

75.15 / 95.05 / TA044

Base de Datos

Guía de ejercicios - Versión 1.1.3

Profesor adjunto: José Luis Cabrera.

JTP: Adrián Sergio Bernardi.

Ayudante: Kailásh Aquista.

Guía 1: Modelo Entidad Interrelación.

Ejercicio 1:

Un comité deportivo internacional requiere un sistema para administrar las competencias de los deportistas de cada país. Para eso solicita un diseño de una base de datos que permita almacenar y consultar:

- Deportistas indicando donde nacieron y sus datos principales.
- Competencias en las cuales compiten los deportistas.
- Resultados de las competencias.

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Ejercicio 2:

Una empresa de subterráneos necesita diseñar una base de datos, para almacenar y consultar:

- Líneas de subterráneo que gestiona.
- Estaciones de cada línea de subterráneo.
- Cantidad de empleados por línea de subterráneo.
- Horarios de funcionamiento de cada línea de subterráneo.
- Cantidad de servicios por línea.
- Duración de un viaje (cabecera inicio-cabecera fin) por línea.
- Frecuencia de cada línea.

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Ejercicio 3:

Dadas las siguientes entidades y relaciones del Modelo Entidad-Interrelación:

- Golfista: { matrícula, nombre, apellido, ranking, torneos_ganados }
- Campo_De_Golf: { nombre, hoyos, es_publico, ciudad }
- Juega_En: { fecha }

indicar cuales son las superclaves, las claves candidatas, y elija las claves primarias.

Ejercicio 4:

Dadas las siguientes entidades y relaciones del Modelo Entidad-Interrelación:

- Club: { nombre, año_fundacion, colores_camiseta, ciudad }
 - Presidente: { dni, nombre, apellido, edad }
 - Dirige_Club: { fecha_desde, fecha_hasta }
 - Jugador: { dni, nombre, apellido, numero_camiseta }
 - Juega_En: { fecha_desde, fecha_hasta }
- Realizar el diagrama Entidad - Relación.
 - Indicar cuales son las superclaves, las claves candidatas, y elija las claves primarias.

Ejercicio 5:

Dadas las siguientes entidades y relaciones del Modelo Entidad-Interrelación:

- Obra: { nombre, género }
 - Actor: { nombre, apellido, fecha_nacimiento, edad }
 - Actúa: { personaje }
 - Teatro: { nombre, dirección, capacidad, ciudad }
- Realizar el diagrama Entidad - Relación.
 - indicar cuales son las relaciones, las superclaves, las claves candidatas, y elija las claves primarias.

Ejercicio 6 (Agregación):

Una asociación de fútbol planea diseñar una base de datos con datos sobre los jugadores que participan en clubes y selecciones. Para eso, nos encarga realizar el diseño de la base de datos. Es interesante para la asociación poder tener los datos de los jugadores tales como DNI, nombre, edad, posición, cantidad de partidos jugados, cantidad de goles, etc. También le interesa a la asociación poder tener los datos de los clubes en los que juega cada jugador, y los datos de las selecciones de cada jugador. En la base de datos puede haber jugadores retirados, que ya no juegan en ningún club, jugadores que actualmente juegan en un club y jugadores que juegan en clubes y selecciones.

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Ejercicio 7 (Especialización / Generalización):

Un negocio de venta de electrodomésticos requiere modelar su base de datos, para mantener y consultar información importante para la empresa. En particular están interesados en almacenar los productos que están a la venta, sus fabricantes, sus características, sus cantidades en stock. También se desea mantener en la base de datos, las ventas realizadas de cada producto indicando qué empleados realizaron dichas ventas (para calcular las comisiones).

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Ejercicio 8 (Especialización / Generalización):

Una importante fábrica de vehículos está interesada en gestionar su base de datos, y nos pide realizar su modelo. La fábrica construye autos, colectivos, y camiones de carga. Además posee los planos para construir dichos vehículos pero de distintas marcas. La empresa está interesada en poder conocer la capacidad de vehículos que puede fabricar, por mes, y de cada marca. Además quiere tener la información de las prestaciones de cada vehículo, tales como la cantidad de ruedas, cantidad máxima de ocupantes, y capacidad del tanque de combustible, entre otras.

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Ejercicio 9 (Unión):

Una agencia de viajes realiza ventas por internet, y nos pide modelar su base de datos. Es interesante poder tener acceso a los datos para realizar consultas y modificaciones. Entre los datos necesarios a guardar se encuentran, el viaje, su descripción, precio, el cliente, y los países de origen y destino. Como simplificación, se supone que la persona contrata un viaje hacia un único país. También es importante destacar que la empresa permite comprar viajes a sus empleados los cuales, serán tratados como clientes al momento de realizar la compra.

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Ejercicio 10:

Un registro civil se encarga de realizar casamientos entre personas, y requiere modelar su base de datos. Tener en cuenta que una persona se casa con otra persona. Además las personas poseen los atributos DNI, nombre, y apellido:

Diseñar la base de datos, utilizando el modelo Entidad-Interrelación.

Guía 2: Modelo Relacional.

Ejercicio 1:

Definir el modelo de datos(MR) definiendo Relaciones, atributos(con sus respectivos dominios), PK, FK para el siguiente caso, especificando las suposiciones que se hagan. Un supermercado vende distintos productos. Cada producto tiene un stock. Los productos pertenecen a distintas familias. Se prevé que en el futuro se puedan agregar otras formas de venta. En algunas familias los productos se venden por peso y en otras por unidad. Los productos se almacenan en góndolas, que se ubican en pasillos. La empresa desea manejar el stock.

Ejercicio 2:

La misma empresa del ejercicio 1 solicita a su sector IT el manejo del personal. Cada sector tiene un supervisor y muchos empleados de otras categorías. Además existe un gerente del local. Cada colaborador, independientemente de su categoría, posee un horario de trabajo y un sueldo. Se desea incorporar esta información al modelo relacional.

Ejercicio 3:

Un lavadero de autos nos solicita un sistema de gestión de base de datos, para fidelizar a sus clientes. El lavadero posee clientes, los cuales piden lavar un vehículo. Hay distintos tipos de vehículos (motos, autos, y camionetas) para los cuales su lavado cuesta un precio diferente. Además posee dos tipos de lavados diferentes: el común, y el completo (cada uno con su precio).

El lavadero desea poder consultar cuántos clientes atendió históricamente, la cantidad de vehículos que se lavaron por tipo, y la ganancia total obtenida.

Diseñar la base de datos, utilizando el Modelo Relacional. Especificar las Relaciones, los atributos de las Relaciones (con sus dominios), las claves primarias, y las claves foráneas.

Ejercicio 4:

Un kiosko nos contrata para crear y mantener su sistema de gestión de base de datos. Al dueño del kiosko, le interesa conocer cuántos productos posee almacenados para vender y de qué tipo, sus precios, sus vencimientos, la ganancia histórica del kiosko, y tener disponible un listado de proveedores para cada producto. Tener en cuenta que un lote de productos puede tener un vencimiento, y otro lote del mismo tipo de producto puede tener otro vencimiento.

Diseñar la base de datos, utilizando el Modelo Relacional. Especificar las Relaciones, los atributos de las Relaciones (con sus dominios), las claves primarias, y las claves foráneas.

Ejercicio 5:

Una biblioteca de una municipalidad nos pide construir su sistema de gestión de base de datos.

La biblioteca requiere conocer qué libros posee, su cantidad de libros por género, los autores de los libros, en qué sala y estantería se encuentra cada libro, y qué socios pidieron prestado un libro al momento de realizar la consulta.

Los libros poseen como datos, su género, su autor, su ISBN (código de identificación internacional), la edición, y la editorial. Se desea conocer datos de los autores tales como sus nombres, apellidos, y fecha de nacimiento. Los socios poseen los típicos datos de una persona (nombre, apellido, etc), más un identificador a elección.

Diseñar la base de datos, utilizando el Modelo Relacional. Especificar las Relaciones, los atributos de las Relaciones (con sus dominios), las claves primarias, y las claves foráneas.

Ejercicio 6:

Un negocio de venta de bicicletas, maneja el stock de las mismas, que pueden ser ebike o tradicional, en el futuro prevén incorporar nuevos productos, como por ejemplo monopatines. Cada bicicleta tiene su número de serie que es importante por el tema de la garantía. El dueño del negocio desea conocer la información acerca de todas las bicicletas, en todos los estados, en stock, vendida con garantía vigente, vendida con garantía vencida, en reparación por garantía, más que puedan sumarse en el futuro. Diseñar la base de datos, utilizando el Modelo Relacional. Especificar las Relaciones, los atributos de las Relaciones (con sus dominios), las claves primarias, y las claves foráneas.

Ejercicio 7:

Una estación de servicio nos pide ayuda para construir su sistema de gestión de base de datos.

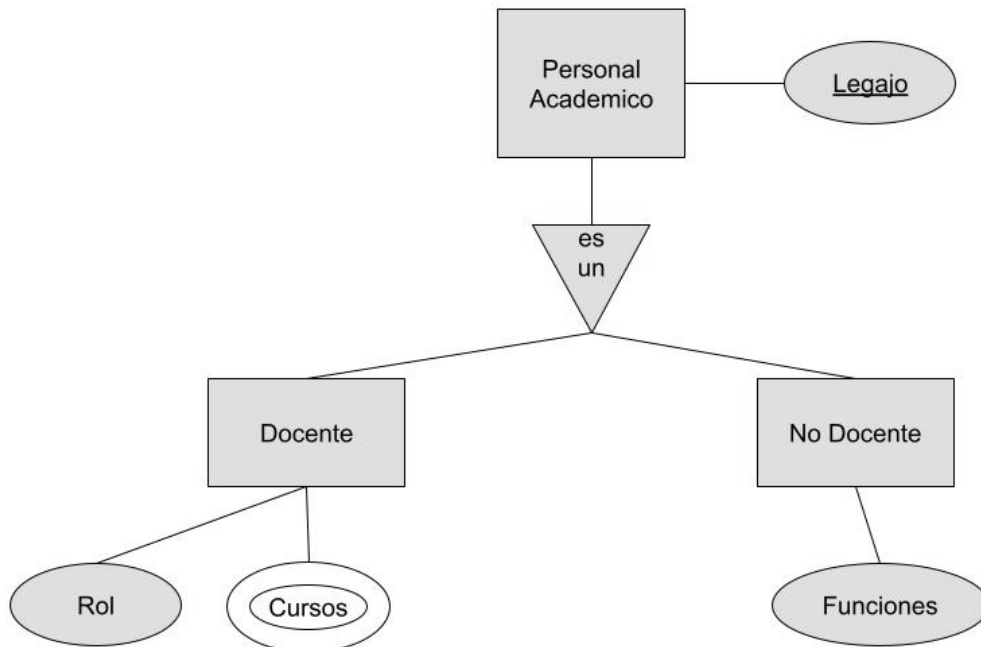
En la estación de servicio se venden distintos tipos de combustible, distintos tipos de lubricantes, y accesorios para el auto (tales como productos limpiavidrios, gamuzas, etc). Adicionalmente la estación vende dos servicios: inflar las ruedas, controlar y recargar agua del motor.

Los socios de la estación de servicio están interesados en poder consultar cuantos clientes se atendieron históricamente, qué servicios se vendieron, la ganancia histórica, poder modificar los precios de los combustibles, lubricantes, accesorios y servicios, e informar qué clientes gastaron más dinero.

Diseñar la base de datos, utilizando el Modelo Relacional. Especificar las Relaciones, los atributos de las Relaciones (con sus dominios), las claves primarias, y las claves foráneas.

Ejercicio 8:

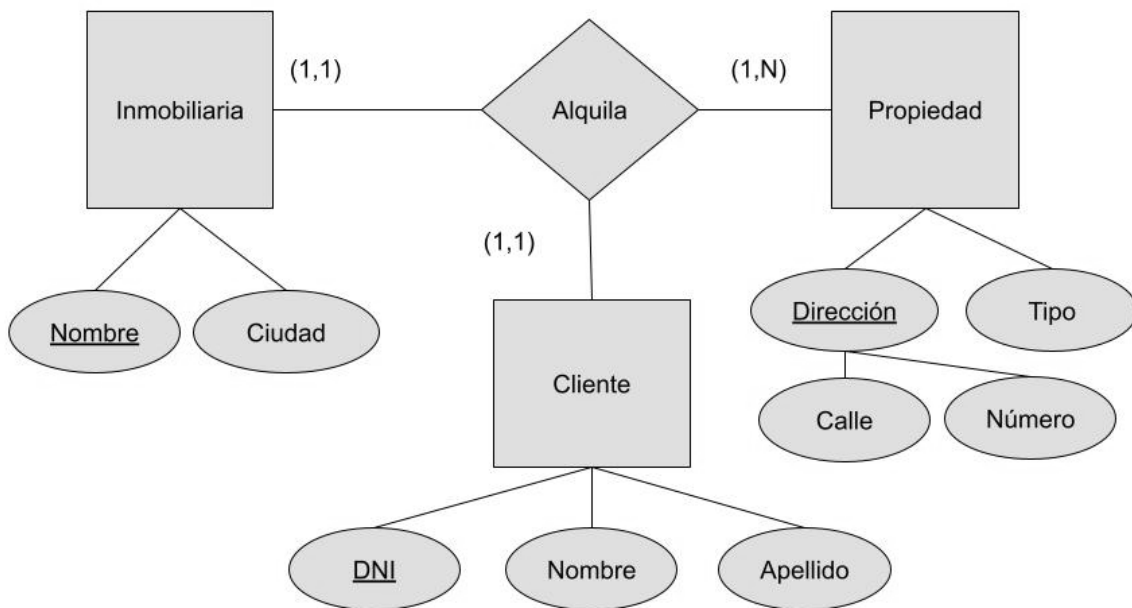
Dado el siguiente Diagrama del Modelo Entidad Relación, realizar el pasaje a Modelo Relacional indicando relaciones, atributos, claves primarias, y claves foráneas:



Ejercicio 9:

Una empresa que gestiona avisos de inmobiliarias, permite a las mismas alquilar propiedades a sus clientes.

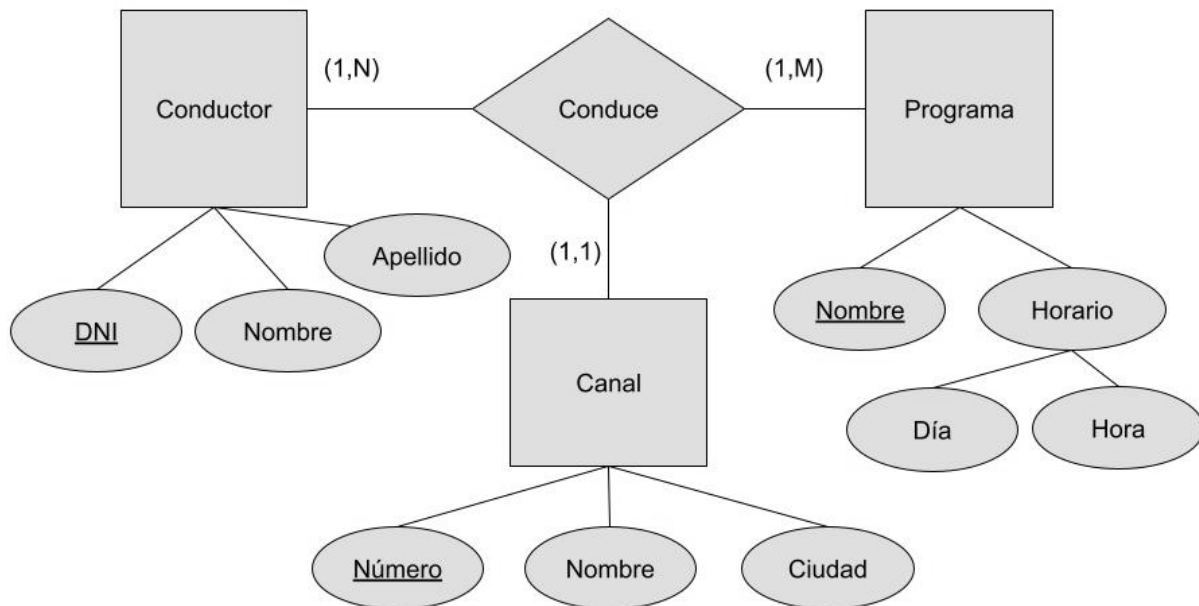
Dado el siguiente Diagrama del Modelo Entidad Relación, realizar el pasaje a Modelo Relacional indicando relaciones, atributos, claves primarias, y claves foráneas:



Ejercicio 10:

Distintos canales de televisión poseen programas que son conducidos por diversos conductores.

Dado el siguiente Diagrama del Modelo Entidad Relación, realizar el pasaje a Modelo Relacional indicando relaciones, atributos, claves primarias, y claves foráneas:



Guía 3: Álgebra Relacional.

Ejercicio 1:

Dado el siguiente esquema

Cientes: {CodigoCliente, Cuit, RazonSocial, Direccion }

Productos: {CodigoProducto, Descripcion, CodigoTipo }

TiposProducto: {CodigoTipo, Descripcion }

Compras: {CodigoCliente, CodigoProducto, Fecha, Cantidad }

En clientes se tiene una tupla por cliente, en producto una tupla por producto, en tipos de producto una tupla por tipo de producto. En Compras una tupla por compra, cada cliente hace una sola compra de cada producto por fecha.

- Obtener la dirección de todos los clientes.
- Obtener la razón social de todos los clientes que compraron productos de tipo "X".
- Obtener los Códigos de cliente que compraron el producto "Y".
- Obtener los clientes que compraron 2 unidades del producto "H" en la fecha "F".
- Obtener los clientes que compraron todos los productos.

Ejercicio 2:

Una empresa de entrega de comidas nos pide realizar algunas consultas sobre su base de datos. Teniendo en cuenta los siguientes esquemas de relaciones:

Cliente: { DNI, Nombres, Apellidos, Fecha_Registro, Fecha_Nacimiento }

Restaurante: { razón_social, CUIT }

Productos: { id_producto, nombre_comida, para_celíacos }

Productos_Restaurante: { id_producto, CUIT }

Pedido: { id_cliente, fecha_y_horario, id_producto, razon_social, precio }

realizar las siguientes consultas utilizando el álgebra relacional:

- Seleccionar los productos para celíacos.
- Proyectar el campo "nombre_comida" del esquema de relación "Productos".
- Unir los productos del restaurante con CUIT 30-99999999-7 con los del restaurante con CUIT 30-88888888-6.
- Obtener los productos ("id_producto") que no posee el restaurante con CUIT 30-99999999-7 en su menú.
- Realizar el producto cartesiano entre los esquemas de relación "Cliente" y "Restaurante".
- Renombrar el esquema de relación "Productos_Restaurante" por "P_R".
- Renombrar los atributos "id_producto", "CUIT" del esquema de relación "Productos_Restaurante" por "producto", "Código Tributario".
- Intersecar los productos del restaurante con CUIT 30-99999999-7 con los del restaurante con CUIT 30-88888888-6.
- Obtener los pedidos del restaurante con razón social "La milanese loca".
- Obtener los restaurantes que tienen todos los productos posibles en su menú.
- Asignar el resultado de la selección de los pedidos de "hoy" a una variable, y luego seleccionar dicha variable.

Ejercicio 3:

Una aplicación de transmisión y descarga de música, posee datos sobre temas (canciones), autores, y discográficas.

Teniendo en cuenta los siguientes esquemas de relaciones:

Temas: { nombre, nombre_autor, nombre_album, año, nombre_discografica, nombre_genero }

Autores: { nombre_autor, año_inicio, actualmente_en_actividad }

Álbumes: { nombre_album, nombre_autor }

Géneros: { nombre_genero }

Discográficas: { nombre_discografica, año_fundacion }

realizar las siguientes consultas utilizando el álgebra relacional:

- Seleccionar los autores (bandas o solistas) cuyo año de inicio sea mayor a 1990.
- Proyectar el campo "nombre_genero" del esquema de relación "Temas".
- Unir los "Temas" cuyo año es menor a 1970 con los "Temas" cuyo año es superior a 2000.
- Obtener las "Discográficas" que no poseen "Temas".
- Obtener el producto cartesiano entre "Autores" y "Géneros".
- Renombrar el esquema de relación "Autores" por "Bandas_Y_Solistas".
- Renombrar los atributos "nombre_autor" y "año_inicio" por "autor" e "inicio" respectivamente, en el esquema de relación "Autores".
- Obtener la intersección de autores, que tengan "Temas" entre 1975 y 1985 por un lado, y por otro lado entre 1990 y 1995.
- Mostrar el resultado de la junta natural entre "Temas" y "Autores".
- Obtener los autores que poseen "Temas" para todos los "Generos" posibles.
- Asignar el resultado de seleccionar "Autores" cuyo año_inicio es mayor a 1984 a una variable, y luego seleccionar dicha variable.

Ejercicio 4:

La asociación nacional de Tenis nos pide obtener información teórica sobre su modelo de datos. Tener en cuenta que un entrenador puede entrenar a varios tenistas, pero cada tenista tendrá sólo un entrenador. También tener en cuenta que un tenista, puede a su vez entrenar a otro tenista.

Teniendo en cuenta los siguientes esquemas de relaciones:

Tenistas: { DNI, nombre, apellido, ranking, DNI_entrenador }

Torneos: { nombre, ciudad }

Tenistas_Torneos: { DNI, nombre, ciudad }

Entrenadores: { DNI_entrenador, nombre, apellido }

realizar las siguientes consultas utilizando el álgebra relacional:

- Seleccionar todos los tenistas con ranking del 1 al 50 inclusive.
- Proyectar las ciudades de los torneos.
- Mostrar el resultado de unir los nombres y apellidos de los tenistas, con los nombres y apellidos de los entrenadores.
- Obtener los tenistas que no son entrenadores.
- Realizar el producto cartesiano entre "Tenistas" y "Torneos".
- Renombrar el esquema de relación "Tenistas_Torneos" por "Participaciones".
- Renombrar el atributo "nombre" del esquema de relación "Tenistas_Torneos" por "nombre_torneo".
- Realizar la intersección entre "Tenistas" y "Entrenadores".
- Realizar la junta natural entre "Torneos" y "Tenistas_Torneos".
- Obtener los "Tenistas" que participaron en todos los "Torneos".
- Asignar el resultado de la primera selección a una variable, y mostrar esa variable.

Ejercicio 5:

Una empresa que posee entre sus productos una red social, nos pide realizar algunas consultas.

Teniendo en cuenta los siguientes esquemas de relaciones:

Usuarios: { DNI_usuario, nombre, apellido, fecha_nacimiento }

Publicaciones: { DNI_usuario , fecha_y_hora_publicacion, texto, imagen }

Amigos: { DNI_usuario, DNI_usuario_amigo }

Comentarios: { DNI_usuario_publicacion, fecha_y_hora_publicacion,
DNI_usuario_comentario, comentario, fecha_y_hora_comentario }

realizar las siguientes consultas utilizando el álgebra relacional:

- Seleccionar los comentarios de la publicación del usuario con DNI 30333444 con fecha y hora "2024-09-01 12:10:00 000000".
- Proyectar los campos "DNI_usuario" y "fecha_y_hora_publicacion" de "Publicaciones".
- Unir los amigos del usuario con DNI "30333444" con los amigos del usuario con DNI "30888777".
- Realizar la diferencia de conjuntos entre los amigos del usuario con DNI "30333444" con los amigos del usuario con DNI "30888777".
- Realizar el producto cartesiano entre "Publicaciones" y "Comentarios".
- Renombrar el esquema de relación "Amigos" por "Relaciones".
- Renombrar el atributo "DNI_usuario_amigo" del esquema de relación "Amigos" por "DNI_amigo".
- Intersecar los amigos del usuario con DNI "30333444" con los amigos del usuario con DNI "30888777".
- Realizar la junta natural entre "Usuarios" y "Publicaciones".
- Obtener los usuarios que son amigos de todos los demás usuarios.
- Asignar el resultado de la anterior consulta a una variable, y seleccionarla.

Guía 4: Diseño Relacional.

Ejercicio 1:

Dados el esquema de relación $R = \{A, B, C, D\}$, y el conjunto de dependencias funcionales $F = \{A \rightarrow B, BC \rightarrow D, \}$, hallar la clausura de F (F^+).

Ejercicio 2:

Dados el esquema de relación $R = \{A, B, C, D, E\}$, y el conjunto de dependencias funcionales $F = \{A \rightarrow D, CD \rightarrow E, E \rightarrow B\}$, hallar la clausura de F (F^+).

Ejercicio 3:

Dados el esquema de relación $R = \{A, B, C, D, E\}$, y el conjunto de dependencias funcionales $F = \{B \rightarrow C, CD \rightarrow E, A \rightarrow D\}$, hallar la clausura de AB respecto a F .

Ejercicio 4:

Dados el esquema de relación $R = \{A, B, C, D, E\}$, y el conjunto de dependencias funcionales $F = \{A \rightarrow BC, B \rightarrow D, C \rightarrow AE\}$, hallar la clausura de C respecto a F .

Ejercicio 5:

Dados el esquema de relación $R = \{A, B, C, D\}$, y el conjunto de dependencias funcionales $F = \{A \rightarrow BC, C \rightarrow D\}$, hallar el cubrimiento minimal de F (FM).

Ejercicio 6:

Dados el esquema de relación $R = \{A, B, C, D, E\}$, y el conjunto de dependencias funcionales $F = \{AB \rightarrow C, B \rightarrow CD, D \rightarrow E\}$, hallar el cubrimiento minimal de F (FM).

Ejercicio 7:

Un cliente precisa materiales de construcción para distintas obras que está llevando a cabo. Los materiales son provistos por diferentes proveedores, y cada proveedor define el precio de cada material que vende. Dado el siguiente esquema de relación:

$U = \{ \text{cliente, proveedor, cuit, material, cantidad, precio} \}$

en donde se cumplen las siguientes dependencias funcionales:

- cliente, material \rightarrow cantidad
- proveedor \rightarrow cuit
- proveedor, material \rightarrow precio

normalizar a tercera forma normal.

Ejercicio 8:

Un taller de autos, mantiene una base de datos, con información sobre los vehículos que arregla. Dado el siguiente esquema de relación:

$U = \{ \text{patente, vehículo, fallo, dias_arreglo, precio_estimado_arreglo, repuestos} \}$

en donde se cumplen las siguientes dependencias funcionales:

- patente \rightarrow vehículo.
- vehículo, fallo \rightarrow dias_arreglo, precio_estimado_arreglo.
- fallo \rightarrow repuestos.

normalizar a tercera forma normal.

Ejercicio 9:

Una juguetería mantiene una base de datos con información de los artículos que posee, las facturas realizadas a clientes, y los proveedores a los cuales les compra los artículos.

Dado el siguiente esquema de relación:

$U = \{ \text{numero_factura, dni_cliente, articulo, cantidad, precio_articulo, importe_total_factura, proveedor} \}$

en donde se cumplen las siguientes dependencias funcionales:

- numero_factura \rightarrow dni_cliente, articulo, cantidad, precio_articulo, importe_total_factura.
- articulo, proveedor \rightarrow precio_articulo.

normalizar a Forma Normal de Boyce-Codd.

Ejercicio 10:

Una librería posee sucursales, en las cuales vende libros de distintas editoriales. Posee un sistema de gestión de base de datos, en el cual consulta y mantiene información relevante para su negocio. Dado el siguiente esquema de relación:

$U = \{ \text{sucursal, libro, editorial, autor, año, género, cantidad_en_stock_en_sucursal, tirada} \}$

en donde se cumplen las siguientes dependencias funcionales:

- libro \rightarrow autor, género.
- libro, editorial, año \rightarrow tirada.
- sucursal, libro \rightarrow cantidad_en_stock_en_sucursal.
- sucursal, libro, editorial \rightarrow autor, año, género, cantidad_en_stock_en_sucursal, tirada.

normalizar a Forma Normal de Boyce-Codd.

Guía 5: SQL.

Ejercicio 1:

Una cadena que vende hamburguesas en una ciudad, utiliza un sistema de gestión de base de datos, para obtener información relevante y tomar decisiones. Dados los siguientes esquemas de relaciones:

Sucursales: { nombre_sucursal, barrio }

Combos: { numero_combo, descripcion, precio }

Empleados: { dni, nombre, apellido, nombre_sucursal }

Barrios: { nombre }

realizar las siguientes consultas en SQL:

1. Crear todas las tablas, indicando claves primarias, claves foráneas, y restricciones.
2. Obtener la cantidad de empleados por sucursal, y ordenar de mayor a menor.
3. Obtener los barrios donde no hay sucursales de la cadena.
4. Obtener los barrios donde hay más sucursales, y mostrar dicha cantidad.
5. Obtener los combos que poseen igual precio, mostrando los campos "numero_combo", "descripción", y "precio".

Ejercicio 2:

Un club de barrio, posee socios que realizan diferentes deportes. Los socios pueden o no participar en diversos deportes dentro del club. Dados los siguientes esquemas de relaciones:

Socios: { numero_socio, nombre, apellido, fecha_nacimiento }

Deportes: { nombre, cantidad_jugadores_por_equipo }

Socios_Deportes: { numero_socio, nombre_deporte }

realizar las siguientes consultas en SQL:

1. Obtener el número de socio, el nombre y el apellido de aquellos socios que no realizan ningún deporte.
2. Obtener el número de socio, el nombre y el apellido de aquellos socios que realizan la máxima cantidad de deportes.
3. Obtener la cantidad de deportes que se realizan en el club.
4. Obtener los deportes que poseen la mayor cantidad de jugadores por equipo.
5. Obtener el número promedio de jugadores por equipo de los deportes.
6. Contar la cantidad de socios que realizan al menos dos deportes.

Ejercicio 3:

Un banco posee un sistema de gestión de base de datos, para poder guardar información y realizar diferentes consultas. Dados los siguientes esquemas de relaciones:

Clientes: { dni, nombre, apellido, fecha_nacimiento }

Préstamos: { id, monto, interes, dni_cliente, cuit_empresa }

Empresas: { cuit, razon_social, cantidad_empleados, patrimonio }

realizar las siguientes consultas en SQL:

1. Indicar cuales son las sentencias para la creación de las tablas. Asegurarse de que en la tabla "Préstamos", no puede ocurrir que ambos campos "dni_cliente" y "cuit_empresa" sean NULL. Definir clave primaria, claves foráneas, y restricciones.
2. Indicar la sentencia para crear un índice en la tabla Empresas, por cantidad_empleados.
3. Obtener los clientes con edad menor a 65 años.
4. Obtener los préstamos a clientes, con un monto mayor a \$50.000.000.
5. Obtener los préstamos a empresas, con un interés mayor al 60%.
6. Obtener las empresas (cuit y razón social) que tienen la máxima cantidad de empleados.

Ejercicio 4:

Una línea de colectivos utiliza un sistema de gestión de Base de Datos para poder gestionar información relacionada a sus colectivos, choferes, y los turnos. Dados los siguientes esquemas de relaciones:

Colectivos: { número_coche, marca_motor, marca_chassis, modelo }

Choferes: { dni, nombre, apellido, numero_coche }

Asignacion_turno: { dni, dia, turno } (turno; m = mañana, t = tarde, n = noche).

realizar las siguientes consultas en SQL:

1. Obtener los colectivos que tienen modelo anterior a 2014.
2. Obtener los colectivos ordenados por número de coche descendente.
3. Obtener los choferes sin colectivo asignado.
4. Obtener los choferes asignados al turno noche.
5. Obtener la cantidad de choferes agrupados por turno.

Ejercicio 5:

Una ferretería está interesada en tener un sistema de gestión de base de datos, para almacenar información relacionada a artículos que vende, fabricantes, clientes, y sus ventas. Dados los siguientes esquemas de relaciones:

Artículos: { codigo_articulo, nombre, descripción, codigo_fabricante, precio }

Fabricantes: { codigo_fabricante, razon_social }

Clientes: { dni_cliente, nombre, apellido }

Ventas: { fecha, dni_cliente, codigo_articulo, cantidad }

Se considera que cada cliente sólo realiza una venta por día. Realizar las siguientes consultas en SQL:

1. Obtener los datos de los clientes, ordenados por apellido.
2. Obtener los fabricantes tales que la ferretería no vendió ninguno de sus artículos.
3. Obtener el importe total de ventas agrupado por cliente, para el día de hoy.
4. Obtener los artículos más caros de cada fabricante.
5. Obtener los clientes que compraron los artículos con nombre "tornillos", y "clavos".
6. Obtener los clientes que compraron los artículos con nombre "tornillos", o "clavos".
7. Obtener un reporte de ventas, con los siguientes campos:
 - Fecha de venta.
 - Nombre y apellido del cliente.
 - Nombre del artículo.
 - Cantidad comprada.
 - Precio del artículo.
 - Razón social del fabricante del artículo.

Ejercicio 6:

Una Facultad de una Universidad, desea gestionar su base de datos, guardando información relevante sobre los alumnos, materias, y docentes. Dados los siguientes esquemas de relaciones:

Alumnos: { padron, nombre, apellido }

Docentes: { legajo, nombre, apellido, cargo }

Materias: { codigo, nombre }

Materias_Docentes: { código, legajo }

Notas: { padron, codigo, nota, año, cuatrimestre }

realizar las siguientes consultas en SQL:

1. Obtener todos los alumnos que aprobaron la materia "Base de Datos" (nota ≥ 4).
2. Obtener los alumnos con mejor promedio.
3. Obtener la cantidad de docentes por cargo.
4. Obtener los docentes que dictan más de una materia.
5. Obtener un listado de materias (código, nombre) y sus docentes (nombre, apellido, cargo), ordenado por código de materia.

6. Obtener los alumnos con más aplazos (nota < 4).
7. Obtener los alumnos que cursaron las materias “Base de Datos” y “Sistemas Operativos”.
8. Obtener los alumnos que cursaron las materias “Base de Datos” o “Sistemas Operativos”.

Ejercicio 7:

Un cine de barrio utiliza una base de datos, para guardar información relevante y realizar reportes sobre sus funciones, entradas vendidas, y los combos (pochoclos/gaseosas) que sus clientes consumen. Hay clientes que deciden contratar un plan adicional con descuentos para las entradas. Dados los siguientes esquemas de relaciones:

Películas: { nombre_pelicula, año, género }

Actores: { nombre, apellido, nombre_personaje, apellido_personaje, nombre_pelicula }

Funciones: { dia, hora, sala, nombre_pelicula, precio }

Cientes_vip: { dni_cliente, nombre, apellido, plan, descuento }

Combos: { numero_combo, descripcion, precio }

Ventas_Funciones: { dni_cliente, dia, hora, sala, nombre_pelicula, precio }

Ventas_Combos: { dni_cliente, numero_combo, precio }

realizar las siguientes consultas en SQL:

1. Obtener las ventas totales de funciones y combos (mostrar campos dni_cliente, y precios).
2. Obtener los clientes (dni_cliente, nombre, apellido) que más entradas de cine compraron.
3. Obtener los clientes (dni_cliente, nombre, apellido) que más combos compraron.
4. Obtener los clientes (dni_cliente, nombre, apellido) que compraron entradas de cine y combos.
5. Obtener los días y horarios de las funciones con menor precio.
6. Obtener un listado con la cantidad de películas por género, ordenada de manera descendente.
7. Obtener los actores que más participaron en películas.

Guía 6: Procesamiento y Optimización de Consultas.

Ejercicio 1:

Se tiene una tabla de clientes, una de Facturas y otra de ítems de Facturas. Es frecuente buscar a los clientes por Razón Social o Cuit para emitir facturas. También es común buscar facturas por fecha. La factura se debe poder imprimir. Definir las tablas con sus columnas y decir que índices se utilizarían, justificando. Escribir SQL- DDL

Ejercicio 2:

Si cada acceso a disco tarda 10uS. Se tiene un factor de Bloque de 10. ¿Cuánto se tardará en recuperar 1000 filas?

Ejercicio 3:

Se tiene un índice cluster con altura= 10. Un Factor de bloque=10. En una consulta de select con atributo=cte en el where, la variabilidad del atributo es 20. La cantidad de bloques es 100. Cada acceso a disco tarda 10uS ¿Cuánto debería tardar el select?

Ejercicio 4:

Buscar la mejor forma de ejecutar esta consulta

$\pi_{\text{Dirección}} \sigma_{\text{RazonSocial}='Esta S.A.'} (\text{Empresas} \oplus \text{Ventas} \text{ Ventas.cuit=Empresas.cuit})$

Ejercicio 5:

Dadas 2 tablas T1 y T2 con Factores de bloque B1=100 B2=80. Calcular el costo de una junta utilizando loops anidados por bloque.

Ejercicio 6:

Dadas 2 tablas T1 y T2 con Factores de bloque B1=50, B2=70, N1=2000, N2=3000, H1=5, H2=7. Calcular el costo de una junta utilizando loop + índice.

Guía 7: Concurrency, Transactions, Recovery.

Ejercicio 1:

Dado el siguiente caso de estudio:

El sistema de facturación de una empresa tiene una tabla de clientes, una tabla de productos, una tabla de facturas y tabla de ítems de facturas.

Se pide definir las operaciones SQL que contendrá la transacción necesaria para emitir una factura.

Ejercicio 2:

Definir la frecuencia de backup para la base de datos de una empresa. Justificar.

Ejercicio 3:

Describir las características de los algoritmos REDO y UNDO.

Ejercicio 4:

Definir los distintos usuarios y los permisos correspondientes, que podría tener una base de datos del sistema de alumnos de la facultad. Escribir las sentencias SQL correspondientes.

Ejercicio 5:

Se tienen 2 transacciones que se ejecutan en forma solapada. ¿En todos los solapamientos posibles el resultado debe ser el mismo? Justificar.

Guía 8: Bases de datos NoSQL (MongoDB).

Ejercicio 1:

En una heladería se tiene un catálogo con los diferentes tipos de helados, precios y tamaños de los mismos.

Se pide escribir una consulta, para obtener los distintos tipos de helados ordenados por precio.

Ejercicio 2:

Una Ferretería genera un catálogo de sus productos para generar pedidos. Cada producto tiene un código y una descripción. Algunos productos pueden llevar datos adicionales.

Se pide escribir las consultas para incorporar un pedido a la ferretería.

Ejercicio 3:

Una consultora de opinión hace una encuesta acerca de los gustos de los clientes de un supermercado. El cuestionario consta de 7 preguntas. Cinco de ellas se responden en escala de 1 a 5 y dos son de respuesta libre.

Se pide definir las consultas para incorporar cada encuesta a la base de datos.

Ejercicio 4:

Una empresa constructora necesita manejar las solicitudes de herramientas a los encargados de cada obra. Cada encargado pide una herramienta determinada desde una fecha por una cantidad de días, manejando el stock de las mismas.

Se pide

- A. Definir la consulta para incorporar las solicitudes.
- B. Definir la consulta para incorporar las devoluciones.
- C. Definir la consulta para consultar el stock de herramientas

Ejercicio 5:

Una empresa aérea desea registrar en el check in el equipaje de mano que llevan los pasajeros. Se pide almacenar el peso de cada uno de los bultos y la cantidad de los mismos, que lleva cada pasajero y emitir un número de despacho.

Se pide

- A. Definir la consulta para incorporar el conjunto de bultos de cada pasajero.
- B. Definir la consulta para consultar el peso total de cada vuelo

Ejercicio 6:

Una clínica desea administrar las solicitudes de turnos de sus pacientes para estudios médicos. Por cada paciente que desea realizar un estudio se debe ingresar el dni, tipo de estudio, fecha y hora deseada y el texto de la orden médica.

Se pide

- A. Definir la consulta para incorporar las solicitudes de turno en una base de datos.
- B. Definir la consulta para consultar los pacientes por tipo de estudio para una fecha dada.

Ejercicio 7:

Una Universidad desea administrar el uso de los laboratorios para las distintas materias que se dictan en ella. Cada docente que requiera el uso de un laboratorio, debe crear una solicitud indicando código de materia, tipo de laboratorio, fecha y hora en que desea utilizarlo (los turnos son de 1 hora cada uno).

Se pide

- A. Definir la consulta para incorporar las solicitudes de uso de laboratorio.
- B. Definir la consulta para consultar las asignaciones de materias en un laboratorio dado, durante un día determinado.