

SQL – EJERCITACIÓN

*Dadas las siguientes relaciones, resolver en **SQL** las consultas planteadas:*

SOCIO = (Cod_Socio, DNI, Apellido, Nombre, Fecha_Nacimiento, Fecha_Ingreso)

LIBRO = (ISBN, Titulo, Cod_Genero, Descripcion)

COPIA = (ISBN, Nro_Ejemplar, Estado)

EDITORIAL = (Cod_Editorial, Denominacion, Telefono, Calle, Numero, Piso, Dpto.)

LIBRO-EDITORIAL = (ISBN, Cod_Editorial, Año_Edicion)

GENERO = (Cod_Genero, Nombre)

PRESTAMO = (Nro_Prestamo, Cod_Socio, ISBN, Nro_Ejemplar, Fecha_Prestamo, Fecha_Devolucion)

1. Listar el Apellido y Nombre de aquellos socios cuya fecha de ingreso este entre el 01/9/2008 y el 30/09/2008. Dicho listado deberá estar ordenado por Apellido.
2. Listar el Titulo, Género (el Nombre del Género) y Descripción de aquellos libros editados por la editorial "Nueva Editorial". Dicho listado deberá estar ordenado por Titulo.
3. Listar el Apellido, Nombre, Fecha de Nacimiento y cantidad de prestamos de aquellos socios que tengan más de 10 préstamos. Dicho listado deberá estar ordenado por Apellido.
4. Listar el DNI, Apellido y Nombre de aquellos socios que tengan préstamos sin devolver de libros editados por la editorial "Gran Editorial". Dicho listado deberá estar ordenado por Apellido y Nombre.
5. Proyectar que cantidad de socios tienen actualmente libros prestados cuyo estado sea "Bueno".
6. Listar el Titulo, Genero, Denominación de la editorial y Año de edición de aquellos libros editados entre los años 1980 y 2000. Dicho listado deberá estar ordenado por año de edición y titulo del libro.
7. Agregar un nuevo socio con el DNI, Apellido, Nombre y Fecha de nacimiento que prefiera.
8. Modificar el titulo del libro cuyo ISBN es 2222-2020 por el titulo "El Código".
9. Listar el DNI, Apellido y Nombre de aquellos socios cuyo Apellido contenga la secuencia de letras „bra" o posea préstamos actuales de libros cuyo género empieza con la letra „T". Dicho listado deberá estar ordenado por Apellido.
10. Listar para cada Género la cantidad de préstamos en que estuvo involucrado. Dicho Listado deberá estar ordenado en forma descendente de acuerdo a la cantidad de préstamos resultante para cada género.

Dadas las siguientes relaciones, resolver en **SQL** las consultas planteadas:

PERSONA = (DNI, Apellido, Nombre, Fecha_Nacimiento, Estado_Civil, Genero)

ALUMNO = (DNI, Legajo, Año_Ingreso)

PROFESOR = (DNI, Matricula, Nro_Expediente)

TITULO = (Cod_Titulo, Nombre, Descripción)

TITULO-PROFESOR = (Cod_Titulo, DNI, Fecha)

CURSO = (Cod_Curso, Nombre, Descripción, Fecha_Creacion, Duracion)

ALUMNO-CURSO = (DNI, Cod_Curso, Año, Desempeño, Calificación)

PROFESOR-CURSO = (DNI, Cod_Curso, Fecha_Desde, Fecha_Hasta)

1. Listar el DNI, Apellido, Nombre, Género y Fecha de nacimiento de los alumnos y profesores de la institución detallando por cada uno que rol cumple, alumno o profesor. Además, el género se debe proyectar de forma que se lea „femenino“ o “masculino” y no „F“ o „M“. Dicho listado deberá estar ordenado por Apellido y Nombre.
2. Listar el DNI, Apellido, Nombre y Matricula de aquellos profesores que posean tres títulos o más. Dicho listado deberá estar ordenado por Apellido y Nombre.
3. Listar el DNI, Apellido, Nombre, Cantidad de horas y Promedio de horas que dicta cada profesor. La cantidad de horas se calcula como la suma de la duración de todos los cursos que dicta. Dicho listado deberá estar ordenado por Apellido y Nombre.
4. Listar el DNI, Apellido, Nombre y Calificación de aquellos alumnos que obtuvieron una calificación superior a 5 en los cursos que dicta el profesor “Rodríguez”. Dicho listado deberá estar ordenado por Apellido.
5. Listar el Nombre, Descripción, Fecha de creación del curso que posea más horas de duración y del que menos horas posea.
6. Listar el DNI, Apellido, Nombre, Legajo y Año_Ingreso de aquellos alumnos que cursaron todos los cursos sin importar la calificación que obtuvieron.
7. Agregar un profesor con los datos que prefiera y agregarle el título con código: 20.
8. Modificar el año de ingreso del alumno cuyo legajo es „1111/1“, el nuevo año a registrar es 2000.
9. Dar de baja el curso con código 5. Realizar todas las bajas necesarias para no dejar el conjunto de relaciones en estado inconsistente.

Dadas las siguientes relaciones, resolver en **SQL** las consultas planteadas:

CLIENTE (Nro_Cliente, DNI, Apellido, Nombre, Renta_Anual, Tipo_Cliente)

EMBARQUE (Nro_Embarque, Nro_Cliente, Peso, Nro_Camión, Nro_Ciudad, Fecha)

CAMION (Nro_Camión, Marca, Modelo, Año)

CIUDAD (Nro_Ciudad, Nombre, Población)

1. Listar la Marca, Modelo y Año de los camiones que han llevado paquetes (embarques) igual o superior a 1000 Kg.
2. Listar todos los datos de los embarques de más de 100 Kg.
3. Listar DNI, Apellido y Nombre de aquellos clientes cuya renta anual es superior a 100000 pesos. Dicho listado deberá estar ordenado por Apellido y Nombre.
4. Listar el DNI, Apellido y Nombre de aquellos clientes cuyo tipo de cliente sea „Minorista“ y que posean embarques a ciudades cuyo nombre empieza con la letra „C“. Dicho listado deberá estar ordenado por Apellido y Nombre.
5. Listar el Nombre y Población de aquellas ciudades que han recibido embarques cuyo peso sea igual o superior a 1000 Kg. Dicho listado deberá estar ordenado por Nombre de la ciudad.
6. Listar el Nombre de aquellas ciudades que han recibido embarques de clientes que tienen más de 100000 pesos de renta anual o que posean una población superior a 30000 habitantes.
7. Listar el DNI, Apellido, Nombre y Renta Anual de aquellos clientes que han tenido embarques transportados en cada camión. Dicho listado deberá estar ordenado por Apellido y Nombre.
8. Proyectar el peso promedio de los embarques destinados a ciudades cuya población es menor a 30000 habitantes.
9. Listar DNI, Apellido y Nombre de aquellos clientes en que el peso de todos sus embarques supera los 1000 kg. Dicho listado deberá estar ordenado por Apellido y Nombre.
10. Listar el Nombre de aquellas ciudades que han recibido embarques de todos los clientes.
11. Listar para cada cliente el DNI, Apellido, Nombre, cantidad de embarques enviados y peso promedio de todos sus embarques.
12. Listar para cada ciudad el Nombre y el peso máximo de todos los paquetes que hayan sido enviado a la ciudad correspondiente.
13. Dar de baja a todas las ciudades con población menor a 2000 habitantes. Realizar todas las bajas necesarias para no dejar el conjunto de relaciones en estado inconsistente.
14. Convierta el peso de cada embarque a libras, para ello se sabe que una libra son 2,2 Kg. (aproximadamente).

Dadas las siguientes relaciones, resolver en **SQL** las consultas planteadas:

Zona (IdZona, NombreZona)

Cine (IdCine, NombreCine, IdZona, Ubicación)

Sala (IdCine, NumeroSala, Capacidad)

Película (IdPelícula, Título, Año, Género, Duración, Calificación, Sinopsis)

ActorDirector (IdAD, NombreAD)

Dirección (IdPelícula, IdAD)

Actuación (IdPelícula, IdAD)

Función (IdCine, NumeroSala, IdPelícula, Fecha, Horario)

1. Listar el Nombre y Ubicación de todos los cines de la zona “La Plata” que tengan función para la película “Spiderman 3”. Dicho listado deberá estar ordenado por Nombre.
2. Listar el Título, Calificación y Sinopsis de todas las películas dirigidas por “Oliver Stone” y que tenga al menos un actor cuyo nombre contenga la cadena “ab”. Dicho listado deberá estar ordenado por Título.
3. Listar los Títulos de las películas que tengan función en todas las zonas.
4. Listar el Nombre y Ubicación de aquellos cines que tengan función para más de 5 películas. Dicho listado deberá estar ordenado por Nombre en forma descendente.
5. Listar la cantidad de cines distintos en las que se exhibe cada película y proyectar además el título de la misma. Dicho listado deberá estar ordenado por cantidad en orden descendente.
6. Listar para cada zona que cantidad de películas distintas se exhiben. Dicho listado deberá estar ordenado por cantidad.

Dadas las siguientes relaciones, resolver utilizando **SQL** las consultas planteadas:

Material (cod_material, tipo, género, título, descripción, duración, cantidad_capítulos)

Capítulo (cod_material, nro_capítulo, descripción)

Canal (cod_canal, nombre, número, categoría)

Programación (cod_canal, cod_material, fecha, hora, nro_capítulo(opcional))

1. Listar el título, género y descripción del material que se emitirá hoy con horario de comienzo entre las 16:00 hs. y las 20:00 hs. en canales de categoría "cine y series", el listado además debe incluir la hora precisa de comienzo, el número de canal y la descripción del capítulo, en caso de que figure su número.
2. Listar el título, género y descripción del material que se emitirá hoy de tipo "serie" y que incluya el término "drama" en género, el listado además debe incluir la hora precisa de comienzo, el número de canal y la descripción del capítulo, en caso de que figure su número.
3. Listar nombre y número de aquellos canales que poseen programación para todos los materiales. (Resolver si o si utilizando la cláusula Not Exists).
4. Listar título, descripción, tipo y duración de aquellos materiales que posean programación para todos sus capítulos.
5. Listar para cada material el título, descripción y la cantidad de canales distintos en los que tiene programación.
6. Asignar 0 al campo cantidad de capítulos y „null" al campo tipo a aquellos materiales que posean programación en un único canal.

Dadas las siguientes relaciones, resolver utilizando **SQL** las consultas planteadas:

Provincia = (IdProvincia, Nombre)

Ciudad (IdCiudad, Nombre, IdProvincia)

Canal (IdCanal, Denominación, Dirección, Teléfono, IdCiudad)

Dibujo (IdDibujo, Nombre, Descripción, Año_Creación)

Personaje = (IdPersonaje, Nombre, Descripción, IdDibujo)

Programación = (IdCanal, IdDibujo, Fecha, Horario)

1. Listar la denominación, dirección y teléfono de aquellos canales que pertenezcan a la provincia de "San Luis" y que tengan programación para el dibujo "Thundercats". Dicho listado deberá estar ordenado por denominación.
2. Listar el nombre y año de creación de todos los dibujos que posean año de creación entre los años 1900 y 2000 o que tengan al menos un personaje cuyo nombre finalice con la cadena "abe". Dicho listado deberá estar ordenado por nombre.
3. Listar el identificador y el Nombre de los dibujos que tengan programación en todos los canales.
4. Listar el Nombre y Descripción de aquellos dibujos que tengan programación en la provincia de "San Luis" y no posean programación en la provincia de "San Juan".
5. Listar la Denominación, Dirección y Teléfono de aquellos canales que tengan programación para más de 10 dibujos.

Dadas las siguientes relaciones, resolver utilizando **SQL** las consultas planteadas:

Congreso = (IdCongreso, Nombre, Fecha_Inicio, Fecha_Fin)

Área = (IdArea, Nombre, Descripción)

Posee = (IdCongreso, IdArea)

Artículo = (IdArtículo, Título, Cantidad_Hojas, Fecha_Recepción, *IdArea*, *IdCongreso*)

Autor = (Dni, Nombre, Apellido)

Escribe = (IdArtículo, Dni)

1. Listar el nombre y apellido de aquellos autores que presentaron artículos en el congreso "WICC 2009" pero que no hayan presentado en el congreso "TE&ET 2009".
2. Anular el id de congreso (IdCongreso) de aquellos artículos presentados en el congreso "CACIC 2008" que tengan código de área nula (artículos rechazados).
3. Listar el nombre, fecha de inicio y fecha de fin de aquellos congresos en los que se hayan presentado más de 100 artículos.
4. Listar la cantidad de artículos recibidos por área para todos los congresos. Dicho listado deberá estar ordenado nombre de congreso, nombre de área y cantidad.
5. Listar el título de todos los artículos enviados al congreso "CACIC 2010" y el nombre del área a la que pertenece. Tener en cuenta que algunos autores no definen el área del congreso, pero es importante mostrar el artículo en el listado.
6. Listar el nombre de todos los congresos que hayan recibido artículos para todas las áreas.

Dadas las siguientes relaciones, resolver utilizando **SQL** las consultas planteadas:

Cliente (nroClte, apellido, nombres, domicilio, teléfono)

Pago (nroClte, año, mes, veces_por_semana, fechaPago, importePagado)

Actividad (codActiv, nombre)

Abono (nroClte, año, mes, codActiv)

Asistencia (codActiv, fecha, nroClte)

1. Reportar el número de cliente y la cantidad de veces por semana que puede asistir” Luque, Joaquín” en diciembre de 2009 junto con la cantidad de veces que asistió entre los días 14 y 17.
2. Reportar la fecha de pago del cliente 1749 para el abono a Stretching de diciembre de 2009.
3. Registrar la asistencia del cliente Luque, Joaquín a la actividad Stretching el “17/12/09”.
4. Reportar apellido y nombre de los clientes que asistieron por primera vez a “Aeróbica” entre el 1 y el 30 de noviembre de 2009.
5. Reportar el apellido y nombre de los clientes que no asistieron a alguna actividad, entre el 14 y el 19, para la que pagaron abono en el mes de diciembre de 2009.
6. Cambiar el abono a la actividad Stretching de Palacios, Virginia de diciembre de 2009 por un abono a la actividad Aeróbica.

Dadas las siguientes relaciones, resolver utilizando **SQL** las consultas planteadas:

Técnico (codTec, nombre, especialidad) // técnicos

Repuesto (codRep, nombre, stock, precio) // repuestos

RepuestoReparacion (nroReparac, codRep, cantidad, precio) // repuestos utilizados en reparaciones.

Reparación (nroReparac, codTec, precio_total, fecha) // reparaciones realizadas.

1. Listar para cada repuesto nombre, stock y cantidad reparaciones en las que participó. Si un repuesto no participo en alguna reparación igual debe aparecer en dicho listado.
2. Listar para cada técnico el nombre, especialidad y la cantidad de reparaciones realizadas en el mes de octubre. Dicho listado deberá estar ordenado por nombre de manera descendente.
3. Listar nombre, stock y precio de todos los repuestos con stock mayor a 0 y que dicho repuesto no haya estado en reparaciones durante el mes de octubre.
4. Proyectar precio, fecha y precio total de aquellas reparaciones donde se utilizo algún repuesto con precio en el momento de la reparación mayor a \$100 y menor a \$500.
5. Listar para cada tecnico la cantidad de reparaciones que realizó. Se debe proyectar el nombre y especialidad del técnico, junto a la cantidad solicitada.
6. Listar el número de reparación y precio total de aquellas reparaciones que necesitaron más de 5 repuestos.