**Diseño de Base de Datos**

Práctica 3 - Álgebra Relacional

**NOTA: Dadas las siguientes relaciones, resolver utilizando Álgebra Relacional las consultas planteadas.**

**Ejercicio 1:**

**Cliente**(idCliente, nombre, apellido, DNI, telefono, direccion)

**Factura**(nroTicket, total, fecha, hora,idCliente(Fk))

**Detalle**(nroTicket, idProducto, cantidad, preciounitario)

**Producto**(idProducto, descripcion, precio, nombreP, stock)

1. Listar nombre, apellido, DNI, teléfono y dirección de clientes con DNI superior a 22222222

π nombre,apellido,DNI,teléfono,direccion(σ (DNI>”22222222”)(Cliente))

1. Listar nombre, apellido, DNI, teléfono y dirección de clientes con DNI superior a 22222222 y que tengan facturas cuyo total no supere los $100000.

π nombre,apellido,DNI,teléfono,direccion(σ (DNI>”22222222”)and(total<100000)(Cliente|X|facutra))

1. Listar nombre, apellido, DNI, teléfono y dirección de clientes que realizaron compras durante 2020.

π nombre,apellido,DNI,teléfono,direccion(σ (fecha>=”1/1/2020”)and(fecha<=”31/12/2020”)(Cliente|X|facutra))

1. Listar nombre, apellido, DNI, teléfono y dirección de clientes que no realizaron compras durante 2020.

Condición<= π idCliente,nombre, apellido, DNI, teléfono, dirección (σ(fecha>=”1/1/2020”) and

(fecha<=”31/12/2020”) (Cliente|X|facutra))

π nombre, apellido, DNI, teléfono, dirección (Cliente-Condicion)

1. Listar nombre, apellido, DNI, teléfono y dirección de clientes que solo tengan compras durante 2020.

CompraAntes <= π idCliente,nombre,apellido,DNI,teléfono,dirección(σ(fecha<=”1/1/2020”) (Cliente|X|facutra))

CompraDespues <= π idCliente, nombre, apellido, DNI, teléfono, dirección (σ(fecha>=”1/1/2020”) (Cliente|X|facutra))

Compra <= π idCliente, nombre, apellido, DNI, teléfono, dirección (CompraAntes U CompraDespues)

π nombre, apellido, DNI, teléfono, dirección (Cliente-Compra)

1. Listar nombre, descripción, precio y stock de productos no vendidos.

Pr <== π idProducto, descripcion, precio, nombreP, stock ((Producto|X|Detalle))

π descripcion, precio, nombreP, stock(Producto-Pr)

1. Listar  nombre, apellido, DNI, teléfono y dirección de clientes que no compraron el producto con nombre ‘ProductoX’ durante 2020.

Personas2020<==π idCliente, nombre,apellido,DNI,teléfono,direccion(σ(nombre=ProductoX)

and(fecha>=1/1/2020)and(fecha<=31/12/2020)(Cliente|X|Producto|X|Detalle|X|Factura))

πnombre,apellido,DNI,teléfono,dirección(Cliente-Personas2020)

1. Listar  nombre, apellido, DNI, teléfono y dirección de clientes que compraron el producto con nombre ‘Producto A’’ y no compraron el producto con nombre ‘Producto B’.

CompraronA <== πidCliente,nombre,apellido,DNI,teléfono,direccion(σ(nombreP= ProductoA)(Cliente|X|Producto|X|Detalle|X|Factura))

CompraronB <== πidCliente,nombre,apellido,DNI,teléfono,direccion(σ(nombreP= ProductoB)(Cliente|X|Producto|X|Detalle|X|Factura))

π nombre,apellido,DNI,teléfono,dirección(CompraronA-CompraronB)

1. Listar nroTicket, total, fecha, hora y DNI del cliente, de aquellas  facturas donde se haya comprado el producto ‘Producto C’.

π nroTicket, total, fecha, hora, DNI(σ(nombreP=ProductoC)(Cliente|X|Producto|X|Detalle|X|Factura))

1. Agregar un producto con id de producto 1000, descripción “mi producto”, precio $10000, nombreP “producto Z” y stock 1000. Se supone que el idProducto 1000 no existe.

**Producto <== Producto U {[** 1000,“mi producto”, $10000, “producto Z”,1000**]}**

**Ejercicio 2:**

**Banda(codigoB, nombreBanda, genero\_musical, año\_creacion)**

**Integrante (DNI, nombre, apellido,dirección,email, fecha\_nacimiento,codigoB(fk))**

**Escenario(nroEscenario, nombre \_ escenario, ubicación,cubierto, m2, descripción)**

**Recital(fecha,hora,nroEscenario, codigoB(fk) )**

1. Listar datos personales de integrantes con apellido ‘Garcia’ o fecha de nacimiento anterior a 2005 que toquen en bandas de rock and roll.

πDni,nombre,apellido,dirección,email,fecha\_nacimiento,codigoB(σ(apellido=Gracia)or(fecha\_nacimiento<=31/12/2004)or(genero\_musical=Rock and roll)(Banda|X| Integrante))

1. Listar nombre de escenario, ubicación y descripción de escenarios que no tuvieron recitales durante 2019.

Recitales2019 <== π nroEscenario, nombre,escenario,ubicación,cubierto,m2,descripción(

σ(Recital.fecha>=1/1/2019)and(Recital.fecha<=31/12/2019)(Escenario|X|Recital))

π nroEscenario, nombre,escenario,ubicación ,descripción(Escenario-Recitales2019)

1. Listar nombre de escenario, ubicación y descripción de escenarios que tuvieron recitales con género musical rock and roll o tuvieron recitales durante 2020.

Recitales2020oGenero <== σ(Recital.fecha>=1/1/2020)and(Recital.fecha<=31/12/2020)or

(Banda.genero\_musical=rock and roll)(Escenario|X|Recital|X|Banda))

π r.nroEscenario, r.nombre\_escenario,r.ubicación,r.descripción( p r (Recitales2020oGenero))

1. Listar nombre, género musical y año de creación de bandas que hayan realizado recitales en escenarios cubiertos durante 2019 .// cubierto es true, false según corresponda

Recitales2019 <== σ(Recital.fecha>=01/01/2019)and(Recital.fecha<=31/12/2019)and

(cubierto=True) (Escenario|X|Recital|X|Banda))

π r.nombreBanda, r.genero\_musical, r.año\_creacion(p r Recitales2019)

1. Listar DNI, nombre, apellido, dirección y email de integrantes nacidos entre 2000 y 2005 y que toquen en bandas con género pop que hayan tenido recitales durante 2020.

Recitales2020 <== σ(fecha\_nacimiento>=1/1/2000) and(fecha\_nacimiento<=31/12/2005)

and (genero\_musical = pop) and(fecha>=1/1/2020) and(fecha<=31/12/2020) (Banda|X|Integrante|X|Recital)

π r.DNI, r.nombre, r.apellido,r.direccion,r.emial(p r Recitales2019)

Alternativa:

IntegranteM ⇐ σ (i.fecha\_nacimiento >= 01/01/2000 and i.fecha\_nacimiento <= 31/12/2005) (ρi (integrante))

BandaPop ⇐ σ b.genero\_musical = “pop” and (r.fecha >= ‘01/01/2020 and r.fecha <= ‘31/12/2020) (ρr (recital) |x| ρb (banda))

π DNI, nombre, apellido, dirección, email (IntegranteM |x| BandaPop)

1. Listar DNI, nombre, apellido, email de integrantes que hayan tocado en el escenario con nombre ‘Gustavo Cerati’ y no hayan tocado en el escenario con nombre ‘Carlos Gardel’.

IntegrantesGustavo<==σ(nombre\_esenario = GustavoCerati) (Integrante|X|recital|X|banda|X|esenario)

IntegrantesCarlos<==σ(nombre\_esenario = CarlosGardel) (Integrante|X|recital|X|banda|X|esenario)

π DNI, nombre, apellido,direccion,emial(IntegrantesGustavo - IntegrantesCarlos)

1. Modificar el año de creación de la banda de nombre ‘Ratones Paranoicos’ a: 1983.

Ratones<== σ (nombreBanda = ‘Ratones Paranoicos’) (Banda)

δ Ratones.año\_creacion ⇐ Ratones.año\_creacion = 1983

1. Reportar nombre, género musical y año de creación de bandas que hayan realizado recitales durante 2019, y además hayan tocado durante 2020.

Tocaron2019 <== σ (fecha\_nacimiento>=1/1/2019) and (fecha\_nacimiento<=31/12/2019) (Banda|X|Recital)

Tocaron2020 <== σ (fecha\_nacimiento>=1/1/2020) and (fecha\_nacimiento<=31/12/2020) (Banda|X|Recital)

π nombre, genero\_musical,año(Tocaron2019 ∩ Tocaron2020)

1. Listar el cronograma de recitales del día 04/12/2019. Se deberá listar: nombre de la banda que ejecutará el recital, fecha, hora, y el nombre y ubicación del escenario correspondiente.

π nombreBanda,fecha,hora,nombre\_esenario,ubicación σ (fecha=04/12/2019)(Recital|X|Escenario|X|Banda)

**Ejercicio 3:**

**Agencia (RAZON\_SOCIAL, dirección, telef, e-mail)**

**Ciudad (CODIGOPOSTAL, nombreCiudad, añoCreación)**

**Cliente (DNI, nombre, apellido, teléfono, dirección)**

**Viaje(FECHA,HORA,DNI, cpOrigen (Fk), cpDestino (Fk), razon\_social(Fk), descripcion)//cpOrigen y cpDestino corresponden a la ciudades origen y destino del viaje**

1. Eliminar el cliente con DNI:25326992.

ClienteDNI <== σ DNI = 25326992(cliente)

ViajesCliente <== π FECHA,HORA,DNI,cpOrigen(Fk),cpDestino(Fk), razon\_social(Fk),

descripción(ClienteDNI|X|Viajes)

Viajes <== Viaje - ViajesCliente

Cliente <== Cliente – ClienteDNI

1. Listar datos personales de clientes que solo realizaron viajes locales. (En cada viaje realizado coincide la localidad origen con la destino, cpOrigen y cpDestino).

π **DNI, nombre, apellido, teléfono, dirección** σ (cpOrigen = cpDestino)(Viajes|X|Cliente)

1. Listar información de agencias que no tengan viajes para el cliente DNI:22222222 durante el primer semestre de 2020.

anterior <== σ DNI = 22222222 and fecha <= 01/01/2020 (Agencia|X|Viaje|X|Cliente))

después <== σ DNI = 22222222 and fecha >= 31/12/2020 (Agencia|X|Viaje|X|Cliente))

π RAZON\_SOCIAL, dirección, telef, e-mail(anterior U después)

Prguntar?

ClienteConViajes2020 <= π RAZON\_SOCIAL, dirección, teléf, e-mail (σ DNI = 22222222 and fecha >= 01/01/2020 and fecha <= 31/6/2020 (Agencia|X|Viaje|X|Cliente))

π RAZON\_SOCIAL, dirección, telef, e-mail(Agencia- ClienteConViajes2020)

1. Listar información de agencias que realizaron viajes durante 2019 y no realizaron viajes durante 2020.

Viajes2019 <== σ fecha >= 01/01/2019 and fecha <= 31/12/2019 (Agencia|X|Viaje))

Viajes2020 <== σ fecha >= 01/01/2019 and fecha <= 31/12/2019 (Agencia|X|Viaje))

π RAZON\_SOCIAL, dirección, telef, e-mail(Viajes2019 - Viajes2020)

1. Agregar una agencia de viajes con los datos que desee.

Agencia <== Agencia U {[2222,”Calle 1 altura 1032”,22222,”pepe@gmail.com”]}

1. Listar datos personales de clientes que viajaron con ciudad destino ‘Lincoln’ pero no realizaron viajes con origen ‘La Plata’.

cpLaplata <== π codigoposital (σ nombreCiudad = “La plata” (Ciudad))

cpLincoln <== π codigoposital (σ nombreCiudad = “Lincoln” (Ciudad))

ViajesLincoln <== σ CODIGOPOSTAL = cpLincoln (Viaje|X|Cliente)

ViajesLaplata <== σ CODIGOPOSTAL = cpLaplata (Viaje|X|Cliente)

πDNI, nombre, apellido, teléfono, dirección(ViajesLincoln - ViajesLaplata)

1. Listar nombre, apellido, dirección y teléfono de clientes que viajaron con todas las agencias.

πnombre, apellido, teléfono, dirección (Agencia|X|Viaje|X|Cliente) % π RAZON\_SOCIAL(Agencia)

1. Listar código postal, nombre Ciudad y año creación de ciudades que no recibieron viajes durante 2020.

Viajes2020 <== σ fecha >= 01/01/2019 and fecha <= 31/12/2019 (Viaje|X|Ciudad)

π CODIGOPOSTAL, nombreCiudad, añoCreación ((Viaje|X|Ciudad)-Viajes2020)

1. Reportar información de agencias que realizaron viajes durante 2019 o que tengan dirección igual a  ‘General Pinto’.

π **RAZON\_SOCIAL, dirección, telef, e-mail** (σ fecha >= 01/01/2019 and fecha <= 31/12/2019 or dirección = General Pinto (Viaje|X|Agencia))

1. Actualizar el teléfono del cliente con DNI: 2789655 a: 221-4400345.

ClienteDNI <== σ (DNI = 2789655 (cliente))

δ Cliente <== ClienteDNI.telef = 221-4400345

**Ejercicio 4:**

**Equipo(codigoE, nombreE, descripcionE)**

**Integrante (DNI, nombre, apellido,ciudad,email, telefono,codigoE(fk))**

**Laguna(nroLaguna, nombreL, ubicación,extension, descripción)**

**TorneoPesca(codTorneo, fecha,hora,nroLaguna(fk), descripcion)**

**Inscripcion(codTorneo,codigoE,asistio, gano)// asistio y gano son true o false según corresponda**

1. Listar DNI, nombre, apellido y email de integrantes que sean de la ciudad ‘La Plata’ y estén inscriptos en torneos que se disputaron durante 2019.

π **DNI,nombre,apellido,ciudad,email**(σ (ciudad=LaPlata and fecha>=01/01/2019 and fecha<=31/12/2019)(Integrante|X|Equipo|X|Inscripcion|X|TorneoPesca))

1. Reportar nombre y descripción de equipos que solo se hayan inscripto en torneos de 2019.

antes <== (σ (fecha<=01/01/2019) (Equipo|X|Integrante|X|TorneoPesca|X|Inscripcion)

despues <== (σ (fecha>=31/12/2019) (Equipo|X|Integrante|X|TorneoPesca|X|Inscripcion)

ambasFechas <== antes U despues

Torneo2019 <== TorneoPesca – ambasFechas

π Nombre,Descripcion(Equipo|X|Inscripcion|X|Torneo2019)

1. Listar nombre, ubicación, extensión y descripción de lagunas que hayan tenido torneos durante 2019 y no hayan tenido torneos durante 2020.

Torneos2019 <== (σ (fecha>=01/01/2019 and fecha<=31/12/2019)) (TorneoPesca)

Torneos2020 <== (σ (fecha>=01/01/2020 and fecha<=31/12/2020)) (TorneoPesca)

TorneosSin2020 <== Torneo2019 – Torneo2020

π Nombre,Ubicacion,Extencion,descripción(Lagunas|X|TorneosSin2020)

1. Listar para la laguna con nombre ‘laguna x’, nombre y descripción de equipos ganadores de torneos que se disputaron durante 2019 en la mencionada laguna.

Opción 100% posible:

Ganadores <== σ (gano=True) (Inscripcion)

LagunaX <== σ (NombreL = “lagunaX”) (Laguna)

Torneos2019 <== σ (fecha>=01/01/2019 and fecha<=31/12/2019) (TorneoPesca)

π Nombre,Descripcion(Equipo |X| Ganadores |X| LagunaX |X| Torneos2019)

Preguntar :

Torneos2019 <== σ (NombreL =” lagunaX” and fecha>=01/01/2019 and fecha<=31/12/2019 and Gano =True)

(TorneoPesca|X|Laguna|X|Equipo|X|Inscripcion)

π Nombre,Descripcion(Torneos2019)

1. Reportar nombre, y descripción de equipos que tengan inscripciones en todas las lagunas.

Lagunas <== σ NroLagunas(Laguna)

π Nombre,Descripcion((Equipo|X|Inscripcion|X|TorneoPesca|X|Laguna) % Lagunas)

1. Eliminar el equipo con código: 10000.

Equipo1000 <== σ CodigoE = 1000(Equipo)

InscripcionEquipo <== π codTorneo, codigoE, asistió, gano (Inscripcion |X| Equipo1000)

Inscripción <== Inscripcion – InscripcionEquipo

Equipo <== Equipo – Equipo1000

1. Listar nombreL, ubicación, extensión y descripción de lagunas que no tuvieron torneos.

LagunaTorneos <== π nroLaguna, nombreL, ubicación, extension, descripción (Laguna|X|TorneoPesca)

π nombreL, ubicación, extension, descripción (Laguna-LagunaTorneos)

1. Reportar nombre, y descripción de equipos que tengan inscripciones a torneos a disputarse durante 2019, pero no tienen inscripciones a torneos de 2020.

Inscripciones2019 <== σ (fecha>=01/01/2019 and fecha<=31/12/2019) (Equipo|X|Inscripcion|X|TorneoPesca)

Inscripciones2020 <== σ (fecha>=01/01/2020 and fecha<=31/12/2020) (Equipo|X|Inscripcion|X|TorneoPesca)

π nombreE, descripcionE (inscripciones2019 - inscripciones2020)

1. Listar DNI, nombre, apellido, ciudad y email de integrantes que asistieron o ganaron algún torneo que se disputó en la laguna con nombre: ‘Laguna Brava’

LagunaBrava ⇐ π nroLaguna (σ nombre=”Laguna Brava” (Laguna))

TorneoEnLagunaBrava ⇐ π codTorneo (TorneoPesca |x| LagunaBrava)

InscripcionLagunaBrava ⇐ π codigoE (TorneoEnLagunaBrava |x| Inscripcion) π DNI, nombre, apellido,ciudad,email (Integrante |x| Equipo |x| InscripcionLagunaBrava)

**Ejercicio 5**

**Club=(codigoClub,nombre,anioFundacion,codigoCiudad(FK))  
Ciudad=(codigoCiudad,nombre)  
Estadio=(codigoEstadiocodigoClub(FK),nombre,direccion)  
Jugador=(dni,nombre,apellido,edad,codigoCiudad(FK))  
ClubJugador(codigoClub, dni, desde, hasta)**

1. Reportar nombre y año de fundación de clubes de la ciudad de La Plata, además del nombre y dirección del estadio del mismo.

LaPlata <== π codigoCiudad (σ (nombre = “La Plata”)(Ciudad))

π Club.Nombre,Club.anioFundacion, Estadio.nombre, Estadio.Direccion σ (codigoCiudad= LaPlata)

(Club|X|Ciudad|X|Estadio)

1. Listar datos personales de jugadores actuales del club River Plate que hayan jugado en el club Boca Juniors.

Boca <== π codigoClub σ(nombre=”Boca”) (Club)

River <== π codigoClub σ(nombre=”River”)(Club)

JugadoresBoca <== π**dni, nombre, apellido, edad, codigoCiudad(FK)** σ(hasta>=31/10/2023)

(Boca|X|ClubJugador|X|Jugador)

JugadoresRiver <== π**dni, nombre, apellido, edad, codigoCiudad(FK)** σ(hasta>=31/10/2023)

(River|X|ClubJugador|X|Jugador)

π dni, nombre, apellido, edad (JugadoresBoca ∩ JugadoresRiver)

1. Listar información de todos los clubes donde se desempeñó el jugador: Marcelo Gallardo. Indicar nombre, año de fundación y localidad del club.

JugadorGalladrdo <== π dni σ (nombre = Marcelo and apellido = Gallardo) (Jugador)

CodigoDeLosClubs <== π codigoClub (JugadorGallardo|X|ClubJugador)

π nombre, anioFundacion, Ciudad.nombre (CodigoDeLosClubs |X| σ (Club.codigoCiudad = Ciudad.codigoCiudad) (Club|X|Ciudad)

JugadorGalladrdo <== π dni σ (nombre = Marcelo and apellido = Gallardo) (Jugador)

CodigoDeLosClubs <== π codigoClub (JugadorGallardo|X|ClubJugador)

π nombre, anioFundacion, Ciudad.nombre (CodigoDeLosClubs |X| Club |X| Ciudad)

1. Reportar dni, nombre y apellido de aquellos jugadores que no tengan más de 25 años y jueguen en algún club de la ciudad de Junín.

CodigoJunin <== π codigoCiudad σ (nombre = “Junin”) (Ciudad)

ClubesDeJunin <== π codigoClub (Club|X|CodigoJunin)

π dni, nombre, apellidó σ(edad<=25) (Jugador|X|ClubJugador|X|ClubesDeJunin)

1. Mostrar el nombre de los clubes que tengan jugadores de la ciudad de Chivilcoy mayores de 25 años.

codigoChivilcoy<==π codigoCiudad σ (nombre = “Chivilcoy”) (Ciudad)

Jugadores25años <== σ (edad>25) (codigoChivilcoy|X|Jugador)

c (Jugadores25años|X|ClubJugador|X|Club)

1. Reportar el nombre y apellido de aquellos jugadores que hayan jugado en todos los clubes.

π nombre, apellido, codigoClub(Jugador|X|ClubJugador) % π codigoClub (Club)

1. Listar nombre de los clubes que no hayan tenido ni tengan jugadores de la ciudad de La Plata.

CodigoLaPlata <== π codigoCiudad σ (nombre=LaPlata) (Ciudad

JugadoresDeLaPlata <== π dni (Jugador|X|CodigoLaPlata)

ClubesDeLaPlata <== π **codigoClub, nombre, anioFundacion, codigoCiudad (FK)** (JugadoresDeLaPlata|X|ClubJugador|X|Club)

π **nombre (Club-ClubesDeLaPlata)**

**opción2:**

ClubesDeLaPlata <== π **codigoClub, nombre, anioFundacion, codigoCiudad (FK)** σ (Ciudad.nombre=” LaPlata”) (Jugador|X|ClubJugador|X|Club|X|Ciudad)

π **nombre (Club-ClubesDeLaPlata)**

1. Mostrar dni, nombre y apellido de aquellos jugadores que jugaron o juegan en el club: Club Atlético Rosario Central.

π **jugador.nombre, jugador.apellido, jugador.dni** σ (Club.nombre=” Club Atletico Rosario Central”) (Jugador|X|ClubJugador|X|Club)

1. Eliminar al jugador cuyo dni es: 24242

JugadorDNI <== π dni, nombre, apellido, edad, codigoCiudad(FK) σ (dni = 24242424 ) (Jugado

Jugador <== Jugador-JugadorDNI

ClubDNI <== π codigoClub, dni, desde, hasta σ (dni = 24242424) (ClubJugador)

ClubJugador <== ClubJugador-ClubDNI

**Ejercicio 6:**

**Proyecto(codProyecto, nombrP,descripcion,fechaInicioP,fechaFinP,fechaFinEstimada,DNIResponsable(fk), equipoBackend(fk), equipoFrontend(fk)) //DNIResponsable corresponde a un empleado, equipoBackend y equipoFrontend corresponden a un equipo**

**Equipo (codEquipo, nombreE, descripcionTecnologias, DNILider(fk)) //DNILider corresponde a un empleado**

**Empleado (DNI, nombre, apellido, telefono, direccion, fechaIngreso)**

**Empleado\_Equipo (codEquipo, DNI, fechaInicio, fechaFin, descripcionRol)**

1. Listar nombre, descripción, fecha de inicio y fecha de fin de proyectos ya finalizados que no fueron terminados antes de la fecha de fin estimada.

π **nombre, descripcion, fechainicioP, fechafinP** σ(fechaFinP>fechaFinEstimada) (Proyecto)

1. Listar DNI, nombre, apellido, teléfono, dirección y fecha de ingreso de empleados que no hayan sido responsables de proyectos.

Responsables <== π **DNI, nombre, apellido, telefono, direccion, fechaIngreso** σ (empleado.dni = proyecto.dni) (Empleado X Proyecto)

π **DNI, nombre, apellido, telefono, direccion, fechaIngreso** (Empleados-Responsables)

1. Listar DNI, nombre, apellido, teléfono y dirección de todos los empleados que trabajan en el proyecto con nombre ‘Proyecto X’. No es necesario informar responsable y líderes.

EquipoBackend <== π equipoBackend(fk) σ (nombre = “Proyecto X”) (Proyecto)

EquipoFronted <== π equipoFronted(fk) σ (nombre = “Proyecto X”) (Proyecto)

EmpleadosBack <== π DNI, nombre, apellido, telefono, direccion, fechaIngreso (EquipoBackend |X| Empleado\_Equipo) |X| Empleado)

EmpleadosFronted <== π DNI, nombre, apellido, telefono, direccion, fechaIngreso (EquipoFronted |X| Empleado\_Equipo |X| Empleado)

π DNI, nombre, apellido, telefono, direccion, fechaIngreso (EmpleadosBack U EmpleadosFrontend)

1. Listar nombre de equipo y datos personales de líderes de equipos que no tengan empleados asignados y trabajen con tecnología ‘Java’.

EquiposConEmpleados <== π codEquipo, nombreE, descripcionTecnologias, DNILider(fk) (Empleado |X| Empleado\_Equipo)

EquiposSinEmpleados <== Equipo – EquiposConEmpleados

π nombreE, DniLider, nombre,apellido σ (descripcionTecnologica = “Java”)and(empleado.DNI = EquipoSinEmpleado.dniLider) (Empleado x EquiposSinEmpleados)

1. Modificar nombre, apellido y dirección del empleado con DNI: 40568965 con los datos que desee.

δ nombre <== “Franco” σ (Dni = 40568965) (Empleado)

δ apellido <== “Cirielli” σ (Dni = 40568965) (Empleado)

δ direccion <== “calle 1” σ (Dni = 40568965) (Empleado)

1. Listar DNI, nombre, apellido, teléfono y dirección de empleados que son responsables de proyectos pero no han sido líderes de equipo.

Lideres <== σ (Empleado.dni = Equipo.dniLider) (Empleado X Equipo)

Responsables <== σ (empleado.dni = proyecto.dni) (Empleado X Proyecto)

π DNI, nombre, apellido, teléfono y dirección (Responsables-Lideres)

1. Listar nombre de equipo y descripción de tecnologías de equipos que hayan sido asignados como equipos frontend y backend.

EquipoBackend <== π equipoBackend(fk) σ (nombre = “Proyecto X”) (Proyecto)

EquipoFronted <== π equipoFronted(fk) σ (nombre = “Proyecto X”) (Proyecto)

π nombreE, descripcionTeconologias (EquipoBack ∩ EquipoFrontend)

1. Listar nombre, descripción, fecha de inicio, nombre y apellido de responsables de proyectos a finalizar durante 2019.

π **DNI, nombre, apellido, telefono, direccion, fechaIngreso** σ (empleado.dni = proyecto.dni) and fechaFinEstimada<= 01/01/2019 and fechaFinEstimada>= 31/12/2019 (Empleado X Proyecto)

1. Listar nombre de equipo, descripción de tecnología y la información personal del líder, de equipos que no estén asignados a ningún proyecto aún.

EquipoBackend <== π equipoBackend(fk) σ (nombre = “Proyecto X”) (Proyecto)

EquipoFrontend <== π equipoFronted(fk) σ (nombre = “Proyecto X”) (Proyecto)

NoAsignado <== Equipo – EquipoBackend – EquipoFrontend

π codEquipo, descripcionTeconologia, DNI, nombre, apellido, telefono, direccion, fechaIngreso σ (empleado.dni = NoAsigando.dniLider) (Empleado x NoAsignado)

**Ejercicio 7**

**Vehiculo = (patente, modelo, marca, peso, km)**

**Camion = (patente, largo, max\_toneladas, cant\_ruedas, tiene\_acoplado)**

**Auto = (patente, es\_electrico, tipo\_motor)**

**Service = (fecha, patente, km\_service, observaciones, monto)**

**Parte = (cod\_parte, nombre, precio\_parte)**

**Service\_Parte = (fecha, patente, cod\_parte, precio)**

1. Listar todos los datos de aquellos camiones que tengan entre 8 y 12 ruedas, y que hayan realizado algún service antes de los 10000 km.

π patente,modelo, marca, peso, km, largo, max\_toneladas, cant\_ruedas, tiene\_acoplado σ (cant\_ruedas=8) or(cant\_ruedas=12) and(km\_service<10000) (Vehiculo|X|Camion |X| Service)

1. Listar los autos que hayan realizado el service “cambio de aceite” antes de los 13.000 km o hayan realizado el service “inspección general” que incluya la parte “filtro de combustible”.

InspeccionGeneral <== σ (observaciones = “inspección general”) and (nombre = filtro de combustible)

(Auto|X|Service|X|Service\_patente|X|Parte)

CambioAceite <== σ (observaciones = “cambio de aceite”) and(km\_service<13000)

(Auto|X|Service|X|Service\_patente)

π patente, es\_electrico, tipo\_motor (InspeccionGeneral U CambioAceite)

1. Dar de baja todos los camiones con más de 350.000 km.

Patentes350KM <== π patente σ (km>=350000) (Vehículo|X|Camion)

Camion <== (Camion – (Patentes350KM |X| Camion))

Vehiculo <== (Vehículo – (Patentes350KM |X| Vehiculo))

Service <== (Service – (Patentes350KM |X| Service))

Service\_Parte <== (Service\_Parte – (Patentes350KM |X| Service\_Parte))

1. Listar el nombre y precio de aquellas partes que figuren en todos los service realizados durante 2019.

Service2019 <== π patente, fecha σ (fecha>=01/01/2019) and(fecha<=31/12/2019) (Service |X| Service\_parte |X| Parte)

π nombre, precio, fecha, precio\_parte σ (Service\_parte |X| Parte) % π cod\_parte Service2019

1. Listar todos los autos cuyo tipo de motor sea eléctrico. Mostrar información de patente, modelo, marca y peso.

π patente, modelo, marca, peso σ (es\_electrico = true)(Vehiculo |X| Auto)

1. Dar de alta una parte, cuyo nombre sea “Aleron” y precio $3400.

Parte <= Parte U [{1321,”Aleron”,3400}]

1. Dar de baja todos los services que se realizaron al auto con patente ‘AAA564’.

ServicePatente <== σ (patente = AAA564) (Service)

Service <== (Service – ServicePatente)

Service\_Parte <== (Service\_Parte – (σ (patente = AAA564) (Service\_Parte)))

1. Modificar el precio de las partes incrementando un 15 % dicho valor.

δ precio\_parte <== precio\_parte \* (0.15)

1. Listar todos los vehículos que hayan tenido services durante el 2019.

π patente, modelo, marca, peso, km σ (fecha>=01/01/2019) and (fecha<=31/12/2019) (Vehiculo|X|Service)

Ejercicio 8

Modelo físico

**Box = (nroBox,m2, ubicación, capacidad, ocupacion) //ocupación es un numérico indicando cantidad de mascotas en el box actualmente, capacidad es una descripción.**

**Mascota = (codMascota,nombre, edad, raza, peso, telefonoContacto)**

**Veterinario = (matricula, CUIT, nombYAp, direccion, telefono)**

**Supervision = (codMascota,nroBox, fechaEntra, fechaSale?, matricula(fk), descripcionEstadia) //fechaSale tiene valor null si la mascota está actualmente en el box**

1. Modificar nombre y apellido al veterinario con matricula: ‘MP 10000’, deberá llamarse: ‘Pablo Lopez’.

δ nombYAp<== “Pablo Lopez” σ (matricula = ‘MP 10000’) (Veterinario)

1. Listar nombre, edad, raza, peso de mascotas que tengan supervisiones con el veterinario con matricula : ‘MP 1000’ y con el veterinario con matricula: ‘MN 4545’.

VeterinarioMP <== σ (matricula = ‘MP 10000’) (Veterinario)

VeterinarioMN <== σ (matricula = ‘MN 4545’) (Veterinario)

MascotasMP <== (VeterianrioMP|X|Supervision|X|Mascota)

MascotasMN <== (VeterianrioMN|X|Supervision|X|Mascota)

π nombre, edad, raza, peso (MascotasMP ∩ MascotasMN)

1. Listar nroBox, m2, ubicación, capacidad y nombre de mascota para supervisiones con fecha de entrada (fechaEntra) durante 2020.

π NroBox, Ubicación, m2, Capacidad, nombre σ (fechaEntra>=01/01/2020 and fechaEntra<=31/12/2020) (Supervision |X| Mascota |X| Box)

Ejercicio 9

**Modelo Físico**

**Barberia = (codBarberia, razon\_social, direccion, telefono)**

**Cliente = (nroCliente,DNI, nombYAp, direccionC, fechaNacimiento, celular)**

**Barbero = (codEmpleado,DNIB, nombYApB, direccionB, telefonoContacto, mail)**

**Atencion = (codEmpleado,Fecha,hora,codBarberia(fk), nroCliente(fk),descTratamiento, valor)**

1. Listar DNIB, nombYApB, direccionB, telefonoContacto y mail de barberos que tengan atenciones con valor superior a 5000.

π DNIB, nombYApB, direccionB,telefonoContacto,mail σ (valor>5000) (Barbero |X| Atencion)

1. Listar DNI, nombYAp, direccionC, fechaNacimiento y celular de clientes que tengan atenciones en la barbería con razón social: ‘Corta barba’ y también se hayan atendido en la barbería con razón social: ‘Barberia Barbara’.

CortaBarba <== π DNI, nombYAp, direccionC, fechaNacimiento, celular σ (razón\_social = “Corta barba”) (Atencion | X | Cliente | X | Barberia)

BarberiaBarbara <== π DNI, nombYAp, direccionC, fechaNacimiento, celular σ (razón\_social = “Barberia Barbara”) (Atencion | X | Cliente | X | Barberia)

π DNI, nombYAp, direccionC, fechaNacimiento, celular (CortaBarba ∩ BarberiaBarbara)

1. Eliminar el cliente con DNI: 22222222.

codCliente <== π nroCliente σ (DNI = 22222222) (Cliente)

Atencion <== Atencion – (Atencion |X| codCliente)

Cliente <== Cliente – (Cliente |X| codCliente)

Ejercicio 10

**Modelo Físico**

**Club(IdClub,nombreClub,ciudad)**

**Complejo(IdComplejo,nombreComplejo, IdClub(fk))**

**Cancha(IdCancha,nombreCancha,IdComplejo(fk))**

**Entrenador(IdEntrenador, nombreEntrenador,fechaNacimiento, direccion)**

**Entrenamiento(IdEntrenamiento, fecha, IdEntrenador(fk), IdCancha(fk))**

1. Listar nombre , fecha de nacimiento y dirección de entrenadores que hayan entrenado en la cancha “Cancha 1” y en la Cancha “Cancha 2”.

Cancha1 <== π σ (nombreCancha = “Cancha 1”) (Entrenador |X| Entrenamiento |X| Cancha)

Cancha2 <== σ (nombreCancha = “Cancha 2”)) (Entrenador |X| Entrenamiento |X| Cancha)

π Nombre,fechaNacimiento,dirección (Cancha1 ∩ Cancha2)

1. Listar todos los clubes en los que entrena el entrenador “Marcos Perez”. Informar nombre del club y ciudad.

MarcoPerez <== π idEntrenador σ (nombreEntrenador = “Marcos Perez”) (Entrenador)

π nombreClub,ciudad (MarcoPerez |X| Entrenamiento |X| Cancha |X| Complejo |X| Club)

1. Eliminar los entrenamientos del entrenador ‘Juan Perez’.

JuanPerez <== σ (nombreEntrenador = “Juan Perez”) (Entrenador)

Entrenamiento <== Entrenamiento – (JuanPerez |X| Entrenamiento)