

UNIVERSIDAD NACIONAL DE LA MATANZA

DEPARTAMENTO DE INGENIERÍA
E INVESTIGACIONES TECNOLÓGICAS

INGENIERIA EN INFORMATICA

BASE DE DATOS

Práctica de Ejercicios de SQL

Docentes

Lic. Gustavo Agustin
Ing. Alfonso Palomares
Ing. Natalia Crespo
Ing. Verónica Ichazo
Ing. Matías López
Ing. Fernando Ybarra
Ing. Hernán Jalil

2025

Dada la siguiente base de datos:

Almacén (**Nro**, Responsable)
Artículo (**CodArt**, descripción, Precio)
Material (**CodMat**, Descripción)
Proveedor (**CodProv**, Nombre, Domicilio, Ciudad)
Tiene (**Nro**, **CodArt**)
Compuesto_por (**CodArt**, **CodMat**)
Provisto_por (**CodMat**, **CodProv**)

1. Listar los nombres de los proveedores de la ciudad de La Plata.
2. Listar los números de artículos cuyo precio sea inferior a \$10.
3. Listar los responsables de los almacenes.
4. Listar los códigos de los materiales que provea el proveedor 10 y no los provea el proveedor 15.
5. Listar los números de almacenes que almacenan el artículo A.
6. Listar los proveedores de Pergamino que se llamen Pérez.
7. Listar los almacenes que contienen los artículos A y los artículos B (ambos).
8. Listar los artículos que cuesten más de \$100 o que estén compuestos por el material M1.
9. Listar los materiales, código y descripción, provistos por proveedores de la ciudad de Rosario.
10. Listar el código, descripción y precio de los artículos que se almacenan en A1.
11. Listar la descripción de los materiales que componen el artículo B.
12. Listar los nombres de los proveedores que proveen los materiales al almacén que Martín Gómez tiene a su cargo.
13. Listar códigos y descripciones de los artículos compuestos por al menos un material provisto por el proveedor López.
14. Hallar los códigos y nombres de los proveedores que proveen al menos un material que se usa en algún artículo cuyo precio es mayor a \$100.
15. Listar los números de almacenes que tienen todos los artículos que incluyen el material con código 123.
16. Listar los proveedores de Capital Federal que sean únicos proveedores de algún material.
17. Listar el/los artículo/s de mayor precio.
18. Listar el/los artículo/s de menor precio.
19. Listar el promedio de precios de los artículos en cada almacén.
20. Listar los almacenes que almacenan la mayor cantidad de artículos.
21. Listar los artículos compuestos por al menos 2 materiales.
22. Listar los artículos compuestos por exactamente 2 materiales.
23. Listar los artículos que estén compuestos con hasta 2 materiales.
24. Listar los artículos compuestos por todos los materiales.
25. Listar las ciudades donde existan proveedores que provean todos los materiales.

EJERCICIO 2

Dada la siguiente base de datos:

Proveedor (**NroProv**, NomProv, Categoria, CiudadProv)
Artículo (**NroArt**, Descripción, CiudadArt, Precio)
Cliente (**NroCli**, NomCli, CiudadCli)
Pedido (**NroPed**, **NroArt**, **NroCli**, **NroProv**, FechaPedido, Cantidad, PrecioTotal)
Stock (**NroArt**, **fecha**, cantidad)

1. Hallar el código (nroProv) de los proveedores que proveen el artículo a146.
2. Hallar los clientes (nomCli) que solicitan artículos provistos por p015.

3. Hallar los clientes que solicitan algún item provisto por proveedores con categoría mayor que 4.
4. Hallar los pedidos en los que un cliente de Rosario solicita artículos producidos en la ciudad de Mendoza.
5. Hallar los pedidos en los que el cliente c23 solicita artículos solicitados por el cliente c30.
6. Hallar los proveedores que suministran todos los artículos cuyo precio es superior al precio promedio de los artículos que se producen en La Plata.
7. Hallar la cantidad de artículos diferentes provistos por cada proveedor que provee a todos los clientes de Junín.
8. Hallar los nombres de los proveedores cuya categoría sea mayor que la de todos los proveedores que proveen el artículo "cuaderno".
9. Hallar los proveedores que han provisto más de 1000 unidades entre los artículos A001y A100.
10. Listar la cantidad y el precio total de cada artículo que han pedido los Clientes a sus proveedores entre las fechas 01-01-2004 y 31-03-2004 (se requiere visualizar Cliente, Artículo, Proveedor, Cantidad y Precio).
11. Idem anterior y que además la Cantidad sea mayor o igual a 1000 o el Precio sea mayor a \$ 1000.
12. Listar la descripción de los artículos en donde se hayan pedido en el día más del stock existente para ese mismo día.
13. Listar los datos de los proveedores que hayan pedido de todos los artículos en un mismo día. Verificar sólo en el último mes de pedidos.
14. Listar los proveedores a los cuales no se les haya solicitado ningún artículo en el último mes, pero sí se les haya pedido en el mismo mes del año anterior.
15. Listar los nombres de los clientes que hayan solicitado más de un artículo cuyo precio sea superior a \$100 y que correspondan a proveedores de Capital Federal. Por ejemplo, se considerará si se ha solicitado el artículo a2 y a3, pero no si solicitaron 5 unidades del artículo a2.

EJERCICIO 3

Dada la siguiente base de datos:

Producto (idProducto, nombre, descrip, estado, idProveedor)
Proveedor (idProveedor, nombre, respdCivil, cuit)
Dirección (idDir, idPers, calle, nro, piso, dpto)
Cliente (idCliente, nombre, respIVA, CUIL)
Vendedor (idEmpleado, nombre, apellido, DNI)
Venta (nroFactura, idCliente, fecha, idVendedor)
Detalle_venta(nroFactura, nro, idProducto, cantidad, precioUnitario)

NOTA: El precio unitario es necesario para almacenar los valores al momento de la venta

1. Indique la cantidad de productos que tiene la empresa.
2. Indique la cantidad de productos en estado 'en stock' que tiene la empresa.
3. Indique los productos que nunca fueron vendidos.
4. Indique la cantidad de unidades que fueron vendidas de cada producto.
5. Indique cual es la cantidad promedio de unidades vendidas de cada producto.
6. Indique quien es el vendedor con mas ventas realizadas.
7. Indique todos los productos de lo que se hayan vendido más de 15.000 unidades.
8. Indique quien es el vendedor con mayor volumen de ventas.

EJERCICIO 4

Dada la siguiente base de datos:

Persona (dni, nomPersona, telefono)
 Empresa (nomEmpresa, telefono)
 Vive (dni, calle, ciudad)
 Trabaja (dni, nomEmpresa, salario, feIngreso, feEgreso)
 Situada_En(nomEmpresa, ciudad)
 Supervisa (dniPer, dniSup)

- Encontrar el nombre de todas las personas que trabajan en la empresa "Banelco".
- Localizar el nombre y la ciudad de todas las personas que trabajan para la empresa "Telecom".
- Buscar el nombre, calle y ciudad de todas las personas que trabajan para la empresa "Paulinas" y ganan más de \$1500.
- Encontrar las personas que viven en la misma ciudad en la que se halla la empresa en donde trabajan.
- Hallar todas las personas que viven en la misma ciudad y en la misma calle que su supervisor.
- Encontrar todas las personas que ganan más que cualquier empleado de la empresa "Clarín".
- Localizar las ciudades en las que todos los trabajadores que vienen en ellas ganan más de \$1000.
- Listar los primeros empleados que la compañía "Sony" contrató.
- Listar los empleados que hayan ingresado en mas de 4 Empresas en el periodo 01-01-2000 y 31-03-2004 y que no hayan tenido menos de 5 supervisores

EJERCICIO 5

Dada la siguiente base de datos:

Pelicula (CodPel, Titulo, Duracion, Año, CodRubro)
 Rubro (CodRubro, NombRubro)
 Ejemplar (Codej, CodPel, Estado, Ubicación) **Estado:** Libre, Ocupado
 Cliente (CodCli, Nombre, Apellido, Dirección, Tel, Email)
 Prestamo (CodPrest, Codej, CodPel, CodCli, FechaPrest, FechaDev)

Nota: FechaDev → Se carga cuando el cliente efectúa la devolución del ejemplar.

- Listar los clientes que no hayan reportado préstamos del rubro "Policial".
- Listar las películas de mayor duración que alguna vez fueron prestadas.
- Listar los clientes que tienen más de un préstamo sobre la misma película (listar Cliente, Película y cantidad de préstamos).
- Listar los clientes que han realizado préstamos del título "Rey León" y "Terminador 3" (Ambos).
- Listar las películas más vistas en cada mes (Mes, Película, Cantidad de Alquileres).
- Listar los clientes que hayan alquilado todas las películas del video.
- Listar las películas que no han registrado ningún préstamo a la fecha.
- Listar los clientes que no han efectuado la devolución de ejemplares.
- Listar los títulos de las películas que tienen la mayor cantidad de préstamos.
- Listar las películas que tienen todos los ejemplares prestados.

EJERCICIO 6

Dada la siguiente base de datos:

Vuelo (NroVuelo, Desde, Hasta, Fecha)
 Avion_utilizado (NroVuelo, TipoAvion, NroAvion)
 Info_pasajeros(NroVuelo, Documento, Nombre, Origen, Destino)

Nota: Los vuelos no pueden tener más de 2 escalas y no hay cambio de tipo de avión para un mismo vuelo.

1. Hallar los números de vuelo desde el origen A hasta el destino F.
2. Hallar los tipos de avión que no son utilizados en ningún vuelo que pase por B.
3. Hallar los pasajeros y números de vuelo para aquellos pasajeros que viajan desde A a D pasando por B.
4. Hallar los tipos de avión que pasan por C.
5. Hallar por cada Avión la cantidad de vuelos distintos en que se encuentra registrado.
6. Listar los distintos tipo y nro. de avión que tienen a H como destino.
7. Hallar los pasajeros que han volado más frecuentemente en el último año.
8. Hallar los pasajeros que han volado la mayor cantidad de veces posible en un B-777.
9. Hallar los aviones que han transportado más veces al pasajero más antiguo.
10. Listar la cantidad promedio de pasajeros transportados por los aviones de la compañía, por tipo de avión.
11. Hallar los pasajeros que han realizado una cantidad de vuelos dentro del 10% en más o en menos del promedio de vuelos de todos los pasajeros de la compañía.

EJERCICIO 7

Dada la siguiente base de datos:

Auto (matrícula, modelo, año)
Chofer(nroLicencia, nombre, apellido, fecha_ingreso, teléfono)
Viaje(FechaHoraInicio, FechaHoraFin, chofer, cliente, auto,
kmTotales, esperaTotal, costoEspera, costoKms)
Cliente (nroCliente, calle, nro, localidad)

1. Indique cuales son los autos con mayor cantidad de kilómetros realizados en el último mes.
2. Indique los clientes que más viajes hayan realizado con el mismo chofer.
3. Indique el o los clientes con mayor cantidad de viajes en este año.
4. Obtenga nombre y apellido de los choferes que no manejaron todos los vehículos que disponemos.
5. Obtenga el nombre y apellido de los clientes que hayan viajado en todos nuestros autos.
6. Queremos conocer el tiempo de espera promedio de los viajes de los últimos 2 meses
7. Indique los kilómetros realizados en viajes por cada auto.
8. Indique el costo promedio de los viajes realizados por cada auto.
9. Indique el costo total de los viajes realizados por cada chofer en el último mes.
10. Indique la fecha inicial, el chofer y el cliente que hayan realizado el viaje más largo de este año.

EJERCICIO 8

Dada la siguiente base de datos:

Frecuenta (persona, bar)
Sirve (bar, cerveza)
Gusta (persona, cerveza)

Usar el SQL para hallar las personas que:

1. Frecuentan solamente bares que sirven alguna cerveza que les guste.
2. No frecuentan ningún bar que sirva alguna cerveza que les guste.

3. Frecuentan solamente los bares que sirven todas las cervezas que les gustan.
4. Frecuentan solamente los bares que no sirven ninguna de las cervezas que no les gusta.

EJERCICIO 9

Dada la siguiente base de datos:

Persona (**TipoDoc**, **NroDoc**, Nombre, Dirección, FechaNac, Sexo)
Progenitor (**TipoDoc**, **NroDoc**, **TipoDocHijo**, **NroDocHijo**)

1. Hallar para una persona dada, por ejemplo José Pérez, los tipos y números de documentos, nombres, dirección y fecha de nacimiento de todos sus hijos.
2. Hallar para cada persona los tipos y números de documento, nombre, domicilio y fecha de nacimiento de:
 - a. Todos sus hermanos, incluyendo medios hermanos.
 - b. Su madre
 - c. Su abuelo materno
 - d. Todos sus nietos

EJERCICIO 10

Dado el siguiente esquema:

Request (**NoRequest**, IP, Fecha, Hora, IDMetodo)
Page (**IP**, WebPage, IDAmbiente)
Método (**ID**, Clase, Metodo)
Ambiente (**ID**, Descripción)

Nota: El ambiente podrá ser Desarrollo, Testing o Producción. La función date() devuelve la fecha actual. Si se resta un valor entero a la función, restará días. El ejercicio consiste en indicar qué enunciado dio origen a cada una de las consultas:

1)

```
Select P.IP, count(distinct fecha), count(distinct IDMetodo),  
max(fecha)  
From Page P Inner join Request R on P.IP=R.IP  
Group by P.IP
```

2)

```
Select *  
From Ambiente A  
Where id not in  
    (Select idambiente  
     From Page P  
     Where not exists (  
         Select 1 From Request R  
         Where R.IP=P.IP and fecha>= date()-7))
```

3)

```
Select Fecha, count(*)  
From Request R  
Where hora between '00:00' and '04:00'  
and not exists(  
    select 1 from Page P  
    inner join Ambiente A on P.IDAmbiente = A.ID
```

```
        where R.IP=P.IP AND A.Descripcion='Desarrollo'    )
Group by fecha
Having count(*) >= 10
```

```
4)
Select W.WebPage, A.Descripcion, max(R.fecha), 'S'
From Request R
        Inner join WebPage W on R.IP=W.IP
        Inner join Ambiente A on A.id=W.IDAmbiente
Where R.Fecha>=date()-7 and W.Webpage like 'www%'
Group by W.WebPage, A.Descripcion
Having count(distinct fecha)>=7
```

```
5)
Select W.WebPage, A.Descripcion,
max(case when R2.fecha is null then '01/01/1900' else R2.fecha
end), 'N'
From WebPage W Left join (
        Select IP, max(fecha)From Request R
        Group by IP )
R2 on R2.IP = W.IP
Where W.Webpage like 'ftp%' and
not exists (
        Select 1
        from Request R
        where R.IP=W.IP and R.Fecha>=date()-7
        group by R.IP
        having count(*)>=7)
Group by W.WebPage, A.Descripcion
```

```
6)
insert into Page
select IP, 'Web ' + IDMetodo, '?'
from request R
where not exists (
        Select 1 from Page P
        where R.IP=P.IP )
and IDMetodo in (select ID from Metodo)
and fecha>=date()-30
```

Trabajando con objetos de base de datos

Los siguientes ejercicios corresponden con la creación de diferentes objetos de las bases de datos.

EJERCICIO 11

Dado el siguiente esquema de relación del Video Club “Orión”:

```

Película (CodPel, Titulo, Duracion, CodGenero, IdDirector)
Genero (Id, NombGenero)
Director (Id, NyA)
Ejemplar (nroEj, CodPel, Estado)
    {Estado: 1 Disponible, 0 No disponible}
Cliente (CodCli, NyA, Direccion, Tel, Email, Borrado)
    {Borrado: 1 Si, 2 No(Default) }
Alquiler (id, NroEj, CodPel, CodCli, FechaAlg, FechaDev)

```

1. Realice las sentencias DDL necesarias para crear en SQL una base de datos correspondiente al modelo relacional del enunciado.
2. Realice los INSERTs necesarios para cargar en las tablas creadas en el punto anterior los datos de 5 clientes, 10 películas (y tablas relacionadas a estas) y al menos 15 alquileres.
3. Agregue el atributo “año” en la tabla Película.
4. Actualice la tabla película para que incluya el año de lanzamiento de las películas en stock.
5. Queremos que al momento de eliminar una película se eliminen todos los ejemplares de la misma. Realice una CONSTRAINT para esta tarea.
6. Queremos que exista un borrado de lógico y no físico de clientes, realice un TRIGGER que usando el atributo “Borrado” haga esta tarea.
7. Elimine las películas de las que no se hayan alquilado ninguna copia.
8. Elimine los clientes sin alquileres.

EJERCICIO 12

Dado el siguiente esquema de relación:

Producto (**CodProd**, Descripcion, CodProv, StockActual)
 Stock (**Nro**, **Fecha**, **CodProd**, Cantidad)
 Proveedor (**CodProv**, RazonSocial, FechaInicio)

Realizar las siguientes tareas utilizando lenguaje SQL:

1. **p_EliminaSinstock():** Realizar un procedimiento que elimine los productos que no poseen stock.
2. **p_ActualizaStock():** Para los casos que se presenten inconvenientes en los datos, se necesita realizar un procedimiento que permita actualizar todos los Stock_Actual de los productos, tomando los datos de la entidad Stock. Para ello, se utilizará como stock válido la última fecha en la cual se haya cargado el stock.
3. **p_DepuraProveedor():** Realizar un procedimiento que permita depurar todos los proveedores de los cuales no se posea stock de ningún producto provisto desde hace más de 1 año.

4. **p_InsertStock(nro,fecha,prod,cantidad)**: Realizar un procedimiento que permita agregar stocks de productos. Al realizar la inserción se deberá validar que:
 - a. El producto debe ser un producto existente
 - b. La cantidad de stock del producto debe ser cualquier número entero mayor a cero.
 - c. El número de stock será un valor correlativo que se irá agregando por cada nuevo stock de producto.
5. **tg_CrearStock**: Realizar un trigger que permita automáticamente agregar un registro en la entidad Stock, cada vez que se inserte un nuevo producto. El stock inicial a tomar, será el valor del campo Stock_Actual.
6. **p_ListaSinStock()**: Crear un procedimiento que permita listar los productos que no posean stock en este momento y que no haya ingresado ninguno en este último mes. De estos productos, listar el código y nombre del producto, razón social del proveedor y stock que se tenía al mes anterior.
7. **p_ListaStock()**: Realizar un procedimiento que permita generar el siguiente reporte:

| Fecha | > 1000 | < 1000 | = 0 |
|------------|--------|--------|-------|
| 01/08/2009 | 100 | 8 | 3 |
| 03/08/2009 | 53 | 50 | 7 |
| 04/08/2009 | 50 | 20 | 40 |
| | | | |

En este listado se observa que se contará la cantidad de productos que posean a una determinada fecha más de 1000 unidades, menos de 1000 unidades o que no existan unidades de ese producto.

Según el ejemplo, el 01/08/2009 existen 100 productos que poseen más de 1000 unidades, en cambio el 03/08/2009 sólo hubo 53 productos con más de 1000 unidades.

8. El siguiente requerimiento consiste en actualizar el campo stock actual de la entidad producto, cada vez que se altere una cantidad (positiva o negativa) de ese producto. El stock actual reflejará el stock que exista del producto, sabiendo que en la entidad Stock se almacenará la cantidad que ingrese o egrese. Además, se debe impedir que el campo "Stock actual" pueda ser actualizado manualmente. Si esto sucede, se deberá dar marcha atrás a la operación indicando que no está permitido.

EJERCICIO 13

Dado el siguiente esquema de relación:

Medición (**fecha**, **hora**, **métrica**, temperatura, presión, humedad, nivel)

Nivel (**código**, descripción)

Realizar las siguientes consultas utilizando lenguaje SQL:

1. **p_CrearEntidades()**: Realizar un procedimiento que permita crear las tablas de nuestro modelo relacional.

2. **f_ultimamedicion(Métrica):** Realizar una función que devuelva la fecha y hora de la última medición realizada en una métrica específica, la cual será enviada por parámetro. La sintaxis de la función deberá respetar lo siguiente:
 Fecha/hora = f_ultimamedicion(vMétrica char(5))
 Ejemplificar el uso de la función.
3. **v_Listado:** Realizar una vista que permita listar las métricas en las cuales se hayan realizado, en la última semana, mediciones para todos los niveles existentes. El resultado del listado deberá mostrar, el nombre de la métrica que respete la condición enunciada, el máximo nivel de temperatura de la última semana y la cantidad de mediciones realizadas también en la última semana.
4. **p_ListaAcumulados(finicio,ffin):** Realizar un procedimiento que permita generar una tabla de acumulados diarios de temperatura por cada métrica y por cada día. El procedimiento deberá admitir como parámetro el rango de fechas que mostrará el reporte. Además, estas fechas deben ser validadas. El informe se deberá visualizar de la siguiente forma:

| Fecha | Métrica | Ac.DiarioTemp | Ac.Temp |
|------------|---------|---------------|---------|
| 01/03/2009 | M1 | 25 | 25 |
| 02/03/2009 | M1 | 20 | 45 |
| 03/03/2009 | M1 | 15 | 60 |
| 01/03/2009 | M2 | 15 | 15 |
| 02/03/2009 | M2 | 10 | 25 |

5. **p_InsertMedicion(fecha,hora, metrica,temp,presion,hum,niv):**
 Realizar un procedimiento que permita agregar una nueva medición en su respectiva entidad. Los parámetros deberán ser validados según:
 - a. Para una nueva fecha hora, no puede haber más de una medida por métrica
 - b. El valor de humedad sólo podrá efectuarse entre 0 y 100.
 - c. El campo nivel deberá ser válido, según su correspondiente entidad.
6. **p_depuraMedicion(días):** Realizar un procedimiento que depure la tabla de mediciones, dejando sólo las últimas mediciones. El resto de las mediciones, no deben ser borradas sino trasladadas a otra entidad que llamaremos Medicion_Hist. El proceso deberá tener como parámetro la cantidad de días de retención de las mediciones.
7. **tg_descNivel:** Realizar un trigger que coloque la descripción en mayúscula cada vez que se inserte un nuevo nivel.

EJERCICIO 14

Dado el siguiente esquema de relación:

```

Festejo  (NroFestejo, descripción, fecha, nrocliente)
Contrata (NroFestejo, Item, NroServicio, HDesde, HHasta)
Servicio (NroServicio, Descripción, Precio)
Cliente  (NroCliente, RazonSocial)
  
```

Realizar las siguientes consultas utilizando lenguaje SQL:

1. **p_Servicios(FDesde, FHasta):** Crear un procedimiento almacenado que permita listar aquellos servicios que fueron contratados en todos los festejos del período enviado por parámetro. De dichos servicios mostrar el nombre y la cantidad de horas que fueron contratadas en el período enviado por parámetro. Ejemplificar la invocación del procedimiento.
2. Agregar el campo “Tiempo” en la tabla “Contrata” de tipo smallint, que no acepte nulos y que posea como valor predeterminado 0 (cero). Este campo servirá para que ya se encuentre precalculado la cantidad de minutos que fue contratado el servicio, sin necesidad de realizar el cálculo con los campos de la tabla.
3. **tg_Tiempo:** Crear un trigger que cada vez que se cambia la hora desde/hasta o bien se agregue un nuevo servicio contratado, recalcule el campo “Tiempo” con el tiempo en minutos del servicio. Validar que la hora desde no puede ser posterior a la hora hasta, si esto sucede se deberá avisar y volver atrás la operación. Además, tener en cuenta las actualizaciones masivas. Ejemplificar la invocación del trigger.