

**CLIENTE:** NIF, nombre, dirección, ciudad, tfno

PK (NIF)

**COCHE:** matricula, marca, modelo, color, precio,NIF

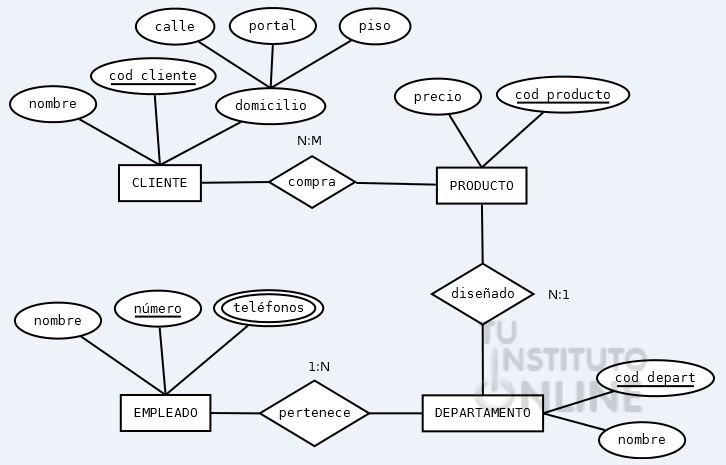
PK: (matricula)

FK: NIF → CLIENTE (NIF)

**REVISION:** codigo, filtro, aceite, frenos, matricula

PK: (codigo)

FK: matricula → COCHE (matricula)



**CLIENTE:** cod\_cliente, nombre, calle, portal, piso

PK: cod\_ cliente

**PRODUCTO:** cod\_producto, precio

PK: cod\_producto

**DEPARTAMENTO**: cod\_depart, nombre

PK: cod\_depart

**EMPLEADOS:** **numero**, nombres, telefonos

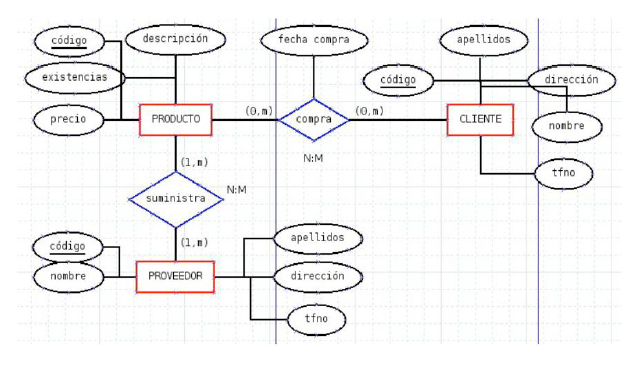
PK: numeros

**COMPRA:** cod\_cliente, cod\_procucto

PK: (cod\_cliente, cod\_procucto)

FK: cod\_cliente -> CLIENTE (cod\_cliente)

FK: cod\_producto -> PRODUCTO (cod\_producto)



**PROVEEDOR:** codigo\_proveedor, nombre, apellidos, dirección, telefono

PK: codigo\_ proveedor

**PRODUCTO:** codigo\_producto, descripción, existencias, precio

PK: codigo\_producto

**CLIENTE:** cod\_cliente, apellidos, direccion, nombre, telefono

PK: cod:cliente

**COMPRA:** cod\_cliente, codigo\_producto, fecha\_compra

PK: (cod\_cliente,codigo\_producto)

FK: cod\_cliente → CLIENTE (cod\_cliente)

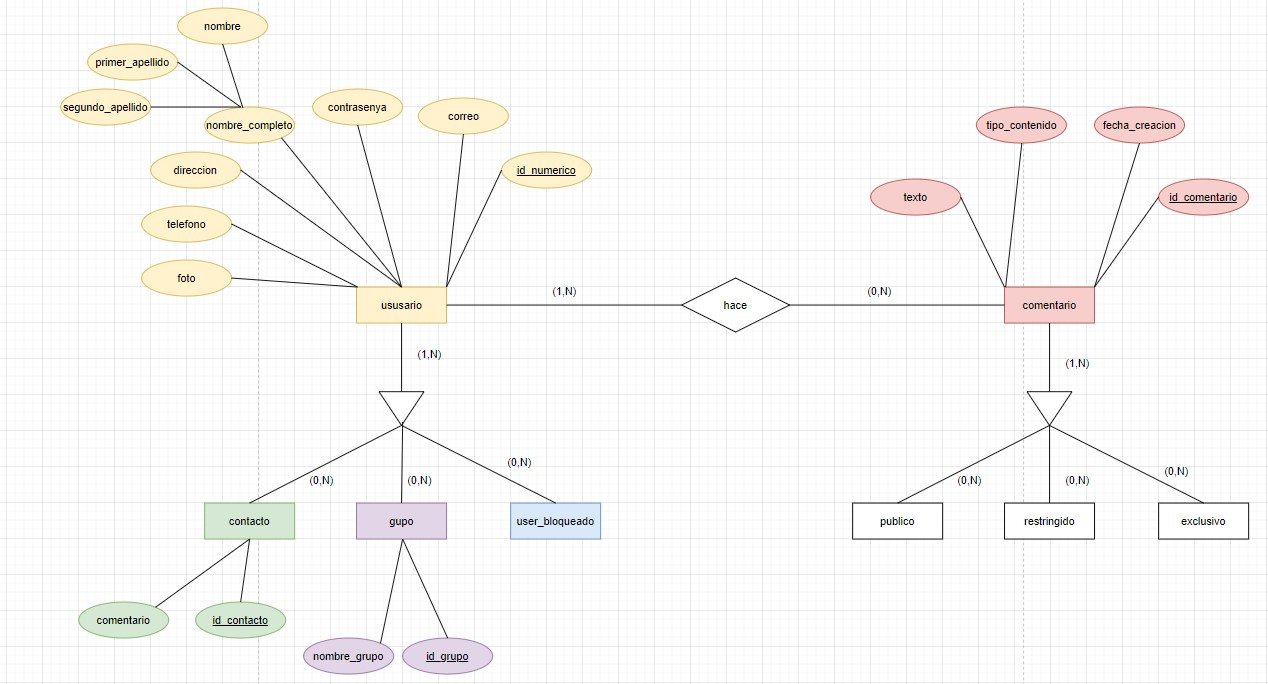
FK: codigo\_producto →PRODUCTO (codigo\_producto)

**SUMINISTRA:** codigo\_proveedor, codigo\_producto

PK: (codigo\_ proveedor,codigo\_producto)

FK: codigo\_ proveedor → PROVEEDOR (cod\_cliente)

FK: codigo\_producto →PRODUCTO (codigo\_producto)



**USUARIO:** id\_numerico, correo,contrasenya, nombre, primer\_apellido, segundo\_apellido, direccion, telefono, foto

PK: id\_numerico

**CONTACTO:** id\_numerico, comentario

PK: id\_numerico

FK: id\_numerico → USUARIO(id\_numerico)

**GRUPO:** id\_numerico, nombre\_grupo

PK: id\_grupo

FK: id\_numerico → USUARIO(id\_numerico)

**USER\_BLOQUEADO:** id\_numerico

PK: id\_numerico

FK: id\_numerico → USUARIO(id\_numerico)

ASERCION 1: CONTACTO[id\_numerico] **∩ GRUPO [id\_numerico] = Ø**

**ASERCION 2: CONTACTO[id\_numerico] ∩ USER\_BLOQUEADO [id\_numerico] = Ø**

**ASERCION 3: GRUPO[id\_numerico] ∩ USER\_BLOQUEADO [id\_numerico] = Ø**

ASERCION 4: USUARIO[id\_numerico] **⊆ CONTACTO[id\_numerico] ∪ GRUPO[id\_numerico] ∪ USER\_BLOQUEADO [id\_numerico]**

**COMENTARIO:** id\_comentario, fecha\_creacion, tipo\_comentario, leido

PK: id\_comentario

**PUBLICO**: id\_comentario

PK: id\_comentario

FK: id\_comentario → COMENTARIO (id\_comentario)

**RESTRINGIDO**: id\_comentario

PK: id\_comentario

FK: id\_comentario → COMENTARIO (id\_comentario)

**EXCLUSIVO**: id\_comentario

PK: id\_comentario

FK: id\_comentario → COMENTARIO (id\_comentario)

ASERCION 1: PUBLICO[id\_comentario]**∩ RESTRINGIDO [id\_comentario] = Ø**

**ASERCION 2: PUBLICO[id\_comentario]∩ EXCLUSIVO [id\_comentario] = Ø**

ASERCION 3: **RESTRINGIDO** [id\_comentario]**∩ EXCLUSIVO [id\_comentario] = Ø**

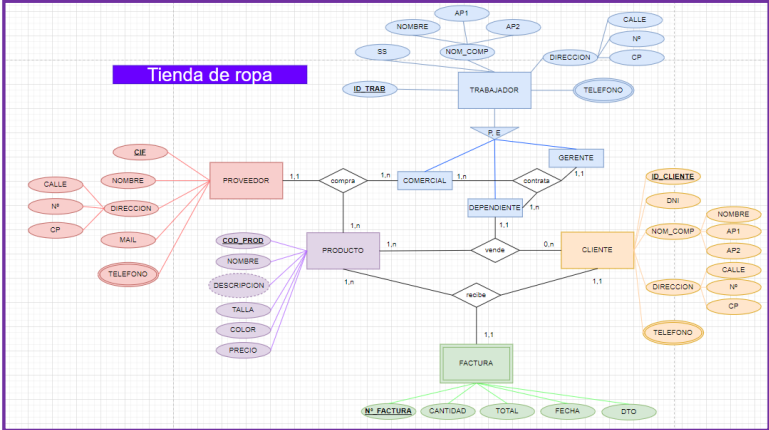
**ASERCIÓN 4: COMENTARIO [id\_comentario] ⊆ PUBLICO[id\_comentario] ∪ RESTRINGIDO [id\_comentario] ∪ EXCLUSIVO [id\_comentario]**

**HACE:** id\_numerico, id\_comentario

PK: (id\_numerico, id\_comentario)

FK: id\_numerico → USUARIO (id\_numerico)

FK: id\_comentario → COMENTARIO (id\_comentario)



**TRABAJADOR**: id\_trab, {telefono}, ss, nombre, ap1, ap2, calle, nº, cp

PK: id\_trab

VNN: ss, nombre, ap1, ap2, calle, nº, cp

**GRERNTE**: id\_gerente

PK: id\_gerente

FK: id\_gerente -> TRABAJADOR (id\_trabajador)

**DEPENDIENTE**: id\_dependiente

PK: id\_dependiente

FK: id\_dependiente -> TRABAJADOR (id\_trabajador)

**COMERCIAL**: id\_comercial

PK: id\_comercial

FK: id\_comercial -> TRABAJADOR (id\_trabajador)

**CLIENTE**: id\_cliente, {telefono}, dni, nombre,ap1,ap2, calle, nº, cp

PK: id\_cliente

VNN: dni, nombre,ap1,ap2, calle, nº, cp

**FACTURA**: nº\_factura, cantidad, total, fecha,dto

PK: nº\_factura

VNN: cantidad, total, fecha,dto

**PRODUCTO**: cod\_producto, nombre, descripcion, talla, color, precio

PK: cod\_producto

VNN: nombre, talla, color, precio

**PROVEEDOR**: CIF, {telefono}, nombre, calle, nº, cp, mail

PK: CIF

VNN: nombre, calle, nº, cp, mail

**CONTRATA:** id\_comercial, id\_dependiente, id\_gerente

PK: (id\_comercial, id\_dependiente)

FK: COMERCIAL (id\_comercial) -> TRABAJADOR (id\_trabajador)

FK: DEPENDIENTE (id\_dependientel) -> TRABAJADOR (id\_trabajador)

FK: GERENTE (id\_gerente) -> TRABAJADOR (id\_trabajador)

VNN: id\_gerente

ASERCION1: COMERCIAL (id\_comercial)**⊆ CONTRATA (id\_comercial)**

**ASERCION2: DEPENDIENTE (id\_dependiente)⊆ CONTRATA (id\_dependiente)**

**ASERCION3: GERENTE (id\_gerente)⊆ CONTRATA (id\_gerente)**

**COMPRA:** id\_comercial, cod\_producto, cif

PK: (id\_comercial, cod\_producto)

FK: id\_comercial -> COMERCIAL (id\_comercial)

FK: cif -> PROVEEDOR (cif)

FK: cod\_producto -> PRODUCTO (cod\_producto)

VNN: cif

ASERCION1: PROVEEDOR (cif) **⊆ COMPRA (cif)**

**ASERCION2: COMERCIAL (id\_comercial) ⊆ COMPRA (id\_comercial)**

**ASERCION3: PRODUCTO (cod\_producto) ⊆ COMPRA (cod\_producto)**

**VENDE:** id\_cliete, cod\_producto, id\_dependiente

PK: (id\_cliete, cod\_producto)

FK: id\_ cliete -> CLIENTE (id\_cliente)

FK: cod\_producto -> PRODUCTO (cod\_producto)

FK: id\_dependiente -> DEPENDIENTE (id\_dependiente)

VNN: id\_dependiente

ASERCION1: DEPENDIENTE (id\_dependiente) **⊆ VENDE (id\_dependiente)**

**ASERCION2: PRODUCTO (cod\_producto) ⊆ VENDE (cod\_producto)**

**RECIBE:** id\_cliente , cod\_producto, nºfactura

PK: (id\_cliete, cod\_producto)

FK: id\_ cliete -> CLIENTE (id\_cliente)

FK: cod\_producto -> PRODUCTO (cod\_producto)

FK: nºfactura -> FACTURA (nºfactura)

VNN: nºfactura

**EJERCICIO 1**. Normaliza hasta 1FN el esquema relacional siguiente:

X= a0 + a1 + {x + y + z}

1FN = No

X = a0 + a1

PK: a0

X1= a0 + x + y + z

PK: (a0 + x)

FK: a0 -> X(a0)

**EJERCICIO 2.** Normaliza hasta 2FN el esquema relacional siguiente:

A = a0 + {a1 + a2} + c + d con DF={a1 → a2, c → d}

¿1FN? = NO

A = a0 + c + d

PK: a0

A1= a0 + a1 + a2

PK: a0 + a1

FK: a0 -> A(a0)

Tabla A

¿1FN? = Si

¿2FN? = Si automatica, porque la clave primaria no es compuesta

Tabla A1

¿1FN? = Si

¿2FN? = No porque a1 -> a2 y a1 forma parte de la clave primaria

A1 = a0 + a1

PK: (a0 + a1)

FK: a0 -> A12(a0)

A12 = a1 +a2

PK: a1

Resultado final

A = a0 + c + d

A12 = a1 +a2

A1 = a0 + a1

FK: a0 -> A12(a0)

FK: a0 -> A(a0)

EJERCICIO 3

PIEZA = pieza(letras+modelo) + descripción + precioVenta + principal(letras+modelo) + estantería + cantidad

FK: principal → PIEZA(pieza)

FK: estantería → ESTANTERÍA( estantería)

ALMACÉN = número + descripción + dirección

ESTANTERÍA = estantería + almacén

FK: almacén → ALMACÉN(número)

VNN(almacén)

Dependencias funcionales

DF = { letras → descripción }

ALMACEN: ¿1FN? = Si

¿2FN? =Si

¿3FN? =Si

ESTANTERIA: ¿1FN? = Si

¿2FN? =Si

¿3FN? =Si

PIEZA: ¿1FN? = No, porque los atributos no son atomicos

PIEZA = (letras+modelo) + descripción + precioVenta + principal\_letras+principal\_modelo + estantería + cantidad

FK: (principal\_letras + principal\_modelo) → PIEZA(letras + modelo)

FK: estantería → ESTANTERÍA( estantería)

¿2FN? = No. Por letras -> descripcion y letras es PK

PIEZA = (letras+modelo) + precioVenta + principal + estantería + cantidad

FK: letras ->PIEZA1 (letras)

FK: (principal\_letras + principal\_modelo) → PIEZA(letras + modelo)

FK: estantería → ESTANTERÍA( estantería)

PIEZA1= letras + descripcion

¿3FN? = SI

RESULTADO FINAL

PIEZA = (letras+modelo) + precioVenta + principal + estantería + cantidad

FK: letras ->PIEZA1 (letras)

FK: (principal\_letras + principal\_modelo) → PIEZA(letras + modelo)

FK: estantería → ESTANTERÍA( estantería)

PIEZA1= letras + descripcion

ALMACÉN = número + descripción + dirección

ESTANTERÍA = estantería + almacén

FK: almacén → ALMACÉN(número)

VNN(almacén)

EJERCICIO 4

CATADORES = dniCatador + nombre + experiencia + dnipareja

VINOS = codBarras + añada + precio + bodega

COMPOSICIÓN = codBarras + tipoUva + porcentaje

CATAS = dniCatador + CodBarras + notaAromas + notaSabor + tipoVino

DF: CodBarras -> tipoVino

CATADOR: ¿1FN? = Si

¿2FN? =Si

¿3FN? =Si

VINOS: ¿1FN? = Si

¿2FN? =Si

¿3FN? =Si

COMPOSICION: ¿1FN? = Si

¿2FN? =Si

¿3FN? =Si

CARTAS: ¿1FN? = Si

¿2FN? =No por la DF de codBarras -> tipoVino porque CodBarras es PK

CATAS = dniCatador + CodBarras + notaAromas + notaSabor

FK= CodBarras -> CARTAS1

CARTAS1 = CodBarras + tipoVino

¿3FN? =Si

UNIFICAR TABLAS

VINOS = codBarras + añada + precio + bodega

CARTAS1 = CodBarras + tipoVino

VINOS = codBarras + añada + precio + bodega + tipovino

RESULTADO FINAL

CATAS = dniCatador + CodBarras + notaAromas + notaSabor

FK= CodBarras -> VINO (CodBarras)

CATADORES = dniCatador + nombre + experiencia + dnipareja

VINOS = codBarras + añada + precio + bodega + tipovino

COMPOSICIÓN = codBarras + tipoUva + porcentaje

EJERCICIO 5. Dado el siguiente esquema relacional de un IES:

ALUMNO = Expediente + Nombre + Ape1 + Ape2 + DNI + NSS + calle + municipio + provincia + CP

PK: Expediente

AK: DNI, NSS

DF: CP -> provincia + municipio

ALUMNO: ¿1FN? = Si

¿2FN? =Si

¿3FN? =No. Porqeu la DF de CP ->provicia + municipio depende de CP no es pk

ALUMNO = Expediente + Nombre + Ape1 + Ape2 + DNI + NSS + calle + CP

PK: Expediente

AK: DNI, NSS

FK: CP -> CODIGOPOSTAL (CP)

CODIGOPOSTAL = CP + provincia, municipio

EJERCICIO 6. Normaliza hasta BCFN las siguientes tablas:

a) ALUMNOS = expediente + nombre + apellido + { teléfono} + dirección.

b) ALUMNOS = expediente + nombre + apellido + {asignatura + nota + curso + aula}

y DF = { asignatura → curso, curso → aula }

c) LIBROS = ISBN + título + editorial + país

d) PERSONAS = DNI + Nombre + Apellidos + Dirección + CP + Población + Provincia

e) EMPLEADOS = DNI + NSS + Nombre + apellidos + departamento + puesto + salario

Clave alternativa: NSS

y DF = { puesto → salario }

f) FIGURAS = nombre + color + tamaño

Clave Alternativa: color + tamaño

y DF = { tamaño → nombre }

ALUMNOS:

¿1FN? = No. Porque hay atributos multievaluados

ALUMNOS = expediente + nombre + apellido + dirección.

ALUMNOS1= expediente + telefono

FK: expediente -> ALUMNOS (expediente)

¿2FN? =Si

¿3FN? =Si

ALUMNOS:

¿1FN? = No. Porque hay atributos compuestos

ALUMNOS = expediente + nombre + apellido

ALUMNOS1= expediente + asignatura + nota + curso + aula

FK: expediente -> ALUMNOS (expediente)

¿2FN? = NO. Por las DF de { asignatura → curso, curso → aula }

ALUMNOS = expediente + nombre + apellido

ALUMNOS1= expediente + telefono

FK: expediente -> ALUMNOS (expediente)

ALUMNOS1\_1= expediente + asignatura + nota

FK: expediente -> ALUMNOS (expediente)

FK: asignatura -> ALUMNOS2\_2 (asignatura)

ALUMNOS1\_2 = asignatura + curso + aula

¿3FN? = ALUMNOS: Si

ALUMNOS1\_1:Si

ALUMNOS1\_2:No Por la DF de {curso → aula }

ALUMNOS1\_2 = asignatura + curso

FK: curso -> ALUMNO1\_2\_1(curso)

ALUMNOS 1\_2\_1= curso + aula

RESULTADO ALUMNOS

ALUMNOS = expediente + nombre + apellido

ALUMNOS1= expediente + telefono

FK: expediente -> ALUMNOS (expediente)

ALUMNOS1\_1= expediente + asignatura + nota

FK: expediente -> ALUMNOS (expediente)

FK: asignatura -> ALUMNOS2\_2 (asignatura)

ALUMNOS1\_2 = asignatura + curso

FK: curso -> ALUMNO1\_2\_1(curso)

ALUMNOS 1\_2\_1= curso + aula

LIBROS: ¿1FN? = Si

¿2FN? = Si

¿3FN? = Si

PERSONAS: ¿1FN? = Si

¿2FN? = Si

¿3FN? = Si

EMPLEADOS: ¿1FN? = Si

¿2FN? = Si

¿3FN? = No por la DF de puesto -> salario porque puesto

EMPLEADOS = DNI + NSS + Nombre + apellidos + departamento + puesto

FK: puesto -> EMPLEADOS1(puesto)

EMPLEADOS\_1= puesto + salario

FIGURAS: ¿1FN? = Si

¿2FN? = Si

¿3FN? = Si

EJERCICIO 7. En el esquema relacional de una empresa, tenemos la relación VENTAS que recoge información sobre cada venta que realiza la empresa.

VENTAS = fecha + vendedor + cliente + producto + NombreProducto + costeProducto + margenProducto + cantidadVendida + precioUnidad + importeBase + tipoIVA + %IVA + importeIVA + importeTotal + %comision + importeComision + ganancia

DF:

vendedor → %comisión

producto → precio + coste + margen,

tipoIVA → %IVA,

costeProducto + margenProducto → precioUnidad,

FIGURAS: ¿1FN? = Si

¿2FN? = NO por la DF de vendedor-> comision y producto -> precio...

VENTAS = fecha + vendedor + cliente + producto + NombreProducto + cantidadVendida + importeBase + tipoIVA + %IVA + importeIVA + importeTotal + importeComision + ganancia

FK: vendedor -> VENTAS1(vendedor)

FK: producto -> VENTAS2(producto)

VENTAS 1= vendedor + %comision

VENTAS 2= producto + precio

FK: precio-> VENTA2\_1(precio)

¿3FN? = NO por la DF de tipoIVA-> %IVA y costeProducto + margenProducto → precioUnidad

VENTAS = fecha + vendedor + cliente + producto + NombreProducto + cantidadVendida + importeBase + tipoIVA + importeIVA + importeTotal + importeComision + ganancia

FK: tipoIVA -> VENTAS3 (tipoIVA)

VENTAS3= tipoIVA + %IVA

VENTAS 2\_1= precio + coste + margen,

RESULTADO FINAL

VENTAS = fecha + vendedor + cliente + producto + NombreProducto + cantidadVendida + importeBase + tipoIVA + importeIVA + importeTotal + importeComision + ganancia

FK: tipoIVA -> VENTAS3 (tipoIVA)

FK: vendedor -> VENTAS1(vendedor)

FK: producto -> VENTAS2(producto)

VENTAS 1= vendedor + %comision

VENTAS 2= producto + precio

FK: precio-> VENTA2\_1(precio)

VENTAS3= tipoIVA + %IVA

VENTAS 2\_1= precio + coste + margen,

EJERCICIO 8. Dado el siguiente esquema relacional de los alquileres de películas de un videoclub: AT= Película + Director + { Ejemplar + estado + {Fecha + Socio} }

DF={ Ejemplar→ Película, Película → Director }

AT: ¿1FN? = NO porque tiene atributos compuestos

AT= Película + Ejemplar + Director

AT1: ¿1FN? = NO porque tiene atributos compuestos

AT1= Ejemplar + estado + Fecha

FK:ejemplar -> AT (ejemplar)

AT1\_2= Fecha + Socio

FK: fecha -> AT1 (fecha)

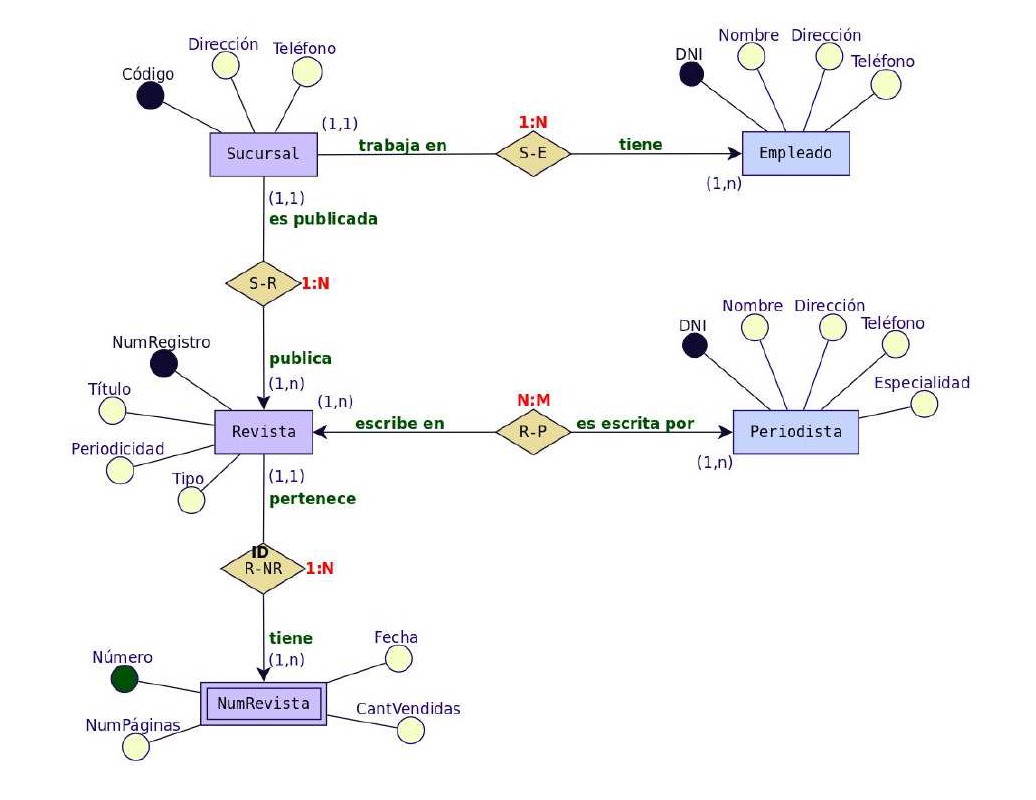
¿2FN? = Si

¿3FN? = Si

¿2FN? = NO

AT2= Película + Ejemplar

¿3FN? = Si

PASO A TABLAS

**EMPLEADO**: DNI + nombre+ direccion + telefono +codigo

PK: DNI

FK: codigo -> SUCURSAL (codigo)

VNN: nombre+ direccion + telefono

**SUCURSAL**: codigo + direccion + telefono

PK: codigo

VNN: direccion + telefono

**REVISTA**: NumRegistro + titulo+ periodicidad + tipo + codigo

PK: NumRegistro

FK: codigo -> SUCURASL (codigo)

VNN: titulo+ periodicidad + tipo

**PERIODISTA**: DNI + nombre + direccion + telefono + especialidad

PK: DNI

VNN: nombre + direccion + telefono + especialidad

**NUMREVISTA**: numregistro + numero + numpaginas + fecha + cantvendidas

PK: numero + numregistro

FK: numregistro -> REVISTA (numregistro)

VNN: numpaginas + fecha + cantvendidas

**ESCRITA**: numregistro + dni

PK: (numregistro + dni)

FK: numregistro -> REVISTA (numregistro)

FK: dni -> PERIODISTA (dni)

PERSONAS: ¿1FN? = Si

¿2FN? = Si

¿3FN? = Si

Información Inicial:

**Libros:** ISBN, Título, Autor, Año de publicación, Biblioteca\_ID, Ubicación

**Autores:** Autor\_ID, Nombre, País

**Bibliotecas:** Biblioteca\_ID, Nombre, Ubicación, Teléfono

Las dependencias funcionales iniciales son:

ISBN -> Título, Autor, Año de publicación

Autor -> Nombre, País

Biblioteca\_ID -> Nombre, Ubicación, Teléfono

NORMALIZACION:

LIBROS: ¿1FN? = Si

¿2FN? = NO por las DF de ISBN -> Título, Autor, Año de publicación

**Libros:** ISBN, Título, Autor, Año de publicación

**Libros1** : Biblioteca\_ID, Ubicación

¿3FN? = Si

AUTORES: ¿1FN? = Si

¿2FN? = Si

¿3FN? = Si

BIBLIOTECA: ¿1FN? = Si

¿2FN? = Si

¿3FN? = Si

EJERCICIO 11. Una cooperativa de productores de cítricos, necesita crear una BD que registre las curas que aplican sus socios a sus cosechas, incluyendo las retiradas de plaguicidas de sus almacenes y el transporte de los mismos. Los socios son a la vez propietarios de sus propias parcelas y trabajadores de la cooperativa. Para ello, utilizan este esquema:

PROPIETARIOS = dni\_socio + cod\_parcela + nombre\_socio + {tel\_socio}

RETIRADAS = cod\_parcela + fecha + tipo\_cítrico + cod\_plaguicida + metros\_parcela + Kg\_plaguicida + plaga + matrícula + marca + modelo + color + PMA + consumo\_medio + carnet\_minimo + año\_compra + conductor + distancia\_parcela

PROPIETARIOS: ¿1FN? = No porqeu hay atributos multievaluado

PROPIETARIOS = dni\_socio + cod\_parcela + nombre\_socio

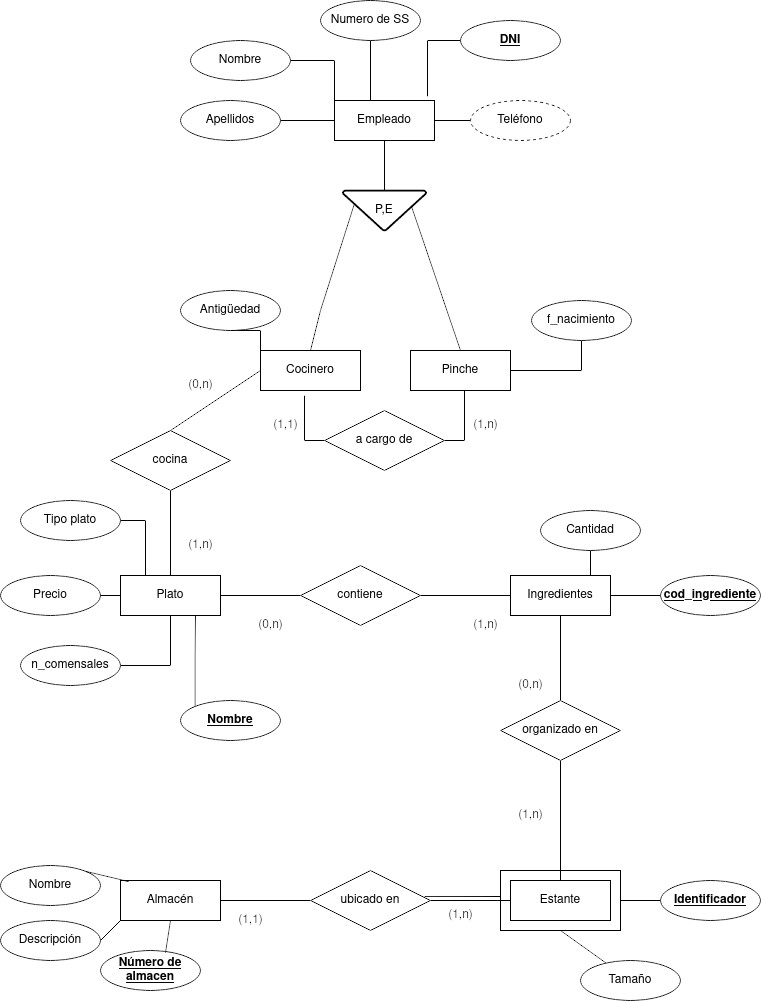
PROPIETARIOS1= dni\_socio + cod\_parcela + tel\_socio

FK: dni\_socio -> PROPIETARIOS(dni\_socio)

FK: cod\_parcela -> PROPIETARIOS(cod\_parcela)

¿2FN? = Si

¿3FN? = Si



PASO A TABLAS:

EMPLEADO: DNI, telefono, numero\_ss, nombre, apellidos

PK: DNI

C.Alt = numero\_ss

VNN = numero\_ss

VNN = nombre,

VNN = apellidos

COCINERO: DNI, antigüedad

PK: DNI

VNN = antigüedad

PINCHE : DNI, f\_nacimiento

PK: DNI

ASSERCION1: COCINERO [dni] **∩ PINCHE [dni] = Ø**

**EXCLUSIVA = DISJUNTA**

SOLAPADA = NO DISJUNTA

SIGNOS **∪∩⊆**

**∪ = Union: Todos los elementos que pertenecen al menos uno de los dos conjuntos.**

**∩ = Interseccion: Todos los elementos pertenecen a ambos conjuntos.**

**⊆ = Subconjunto**

**NO DISJUNTA Y PARCIAL**

**No se ponen aserciones**

**DISJUNTA Y PARCIAL**

**Se ponen aserciones entre los entidades de la herencia**

**ASERCION 1: A1[a] ∩ A2 [a] = Ø**

**ASERCION 2: A1[a] ∩ A3 [a] = Ø**

**ASERCION 3: A2[a] ∩ A3 [a] = Ø**

**NO DISJUNTA Y TOTAL**

**Se pone una asecion de la principal que es un subconjunto de las entidades de la herencia.**

**ASERCION1: A[a] ⊆ A1[a] ∪ A2[a] ∪A3[a]**

**DISJUNTA Y TOTAL**

**Se ponen aserciones entre los entidades de la herencia y que es un subconjunto de las entidad principal**

**ASERCION 1: A1[a] ∩ A2 [a] = Ø**

**ASERCION 2: A1[a] ∩ A3 [a] = Ø**

**ASERCION 3: A2[a] ∩ A3 [a] = Ø**

**ASERCION 4: A[a] ⊆ A1[a] ∪ A2[a] ∪A3[a]**