en este caso son subtipos disjoint y completos, es decir, un artículo solo puede tener uno y solo un subtipo asociado. En la siguiente figura se muestra el modelo relacional. Observar el discriminante TIPO_ARTICULO



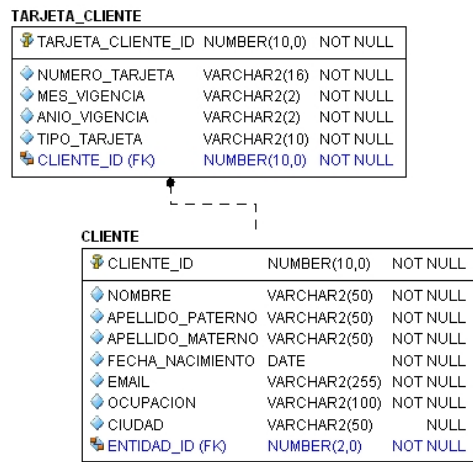
- El siguiente punto a mostrar es el diseño del catálogo de status y el histórico, se crea una tabla para representar el catálogo. Cuando cambie el status del artículo se realiza un UPDATE en la tabla ARTICULO, y posteriormente un INSERT en el histórico.



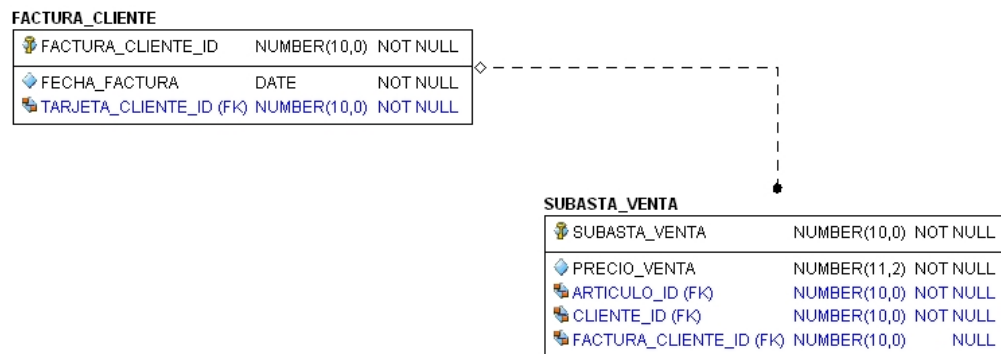
- Posteriormente, se muestra el modelo para guardar los datos de una subasta. Observar que en la tabla ARTICULO se agrega un FK opcional con SUBASTA. Una subasta contiene varios artículos y un artículo pertenece máximo a una subasta. Mientras no sea seleccionado el artículo para ser subastado, el valor del campo SUBASTA_ID será nulo.



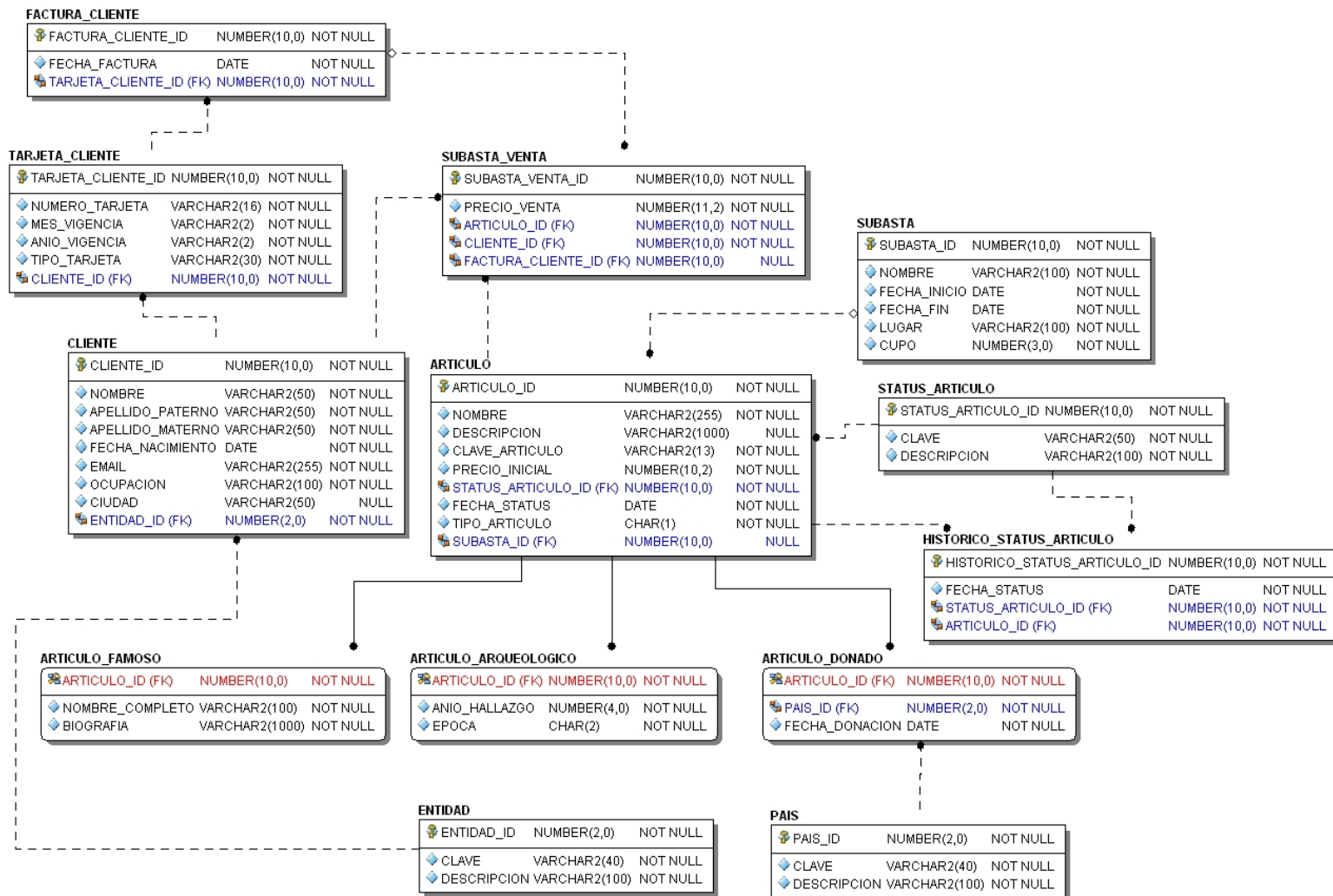
- El siguiente paso es el diseño del cliente y de sus tarjetas de crédito, relación 1:M



- Finalmente, se muestra el diseño de la factura y de sus artículos. Observar que en la tabla SUBASTA_VENTA se guarda el precio de venta final del artículo, el cliente que lo adquirió y la factura en la que se cobra. Observar que este último campo es nulo, normalmente la factura se genera hasta que el cliente termine de comprar y solicite su factura.



- El diseño completo se muestra en la siguiente figura.



1.4. LENGUAJE DE CONSULTA DE DATOS.

1.4.1. Actividad 1- configuración del ambiente.

Crear un script llamado `s-01-main.sql`. El script deberá realizar las siguientes actividades:

1. Crear un usuario llamado `<iniciales>_p1001_subastas` Donde `<iniciales>` es una cadena de 4 caracteres. Los primeros 2 corresponden a los 2 primeros caracteres del apellido paterno del integrante 1 del equipo, y los 2 restantes, al apellido paterno del segundo integrante. En caso de ser individual, agregar los 4 primeros caracteres.
2. Asignarle los privilegios necesarios para poder crear los objetos del modelo relacional. Adicionalmente, agregar el privilegio `create procedure`.
3. Invocar a un nuevo script llamado `s-02-ddl.sql`. El script contendrá la definición de todos los objetos. No es necesario crear el código de forma manual, se puede hacer uso del archivo `subastas.dml` (incluido en la carpeta compartida) y generar el código SQL a través de ER-Studio empleando la vista física llamada `VISTA-ORACLE`.
4. Invocar la ejecución del script `s-03-carga-inicial.sql` que se encuentra en la carpeta compartida de la práctica. Este script contiene los datos del modelo relacional.
5. Ejecutar la instrucción `commit` para confirmar la carga de datos.

1.4.2. Actividad 2 -Creación de sentencias.

Para cada una de las siguientes consultas proporcionar una sentencia SQL que recupere los datos seleccionados. Seguir los siguientes lineamientos:

- El resultado de las sentencias deberá ser almacenado en una tabla llamada `consulta_<n>` donde `<n>` es el número de ejercicio. Para que el resultado de la consulta se guarde en dicha tabla, se puede emplear la instrucción `create table` seguida de la instrucción `select`.

Ejemplo:

- Suponer que se desea mostrar el nombre y la fecha de inicio de todas las subastas que iniciaron en diciembre del 2009. El resultado de la consulta se guardará en la tabla `consulta_1`:

```
create table consulta_1 as
select s.nombre, fecha_inicio
from subasta s
where to_char(fecha_inicio, 'yyyy-mm')='2009-12';
```

- Ejecutar la siguiente sentencia para mostrar los resultados:

```
select * from consulta_1;
```

NOMBRE	FECHA_INICIO
GALA-MAZATLAN	2009-12-03 00:00:00
SALUD-SAN CRISTOBAL DE LA...	2009-12-10 00:00:00
PROMO-MORELIA	2009-12-17 00:00:00
GALA-CIUDAD DE MEXICO	2009-12-24 00:00:00
ALTRUISTA-GUADALAJARA	2009-12-31 00:00:00

- Generar un script llamado s-04-respuestas.sql El script deberá contener todas las sentencias `create table as select` para cada una de las siguientes consultas.
- Se recomienda primero ejecutar la sentencia `select` para verificar resultados. Una vez que se tenga el resultado correcto, ejecutar nuevamente la sentencia pero ahora acompañada de la sentencia `create`.
- En caso de requerir, ejecutar la instrucción `drop table consulta_<n>;` para eliminar las tablas generadas.

Considerar los siguientes puntos generales con respecto al código SQL:

- No emplear NATURAL JOIN, USING a menos que se indique de manera explícita.
 - En caso de no especificarse una sintaxis en particular, se puede emplear cualquiera de las vistas en clase (sintaxis anterior o estándar).
1. Genere un reporte que muestre nombre del artículo, clave, clave del status de todos los artículos donados por el artista 'William Harvey'. Emplear sintaxis estándar.
R: Se deben obtener 3 registros.
 2. Generar un reporte que muestre el id, nombre y clave de todos los artículos que hayan sido entregados al cliente (status= ENTREGADO, no emplear el id del status) en cualquier subasta y que hayan sido donados por 'BELGICA'. Emplear natural join.
R: Se deben obtener 2 registros.
 3. Empleando natural join, generar un reporte que muestre el id, nombre, precio inicial, precio de venta y tipo de artículo, así como el nombre y fecha inicio de la subasta de todos los artículos que compró el cliente MARICELA PAEZ MARTINEZ durante las subastas realizadas en el 2010. Nota: Un artículo se considera comprado cuando se registra el cliente que lo adquirió y se registra el precio de venta. Para el caso de la fecha de inicio emplear el formato como se indica en los siguientes ejemplos. Emplear como nombre de columna `fecha_inicio`

2019/10/24 10:16:44 PM Para fechas después de las 12 PM

2019/10/25 10:19:03 AM Para fechas antes de las 12 PM

R: Se deben obtener el artículo con id 167

4. Generar un reporte que muestre los datos de los clientes (id, nombre y apellidos) y los datos de sus tarjetas (número de tarjeta, tipo, año de vigencia, mes de vigencia) que ya hayan expirado. Notas:
 - a. para validar si ya la tarjeta expiró emplear el mes y año de vigencia.
 - b. Considerar que la consulta se ejecuta en noviembre del 2011
 - c. Si una tarjeta tiene como año y mes de expiración los valores 11/11, la tarjeta aún se considera como vigente.*R: Se deben obtener 16 registros.*
5. Generar un reporte que muestre: identificador del artículo, nombre, clave, tipo, año de hallazgo precio inicial y precio venta (para los que ya fueron vendidos o entregados) de todos los artículos que sean de tipo arqueológico, y que tengan un precio inicial mayor a \$800,000. Emplear notación SQL estándar.
R: Se deben obtener 10 registros de los cuales 3 ya fueron vendidos.
6. Generar un reporte que muestre nombre, apellidos, email, de todos los clientes cuya ocupación sea ABOGADO, y en caso de tener registrado una o más tarjetas de crédito, incluir el tipo de tarjeta. Emplear notación SQL anterior compatible con Oracle.
R: Se deben obtener 5 clientes, uno de ellos cuenta con 2 tarjetas.
7. Suponga que se desea retirar del catálogo a todos los artículos que tengan un precio inicial de más de 900,000, siempre y cuando el artículo todavía no inicie el proceso de subasta, es decir, el artículo no debe tener status EN SUBASTA, ENTREGADO O VENDIDO. Empleando operadores del álgebra

relacional (operadores SET: union, intersection, minus), determine el id, nombre, clave, precio inicial y e identificador del status de los artículos que se deben retirar.

R: Se deben obtener 6 artículos, verificar su precio.

8. SUBMEX ha decidido incrementar en un 10% el precio inicial de todos aquellos artículos arqueológicos que tengan más de 150 años de antigüedad y que aún no se han incluido en un proceso de subasta, es decir, solo se han registrado en la BD. Determine id, clave, nombre, id de status, año de hallazgo, y antigüedad en años de dichos artículos.

R: Se deben obtener al menos 7 artículos.

9. Suponga que un cliente decide realizar una consulta en el catálogo de artículos. El cliente está interesado por todos aquellos artículos cuyo nombre o descripción hagan referencia o hablen de la palabra "Colonial" que no han sido aún vendidos, y que el artículo este en proceso de ser subastado. Generar la sentencia SQL que muestre el nombre y tipo de todos los artículos que cumplan con los criterios de búsqueda.

R: Se debe obtener 1 registro.

10. Suponga que se desea generar un reporte a detalle de todos los artículos que fueron comprados y pagados con la tarjeta de crédito 5681375824866375. Los datos que el reporte debe mostrar son los siguientes:

- Fecha de la factura en formato dd/mm/yyyy, emplear 'fecha_factura' como nombre de columna.
- Numero de la tarjeta
- Nombre y apellidos del cliente
- Precio de venta de cada articulo
- Precio de compra de cada articulo
- Diferencia entre el precio de compra y el de venta
- Nombre y clave del artículo
- Tipo de artículo
- Nombre completo del famoso al que perteneció el artículo (en caso de ser artículo perteneciente a un famoso)
- Año de hallazgo, en caso de que el artículo sea arqueológico.
- Clave del país, en caso de que el artículo haya sido donado por dicho país.

Emplear Notación estándar

R: Se deben obtener 4 registros.

11. Reescribir la consulta anterior pero ahora empleando notación anterior compatible con Oracle.

R: Se deben obtener 4 registros.

1.5. TABLA DE RESPUESTAS.

C1. Incluir en el reporte la siguiente tabla con las sentencias `select`. No incluir los enunciados ni los resultados. Dar formato a las consultas. Tomar como base los ejemplos mostrados anteriormente. Tip: Los ejemplos de esta práctica emplean Fuente Courier New o similar a 8 puntos.

Consulta 1	Consulta 2
Consulta 3	Consulta 4
Consulta 5	Consulta 6
Consulta 7	Consulta 8
Consulta 9	Consulta 10
Consulta 11	

1.6. VALIDACIÓN DE RESULTADOS.

En esta actividad se realizará la validación de las respuestas del ejercicio anterior. Para ello, realizar las siguientes acciones:

- En la carpeta compartida de la práctica obtener todos los scripts `sql/plb`.
- Editar el script `s-05-validador-main.sql` con los valores correspondientes
- En una nueva terminal cambiarse al directorio donde se encuentran los scripts, y ejecutar el script editado (no se requiere emplear al usuario Oracle).

```
sqlplus /nolog  
start s-05-validador-main.sql
```

- En caso de existir errores, revisar y leer cuidadosamente los mensajes de error, corregir y reintentar. **C2. Incluir en el reporte** la captura de pantalla con la salida del validador.

1.7. CONTENIDO DEL REPORTE.

Para realizar la evaluación de la práctica se deberá anexar en la última página del reporte la rúbrica correspondiente:

- Rúbrica para el grupo de laboratorio.
- Rúbrica para el grupo de teoría plan 2010
- Rúbrica para grupo de teoría plan 2016 inscritos en otro grupo de laboratorio

Imprimir alguna de las siguientes páginas de este documento e incluirla en el reporte. Prácticas que no incluyan esta tabla se considerarán como prácticas no entregadas. La rúbrica permite conocer a detalle los criterios empleados para asignar la calificación final.

PRACTICA 10
Rubrica para grupo del laboratorio

Contenido	Penalizaciones	Puntaje máximo	Observaciones
Carátula *	-5P	5P	
objetivos e Introducción *	-5P	5P	
Actividades Práctica complementaria			
C1. Tabla de sentencias SQL	-10P Incompleta o ausente	10P	<ul style="list-style-type: none"> Si se detecta copia de sentencias la práctica se anula.
C2. Resultado del validador de la práctica 9 complementaria.	-10P por cada error detectado.	70P	<ul style="list-style-type: none"> La práctica no se evalúa si no se incluye la salida del validador.
Conclusiones, comentarios, recomendaciones *	-5P	5P	
Bibliografía *	-5P	5P	

* Ver Rubrica general de prácticas para mayores detalles en cuanto a los requisitos que debe cumplir el elemento de evaluación y los puntajes asignados.

PRACTICA 10
Rubrica para grupo de teoría plan 2010

Contenido	Penalizaciones	Puntaje máximo	Observaciones
Carátula *	-5P	5P	
objetivos e Introducción *	-5P	5P	
Actividades práctica complementaria			
C1. Tabla de sentencias SQL	-10P Incompleta o ausente	10P	<ul style="list-style-type: none"> Si se detecta copia de sentencias la práctica se anula.
C2. Resultado del validador de la práctica 9 complementaria.	-10P por cada error detectado.	70P	<ul style="list-style-type: none"> La práctica no se evalúa si no se incluye la salida del validador.
Conclusiones, comentarios, recomendaciones *	-5P	5P	
Bibliografía *	-5P	5P	

* Ver Rubrica general de prácticas para mayores detalles en cuanto a los requisitos que debe cumplir el elemento de evaluación y los puntajes asignados.

PRACTICA 10**Rubrica para grupo de teoría plan 2016 inscritos en otro grupo de laboratorio**

Contenido	Penalizaciones	Puntaje máximo	Observaciones
Carátula *	-5P	5P	
Actividades Práctica complementaria			
C1. Tabla de sentencias SQL	-10P Incompleta o ausente	10P	<ul style="list-style-type: none"> • Si se detecta copia de sentencias la práctica se anula.
C2. Resultado del validador de la práctica 9 complementaria.	-15P por cada error detectado.	80P	<ul style="list-style-type: none"> • La práctica no se evalúa si no se incluye la salida del validador.
Conclusiones, comentarios, recomendaciones *	-5P	5P	

* Ver Rubrica general de prácticas para mayores detalles en cuanto a los requisitos que debe cumplir el elemento de evaluación y los puntajes asignados.