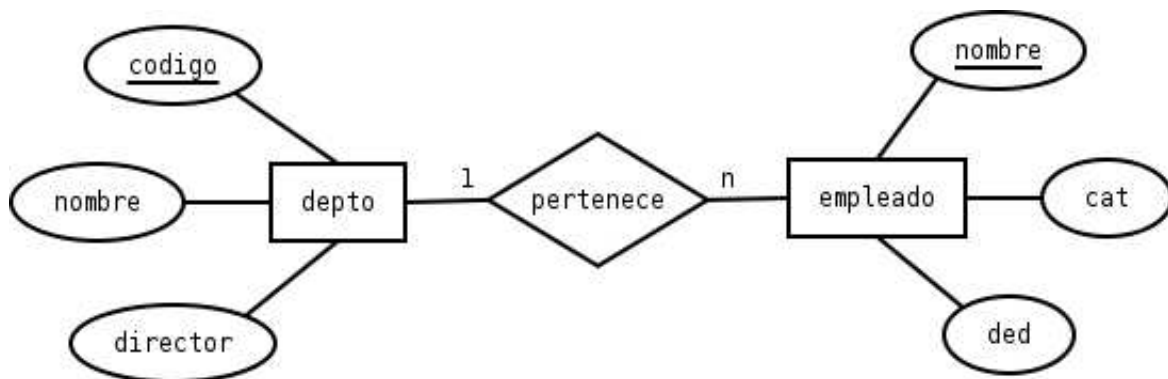


Ejercicios de bases de datos. Soluciones

Introducción a la Informática
Licenciado en ADE
Fac. de Administración y Dirección de Empresas. UPV

Ejemplo 1¹

Dado un código de departamento, conocer su nombre, director y los empleados de ese departamento con su nombre, categoría y dedicación.



DEPTO

(codigo, nombre, director)

CP codigo

EMPLEADO

(nombre, cat, ded, codigo)

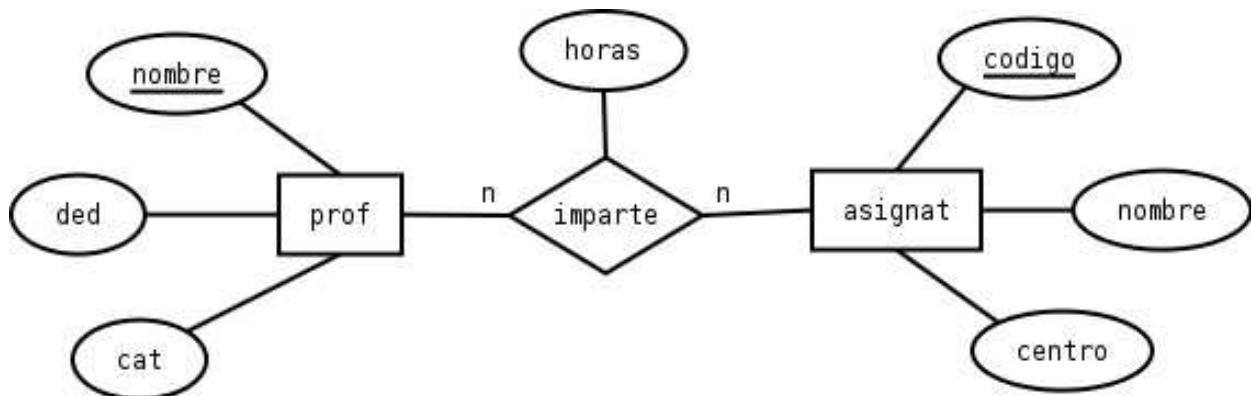
CP nombre

CAj codigo -> DEPTO

¹ En este ejemplo se muestra cómo se modelan las relaciones 1:n (uno a muchos)

Ejemplo 2²

Dado un código de profesor, determinar su nombre, dedicación y categoría, así como el conjunto de asignaturas que imparte con el código de esa asignatura, su nombre, el centro en el que se imparte y el nº de horas.



PROF

(nombre, ded, cat)

CP nombre

ASIGNAT

(codigo, nombre, centro)

CP codigo

IMPARTE

(nombre, codigo, horas)

CP nombre, codigo

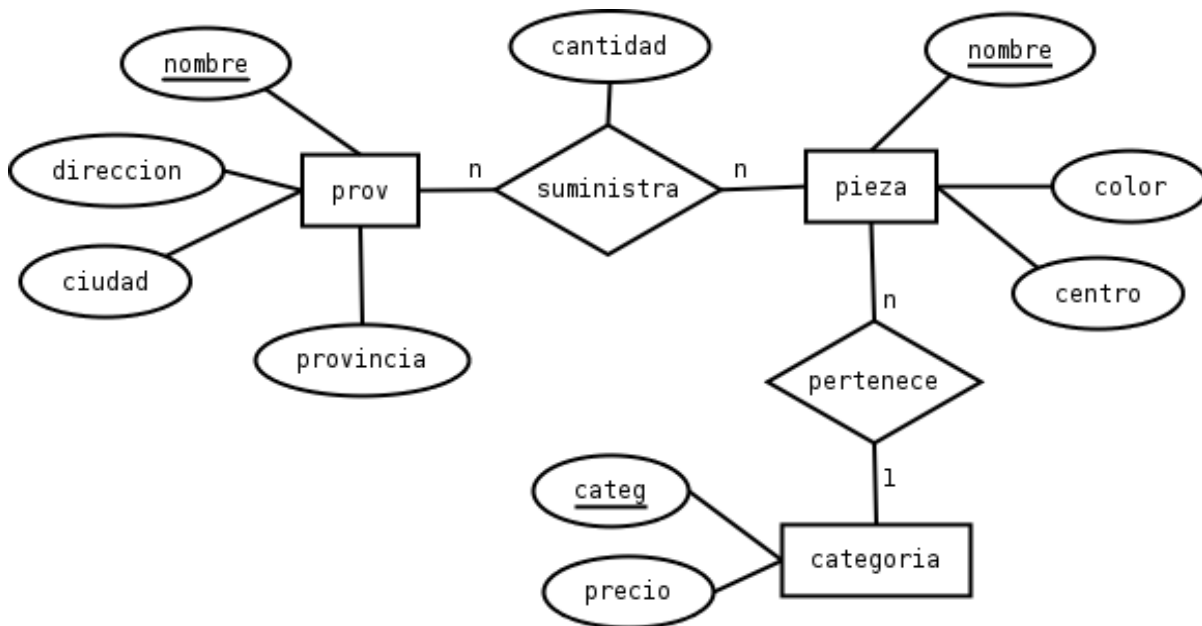
CAj nombre -> PROF

CAj codigo -> ASIGNAT

2 En este ejemplo se muestran las relaciones n:m (muchos a muchos) y la presencia de atributos en las relaciones

Ejemplo 3³

Dado un proveedor, queremos conocer su nombre, dirección, ciudad y provincia, así como las piezas que suministra, conociéndose color, categoría, precio y cantidad (donde para cada categoría hay un único precio)



PROV

(nombre, direccion, ciudad, provincia)

CP nombre

PIEZA

(nombre, color, centro, categ)

CP nombre

CAj categ -> CATEGORIA

CATEGORIA

(categ, precio)

CP categ

SUMINISTRA

(nombre, nombre, cantidad)

CP nombre, nombre

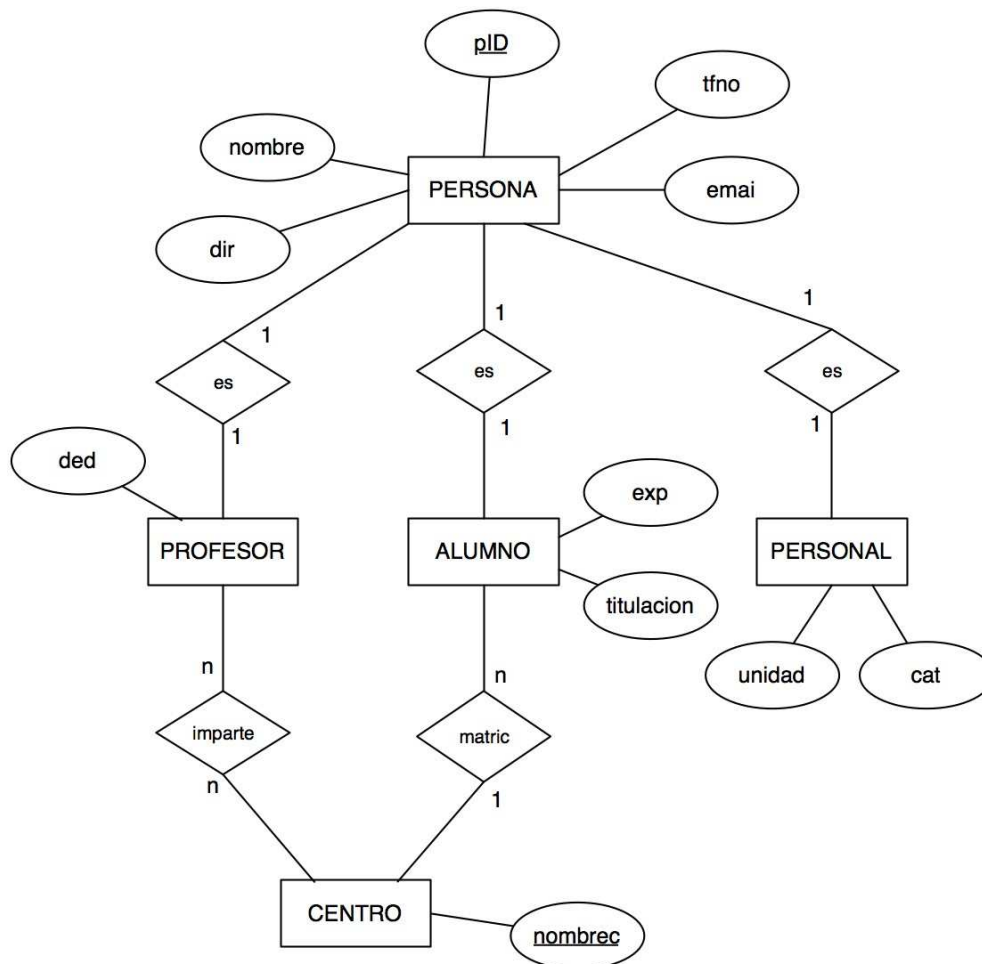
CAj nombre -> PROV

CAj nombre -> PIEZA

3 Este ejemplo muestra la dependencia de todos los atributos de una entidad de la clave

Ejemplo 4⁴

La BD de una universidad almacena información sobre todos sus miembros. De cualquier persona se almacena su nombre, dirección, teléfono y email. Los profesores además tienen el departamento al que pertenecen y su dedicación, además de los centros en los que imparten clase. Los alumnos están matriculados en un único centro y tienen un número de expediente y la titulación en la que están matriculados. Del personal se almacena la unidad administrativa a la que pertenecen y su categoría profesional



PERSONA
(pID, nombre, dir, tfno, email)
CP: pID

PROFESOR
(pID, ded)
CP: pID
CAj: pID → PERSONA

ALUMNO
(pID, exp, tit, nombrec)
CP: pID
CAj: pID → PERSONA
CAj: nombrec → CENTRO

PERSONAL
(pID, cat, unidad)
CP: pID
CAj: pID → PERSONA

CENTRO
(nombrec)
CP: nombrec

IMPARTE
(pID, nombrec)
CP: pID, nombrec
CAj: pID → PROFESOR
CAj: nombrec → CENTRO

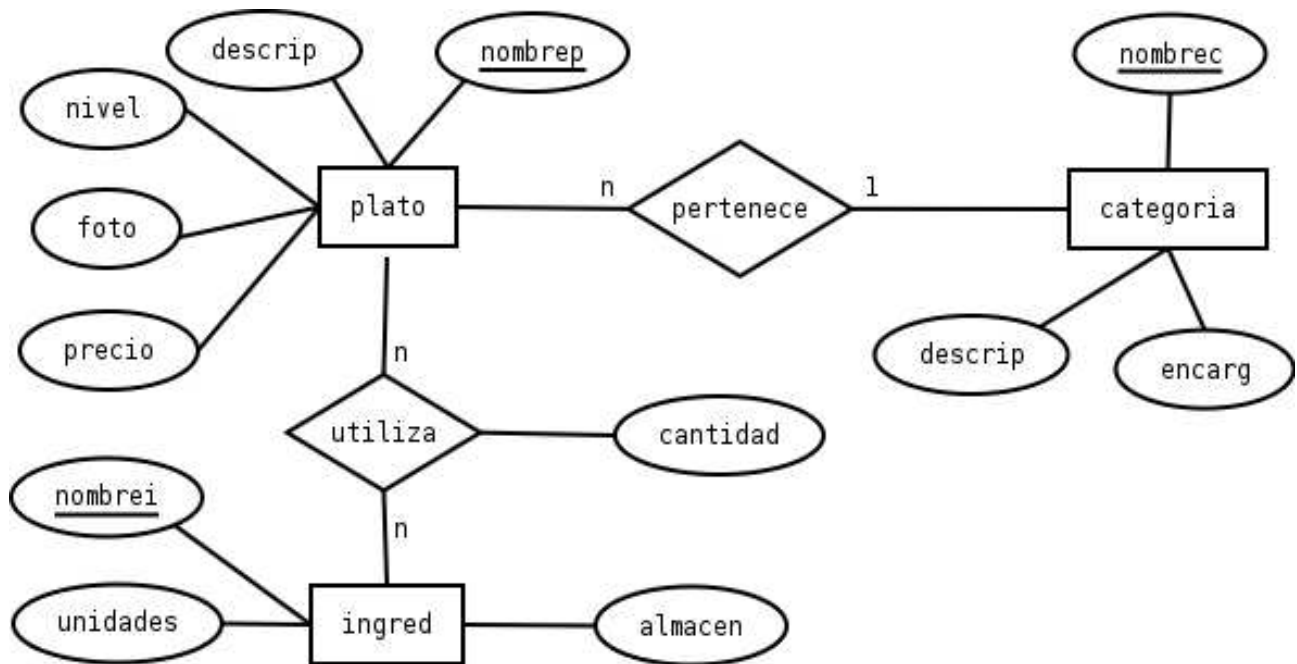
⁴ Este ejemplo muestra cómo se modelan las relaciones 1:1 (uno a uno)

Ejercicio 1

Se desea construir una base de datos que almacene la carta de un restaurante.

Para cada plato, se desea obtener su nombre, descripción, nivel de dificultad (de elaboración), una foto y el precio final para el cliente. Cada plato pertenece a una categoría.

Las categorías se caracterizan por su nombre, una breve descripción y el nombre del encargado. Además de los platos, se desea conocer las recetas para su realización, con la lista de ingredientes necesarios, aportando la cantidad requerida, las unidades de medida (gramos, litros, etc. . .) y cantidad actual en el almacén.



PLATO

(nombrep, descrip, nivel, foto, precio, nombrec)

CP nombrep

CAj nombrec -> CATEGORIA

CATEGORIA

(nombrec, descrip, encarg)

CP nombrec

INGRED

(nombrei, unidades, almacen)

CP nombrei

UTILIZA

(nombrep, nombrei, cantidad)

CP nombrep, nombrei

CAj nombrep -> PLATO

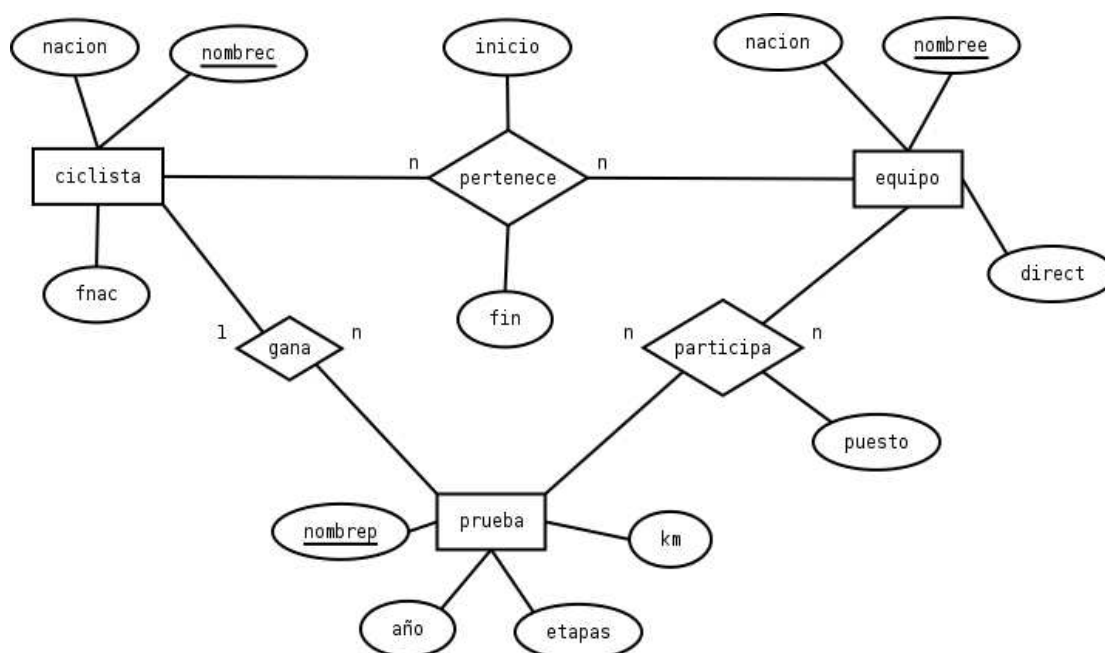
CAj nombrei -> INGRED

Ejercicio 2.

Con la próxima edición de la Vuelta Ciclista a España, un periódico deportivo quiere crear una base de datos para mantener información sobre las pruebas ciclistas por etapas. En la base de datos debe aparecer información sobre los ciclistas, los equipos a los que pertenecen y las pruebas en las que cada equipo ha participado (se asume que participa todo el equipo).

De cada ciclista, se desea conocer su nombre, nacionalidad y fecha de nacimiento, así como el equipo al que pertenece, manteniendo la fecha de inicio y fin de contrato con el equipo.

De cada equipo también se desea conocer su nombre, su nacionalidad, el nombre del director y las pruebas en las que ha participado, con su nombre, año de edición, nº de etapas, kilómetros totales y puesto que ocupó el equipo en la clasificación final. Un dato adicional para las pruebas es saber el nombre del ciclista que quedó ganador.



CICLISTA

(nombrec, nacion, fnac)

CP nombrec

EQUIPO

(nombree, nacion, direct)

CP nombree

PRUEBA

(nombrep, año, etapas, km, nombrec)

CP nombrep

CAj nombrec -> CICLISTA

PERTENECE

(nombrec, nombree, inicio, fin)

CP nombrec, nombree

CAj nombrec -> CICLISTA

CAj nombree -> EQUIPO

PARTICIPA

(nombree, nombrep, puesto)

CP nombree, nombrep

CAj nombree -> EQUIPO

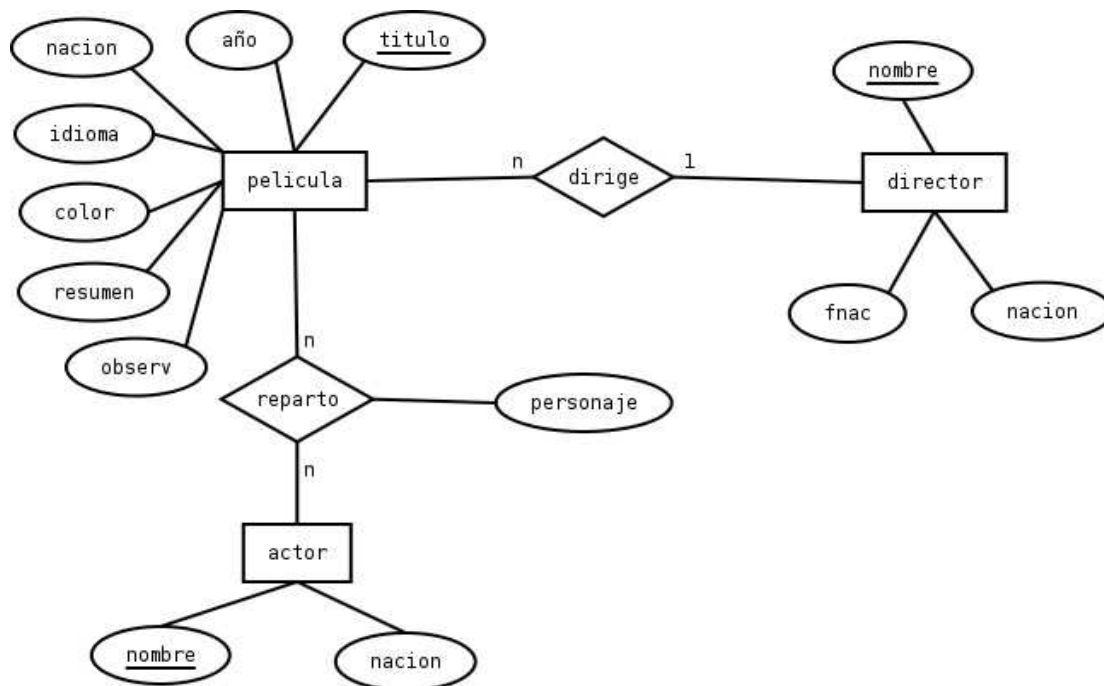
CAj nombrep -> PRUEBA

Ejercicio 3

Acabas de empezar tu colección de películas y quieres hacer una base de datos para construir su ficha técnica. De cada película, necesitas su título, año, nacionalidad y algunos datos de su director: el nombre, la fecha de nacimiento y su país de origen. Además, quieres saber su idioma, si es en blanco y negro o en color, si tiene alguna restricción de edad, un resumen y poder poner tus propias observaciones.

La ficha técnica de cada película también debe incluir el reparto de actores, donde aparecerá su nombre, su nacionalidad y el nombre del personaje que tiene en la película.

Diseñar una base de datos que se ajuste al requerimiento arriba expuesto, identificando tablas, atributos, claves principales y relaciones existentes.



DIRECTOR
(nombre, nacion, fnac)
CP nombre

PELICULA
(titulo, año, nacion, idioma, color, resumen, observ, nombre)
CP titulo
CAj nombre -> DIRECTOR

ACTOR
(nombre, nacion)
CP nombre

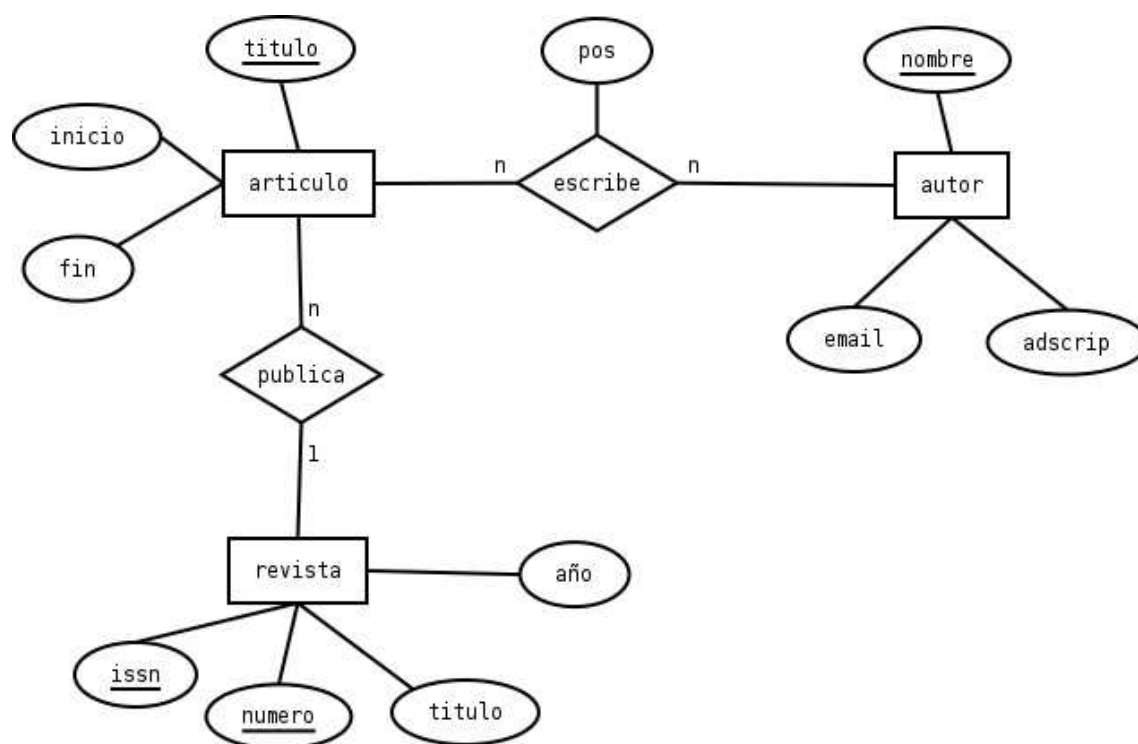
REPARTO
(titulo, nombre, personaje)
CP titulo, nombre
CAj titulo -> PELICULA
CAj nombre -> ACTOR

Ejercicio 4

Se desea crear una base de datos que contenga información sobre las revistas a las que estás suscrito o compras habitualmente. De cada revista, se pide su título, el ISSN (un código que identifica a la publicación), el número y el año de publicación. También se desea almacenar información de cada uno de los artículos publicados: el título, la página de inicio y la página de fin. Se asume que no hay dos artículos con el mismo título.

Cada artículo puede estar escrito por varios autores, de quienes interesa conocer su nombre, una dirección de correo electrónico y su adscripción, así como un número que indique la posición en la que aparece en cada artículo: un 1 si es el primer autor, un 2 si aparece en segundo lugar, etc.

Diseñar una base de datos que se ajuste al requerimiento arriba expuesto, identificando tablas, atributos, claves principales y relaciones existentes.



REVISTA
(issn, numero, año, titulo)
CP issn, numero

ARTICULO
(titulo, inicio, fin, issn, numero)
CP titulo
CAj issn, numero -> REVISTA

AUTOR
(nombre, email, adscrip)
CP nombre

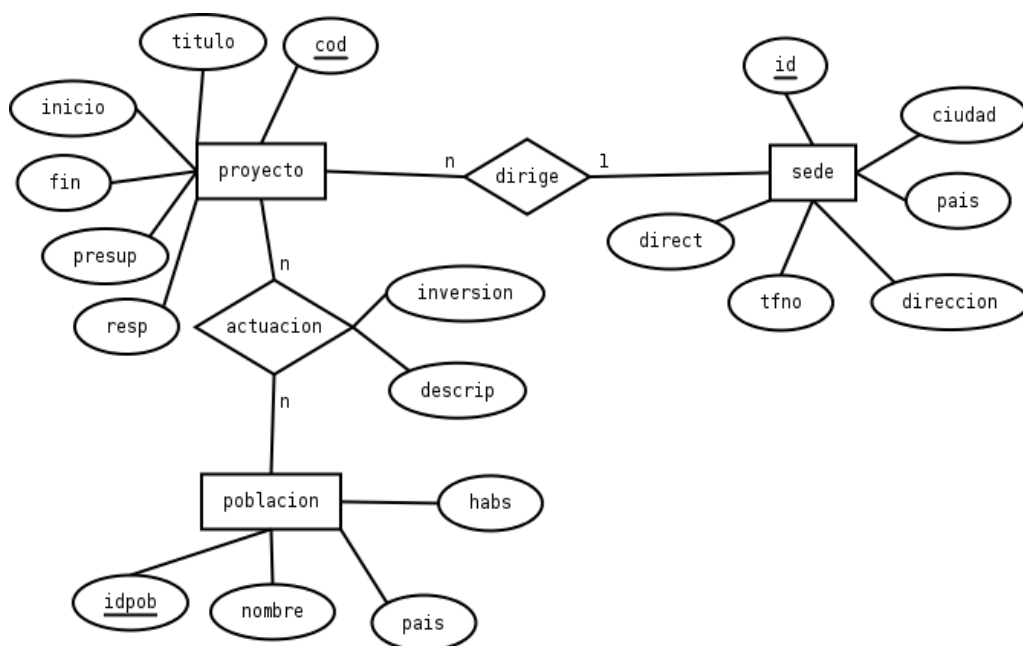
ESCRIBE
(titulo, nombre, pos)
CP titulo, autor
CAj titulo -> ARTICULO
CAj nombre -> AUTOR

Ejercicio 5

Una ONG desea elaborar una base de datos para llevar el seguimiento de todos sus proyectos. Tiene diversas sedes en varios países que se encargan de gestionar y coordinar los proyectos de ese país, cada uno de los cuales puede afectar a una o varias poblaciones.

Sobre la sedes se desea mantener un identificador, la ciudad y país en el que se encuentra, junto co su dirección, un teléfono de contacto y el nombre del director. Cada sede gestiona un conjunto de proyectos, con un código, un título, fechas de inicio y finalización, el presupuesto asignado y el nombre del responsable.

De cada proyecto es necesario conocer qué actuaciones se realizan en cada población, almacenando el nombre, país y nº de habitantes y un identificador para diferenciarlas. Además se desea la inversión del proyecto que corresponde a la población y una pequeña descripción de la actuación.



SEDE
(id, ciudad, pais, direccion, tfno, direct)
CP id

POBLACION
(idpob, nombre, pais, habs)
CP idpob

PROYECTO
(cod, titulo, inicio, fin, presup, resp, id)
CP cod
CAj id -> SEDE

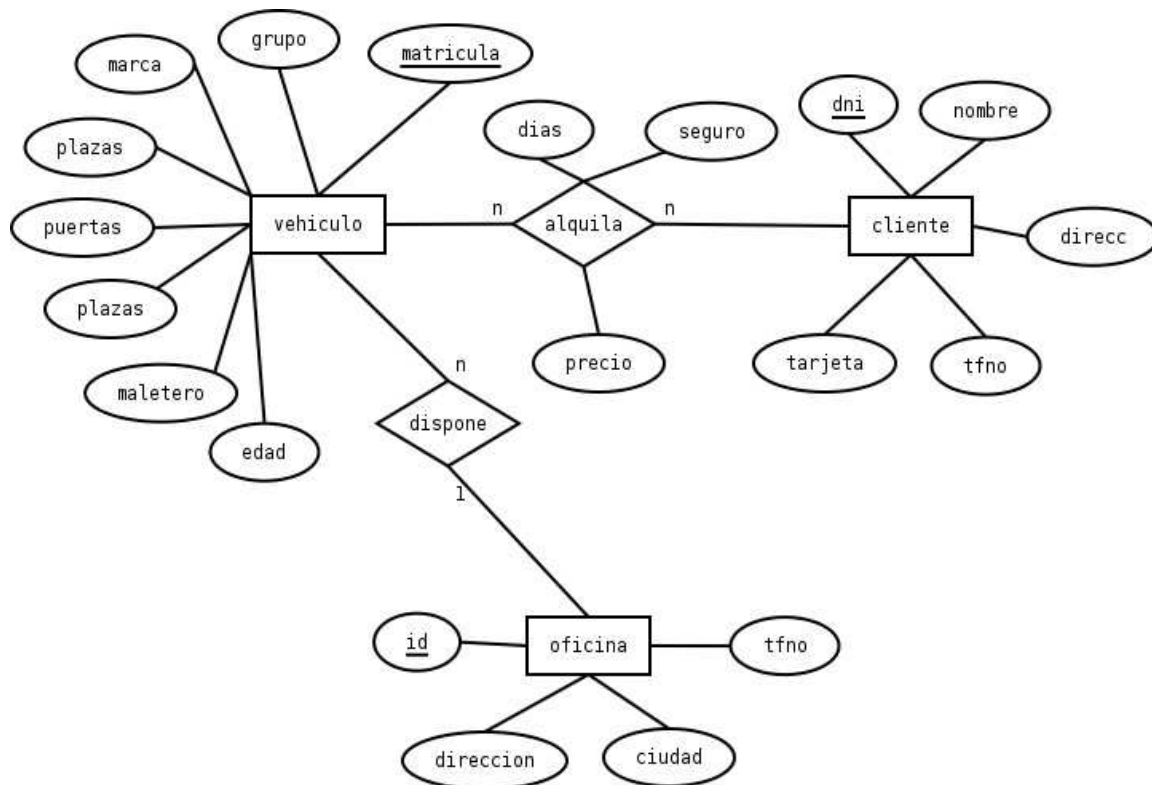
ACTUACION
(cod, idpob, inversion, descrip)
CP cod, idpob
CAj cod -> PROYECTO
CAj idpob -> POBLACION

Ejercicio 6

Una empresa de alquiler de vehículos desea conocer en todo momento el estado de su flota. La empresa tiene diversas oficinas repartidas por todo el territorio español. Cada oficina se identifica por un código único y se caracteriza por la ciudad en la que se encuentra y su dirección completa (calle, número y código postal) y teléfono.

En cada oficina hay disponible un conjunto de coches, de los cuales se conoce su matrícula, el grupo al que pertenece: A, B, C, D, E, F o G (depende del tipo y tamaño del vehículo), la marca, el modelo, el número de puertas, el número de plazas, la capacidad del maletero y la edad mínima exigida para el alquiler.

Para llevar el control del estado de cada vehículo, la empresa mantiene un registro de todos los alquileres que ha sufrido, indicando para cada uno de ellos el nombre del conductor, su DNI, su dirección, un teléfono de contacto y un número de tarjeta de crédito sobre la que realizar los cargos correspondientes. Además de esta información de los clientes, para cada alquiler se almacena su duración (en días), el tipo de seguro contratado y el precio total.



OFICINA
(id, direccion, ciudad, tfno)
CP id

CLIENTE
(dni, nombre, direccion, tfno, tarjeta)
CP dni

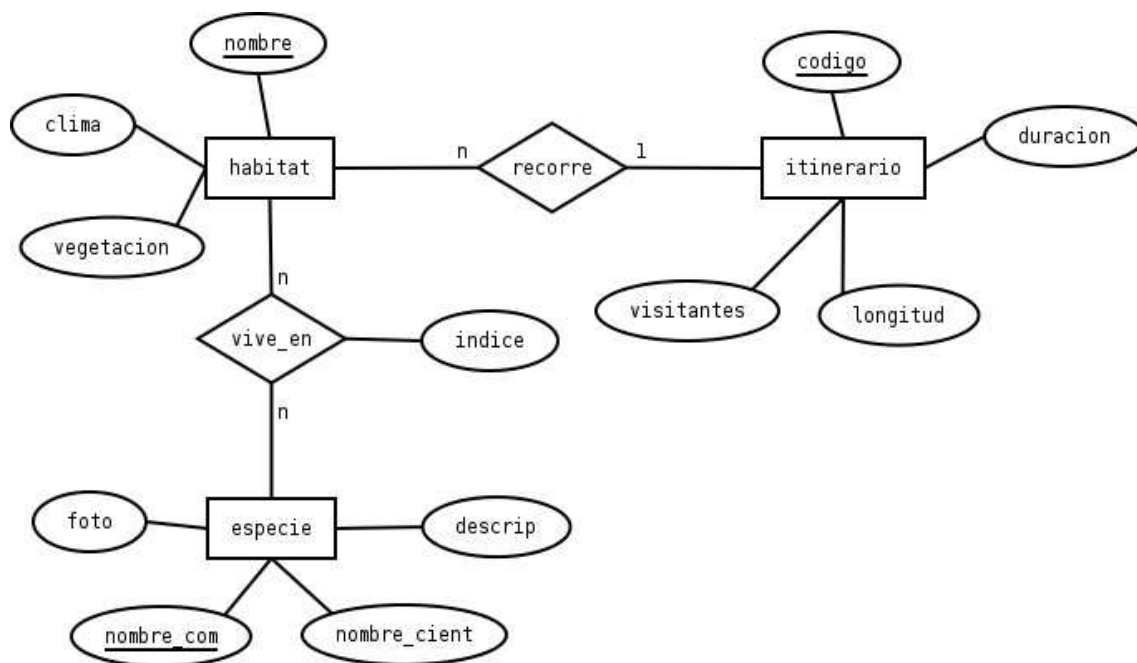
VEHICULO
(matricula, grupo, marca, modelo, plazas, puertas, maletero, edad, id)
CP matricula
CAj id -> OFICINA

ALQUILA
(dni, matricula, dias, seguro, precio)
CP dni, matricula
CAj dni -> CLIENTE
CAj matricula -> VEHICULO

Ejercicio 7

Un parque zoológico quiere construir una BD para organizar las especies que posee y los distintos itinerarios para visitar el parque. La información se estructura de la siguiente forma. De las especies, se desea conocer su nombre común y su nombre científico, así como una descripción general y una fotografía. Cada especie puede vivir en distintos hábitats naturales, definidos por su nombre, clima y vegetación predominante. Cada especie tiene asociado un índice de vulnerabilidad dentro de cada hábitat, que mide el riesgo de extinción de la especie en el dicho hábitat.

Para organizar las visitas, y en función de los hábitats que desee recorrer un visitante, el parque le ofrece una serie de recorridos por los hábitats, que se identifican por su código y se caracterizan por su duración estimada, longitud y número máximo de visitantes permitidos. Un hábitat sólo puede formar parte de un itinerario.



ESPECIE

(nombre_com, nombre_cient, foto, descrip)
CP nombre_com

HABITAT

(nombre, clima, vegetacion, codigo)
CP nombre
CAj codigo -> ITINERARIO

ITINERARIO

(codigo, duracion, longitud, visitantes)
CP codigo

VIVE_EN

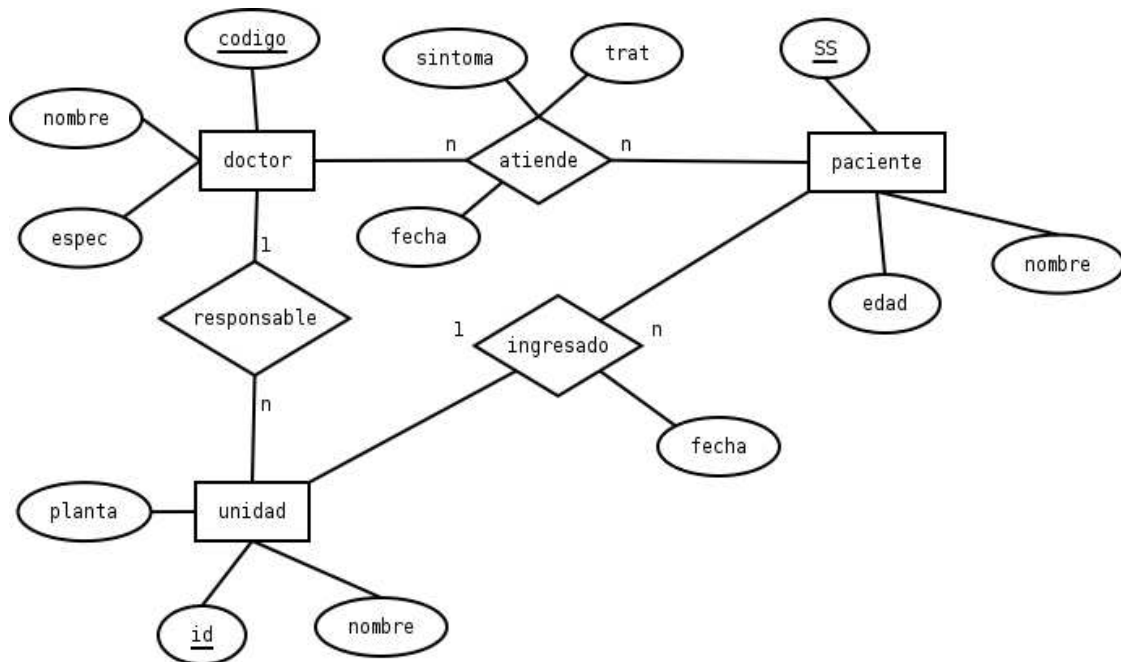
(nombre_com, nombre, indice)
CP nombre_com, nombre
CAj nombre_com -> ESPECIE
CAj nombre -> HABITAT

Ejercicio 8

Una clínica desea mantener una base de datos con el historial de todos los pacientes que tiene ingresados.

La clínica está dividida en varias unidades, cada una de las cuales tiene un identificador, su nombre y la planta en la que se encuentra. La unidad tiene un único doctor responsable, del cual se desea almacenar su código, el nombre y su especialidad.

Cuando llega un paciente, se le ingresa en una unidad y se registra su número de la S.S., nombre, edad y fecha de ingreso. Durante toda su estancia en la clínica, se anotan todas las intervenciones que realizan cada uno de los doctores, indicando la fecha, el síntoma observado y el tratamiento prescrito.



PACIENTE
(SS, nombre, edad, id)
CP SS
CAj id -> UNIDAD

UNIDAD
(id, nombre, planta, codigo)
CP id
CAj codigo -> DOCTOR

DOCTOR
(codigo, nombre, espec)
CP codigo

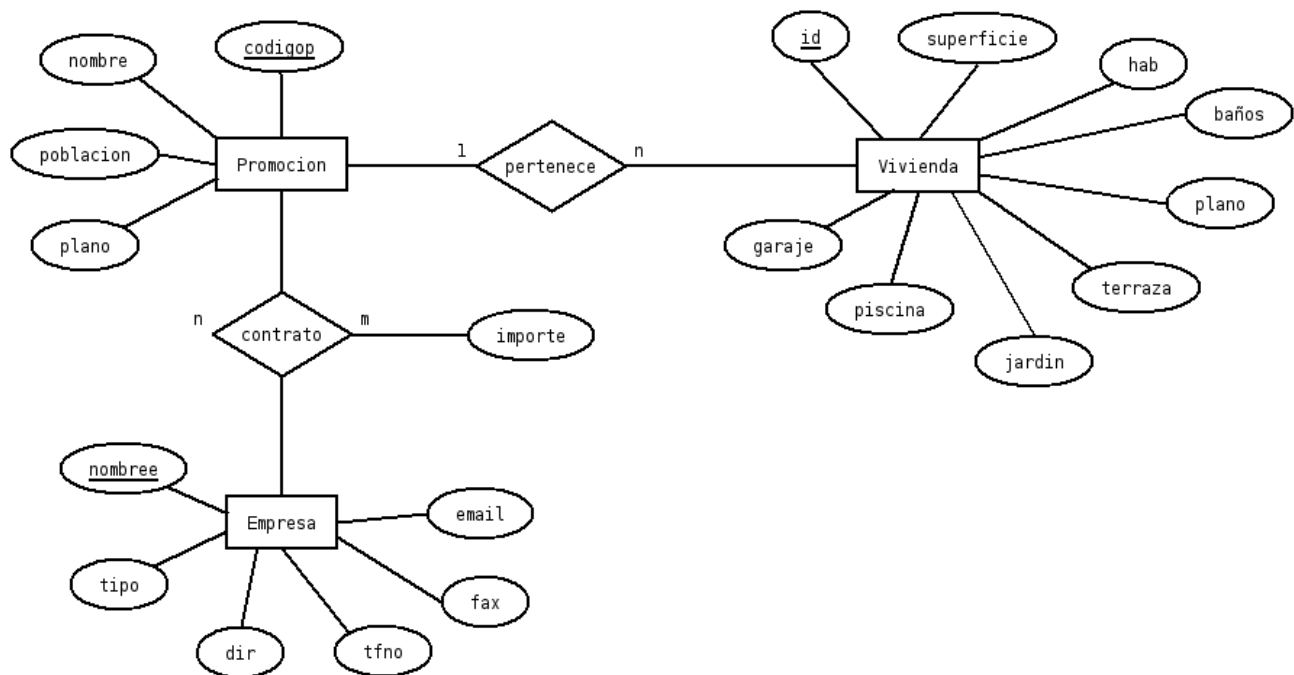
ATIENDE
(SS, codigo, fecha, sintoma, trat)
CP SS, codigo
CAj SS -> PACIENTE
CAj codigo -> DOCTOR

Ejercicio 9

Una promotora inmobiliaria de viviendas quiere crear una base de datos para llevar un registro de las promociones que tiene en venta. Una promoción está caracterizada por un código interno, su nombre, la población en la que está ubicada y un plano de situación.

Cada promoción está formada por un conjunto de viviendas, cada una de las cuales tiene un identificador, superficie, número de habitaciones, número de baños, el plano de la vivienda, una foto y el precio. Además es necesario indicar si tiene o no terraza, jardín privado, piscina y garaje.

Para la construcción, publicidad y venta de una promoción puede contratar a distintas empresas. De cada empresa se desea almacenar su nombre, tipo, dirección completa, teléfono, fax y dirección de correo electrónico, así como el importe del contrato entre la empresa y la promotora.



PROMOCION

(codigop, nombre, pooblacion, plano)

CP: codigop

VIVIENDA

(id, superficie, hab, baños, plano, terraza, jardin, piscina, garaje, codigop)

CP: id

CAj: codigop -> PROMOCION

EMPRESA

(nombree, tipo, dir, tfno, fax, email)

CP: nombree

CONTRATO

(codigop, nombree, importe)

CP: codigop, nombree

CAj: codigop -> PROMOCION

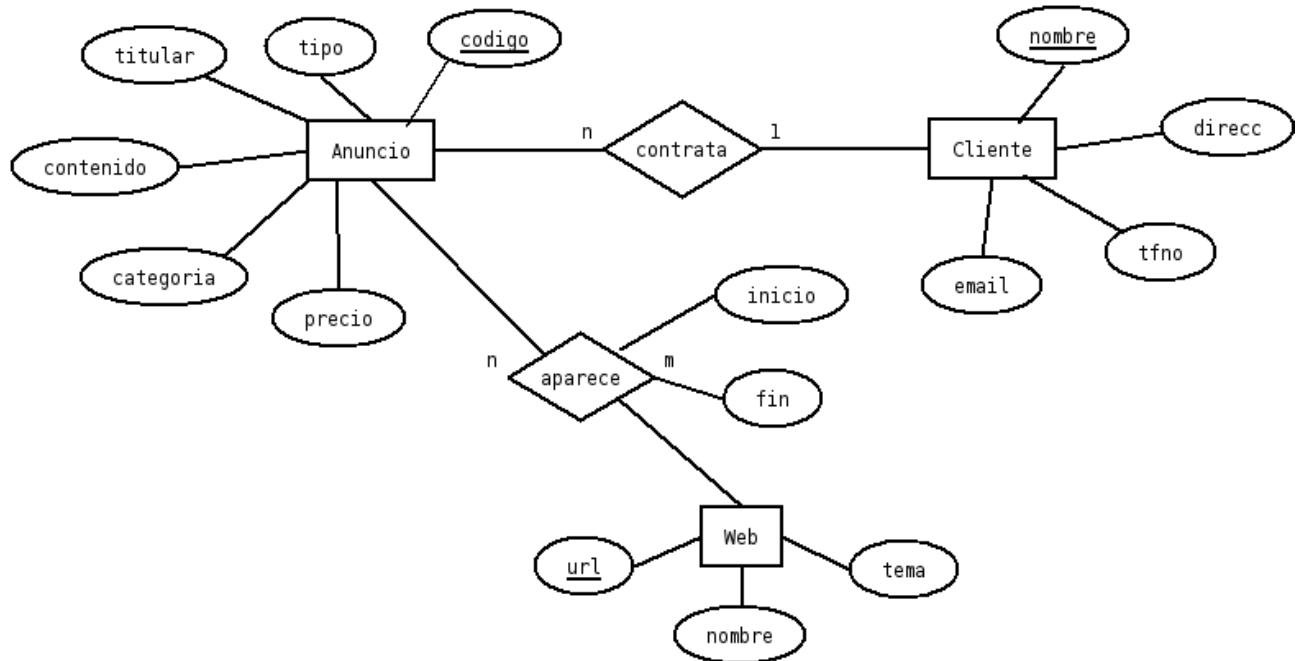
CAj: nombree -> EMPRESA

Ejercicio 10

Una agencia de publicidad necesita una base de datos para registrar todas sus campañas en la web.

Sus clientes tienen un nombre, una dirección postal, el número de teléfono y una dirección de email. Cada cliente puede contratar varios anuncios. Los anuncios quedan identificados por un código y se caracterizan por un nombre, tipo (banner, popup, enlace patrocinado,...), título, contenido, categoría (tipo del producto que anuncia) y precio. Los anuncios pueden aparecer en más de una página web.

Cada web se caracteriza por su URL, nombre y tópico de interés. También se debe almacenar la fecha de inicio y de fin de la aparición del anuncio en la página web.



CLIENTE
(nombre, direccion, tfno, email)
CP: nombre

WEB
(url,nombre, tema)
CP: url

ANUNCIO
(codigo, tipo, titular, contenido, categoria, precio, nombre)
CP: codigo
CAj: nombre -> CLIENTE

APARECE
(codigo, url, inicio, fin)
CP: codigo, url
CAj: codigo -> ANUNCIO
CAj: url -> WEB

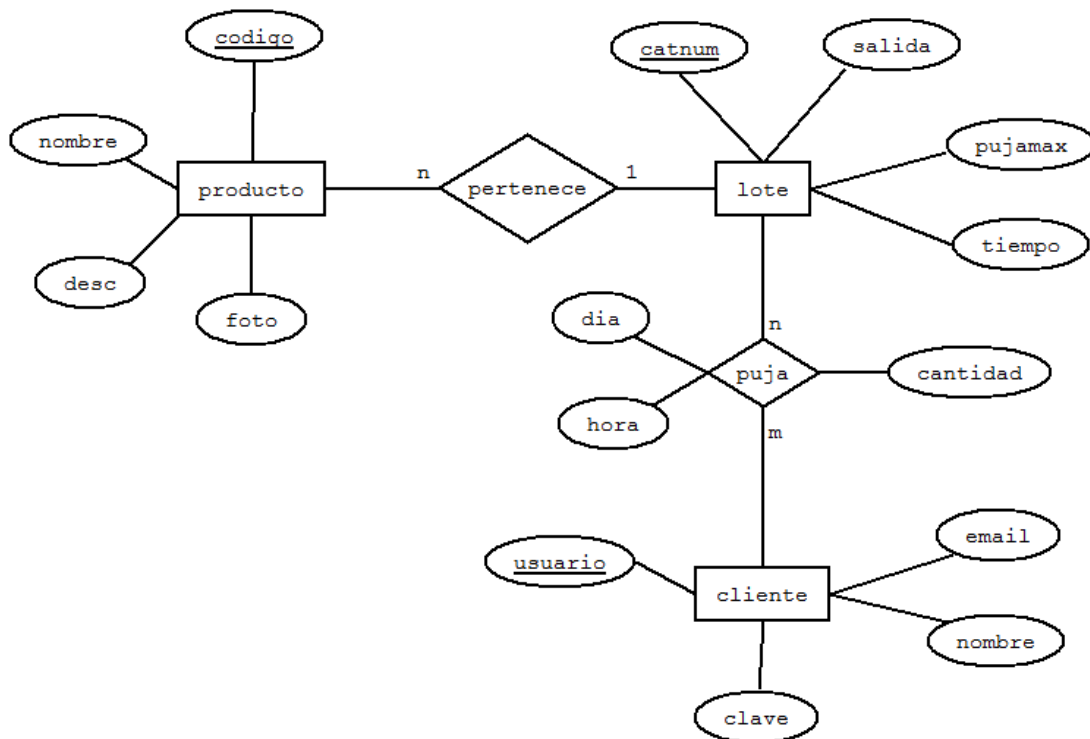
Ejercicio 11

Una casa de subastas en Internet quiere mantener una base de datos para registrar todas las transacciones que realiza.

Los productos que se subastan se agrupan en lotes. Cada lote tiene un número de catálogo, un precio de salida, la mayor puja realizada hasta el momento y el tiempo que queda de subasta. De cada producto se almacena un código, su nombre, una descripción corta y una fotografía.

Los clientes que participan en la subasta deben pujar por un lote completo (no se admiten pujas por productos individuales). Cada vez que un cliente puja, queda registrada la cantidad, el día y la hora en la que se ha producido. Para identificar a los clientes, todos deben tener un nombre de usuario, además de una contraseña, su nombre y una dirección de correo electrónico.

Construye en esquema conceptual y el esquema lógico de la base de datos que contenga la información arriba expuesta.



CLIENTE
(usuario, clave, nombre, email)
CP usuario

PRODUCTO
(código, nombre, desc, foto)
CP código

LOTE
(catnum, salida, pujamax, tiempo, código)
CP catnum
CAj código -> PRODUCTO

PUJA
(catnum, usuario, día, hora, cantidad)
CP catnum, usuario, cantidad
CAj catnum ->LOTE
CAj usuario -> CLIENTE

Ejercicio 12

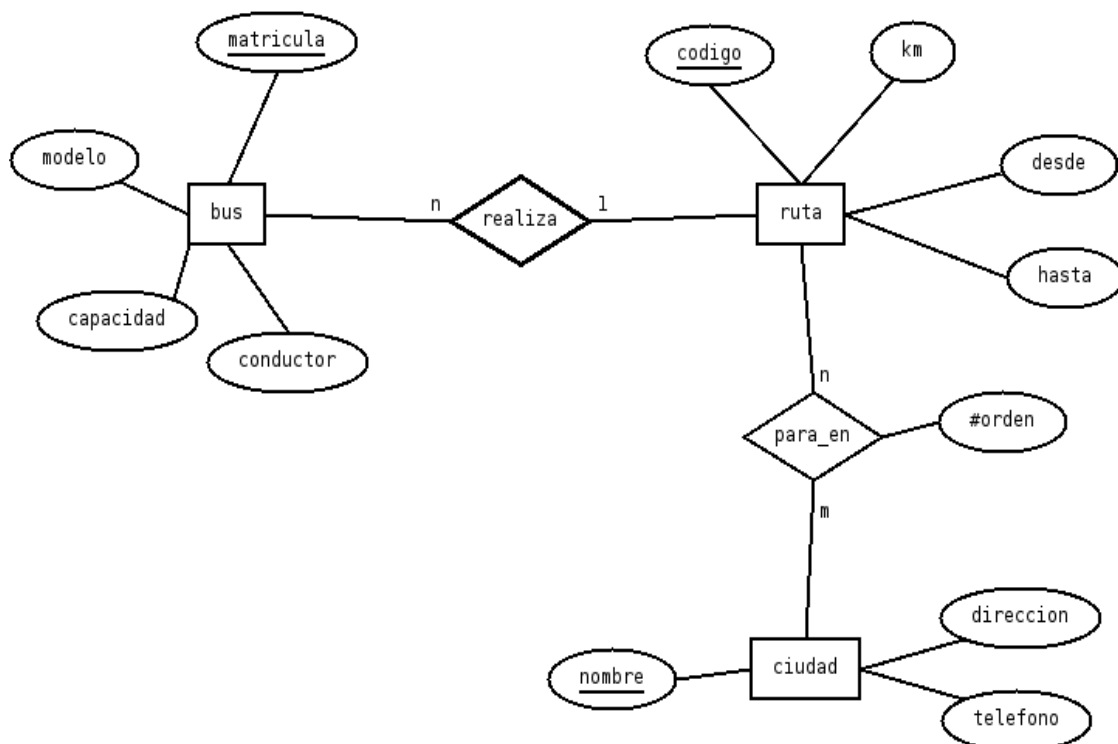
Una empresa de transporte desea crear una base de datos para almacenar información sobre sus rutas.

La empresa dispone de una flota de autobuses que distribuye en una serie de rutas. En cada ruta, el autobús pasa por un conjunto de ciudades en las que tiene parada.

Una ruta se identifica por un código y se caracteriza por los km. totales de recorrido, el origen y el destino final. De cada autobús, se almacena su matrícula, el modelo, su capacidad (plazas) y el nombre del conductor. Se asume que un autobús sólo puede estar realizando una ruta.

Las rutas tienen paradas en distintas ciudades. De cada ciudad, almacenaremos el nombre, junto con la dirección y el teléfono del lugar de parada. Para organizar las rutas, cada parada tiene un número de orden, que puede variar entre distintas rutas (una misma ciudad puede pertenecer a varias rutas).

Construye en esquema conceptual y el esquema lógico de la base de datos que contenga la información arriba expuesta.



BUS

(matricula, modelo, capacidad, conductor, codigo)

CP: matricula

CAj: codigo → RUTA

RUTA

(codigo, km, desde, hasta)

CP: codigo

CIUDAD

(nombre, direccion, telefono)

CP: nombre

PARA_EN

(codigo, nombre, #orden)

CP: codigo, nombre

CAj: codigo → RUTA

CAj: nombre → CIUDAD

Ejercicio 13

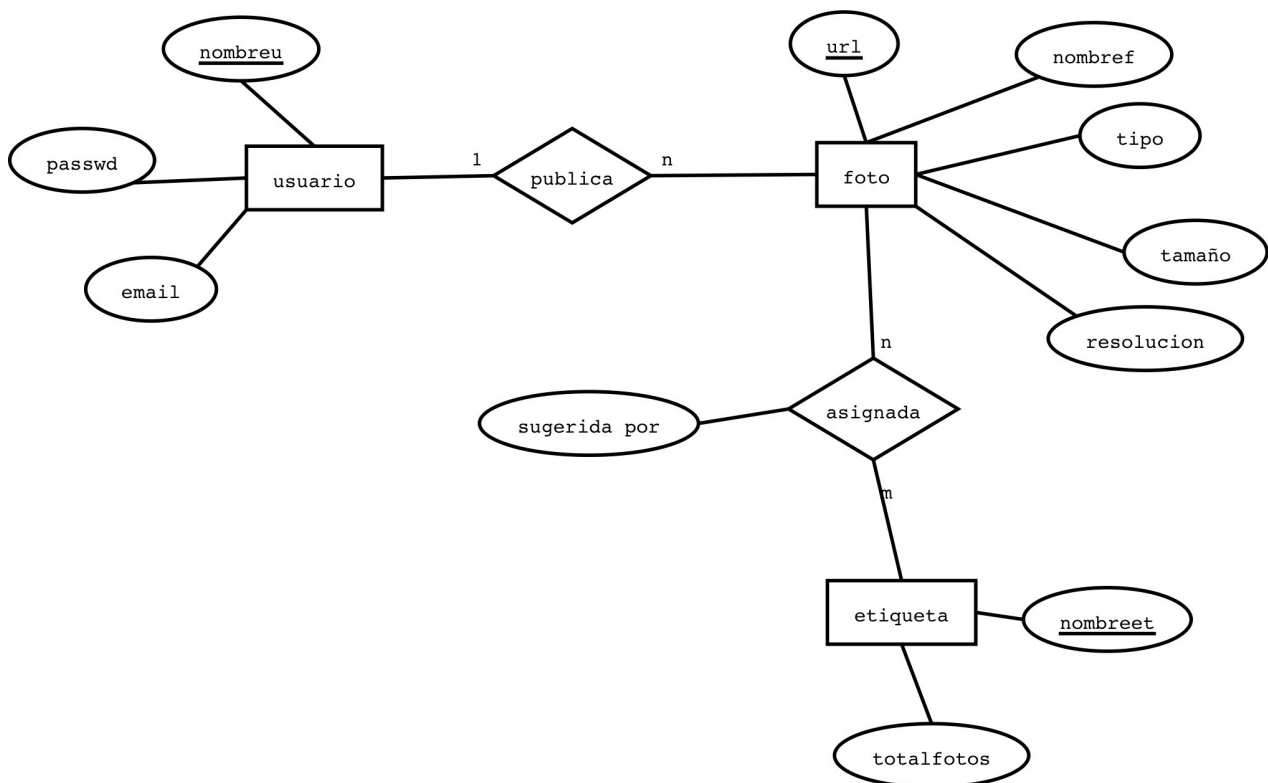
Una empresa que proporciona almacenamiento de fotos en la web necesita una base de datos para almacenar todos los archivos.

Las fotos se caracterizan por su URL (es única); además se almacena el nombre del archivo, tipo, tamaño, resolución y un campo que indica si la fotografía es pública o privada.

Los usuarios pueden añadir todas las etiquetas que necesiten a cada una de sus fotos para clasificarlas. Además del nombre de la etiqueta, se almacena el número total de fotos que la emplean. El sistema también puede sugerir etiquetas adicionales, por lo que se debe indicar quién ha sugerido la etiqueta: el usuario o el sistema. Y eso para cada foto.

Para identificar a los usuarios, se almacena el nombre de usuario, su password y su dirección de correo electrónico.

Construye el esquema conceptual y el esquema lógico de la base de datos que cumpla los requerimientos expuestos.



USUARIO
(nombreu, passwd email)
CP: nombreu

ETIQUETA
(nombreet, totalfotos)
CP: nombreet

FOTO
(url, nombref, tipo, tamaño, resolucion, nombreu)
CP: url
CAj: nombreu-> USUARIO

ASIGNADA
(url, nombreet, sugerida_por)
CP: url, nombreet
CAj: url -> FOTO
CAj: nombreet -> ETIQUETA

Ejercicio 14

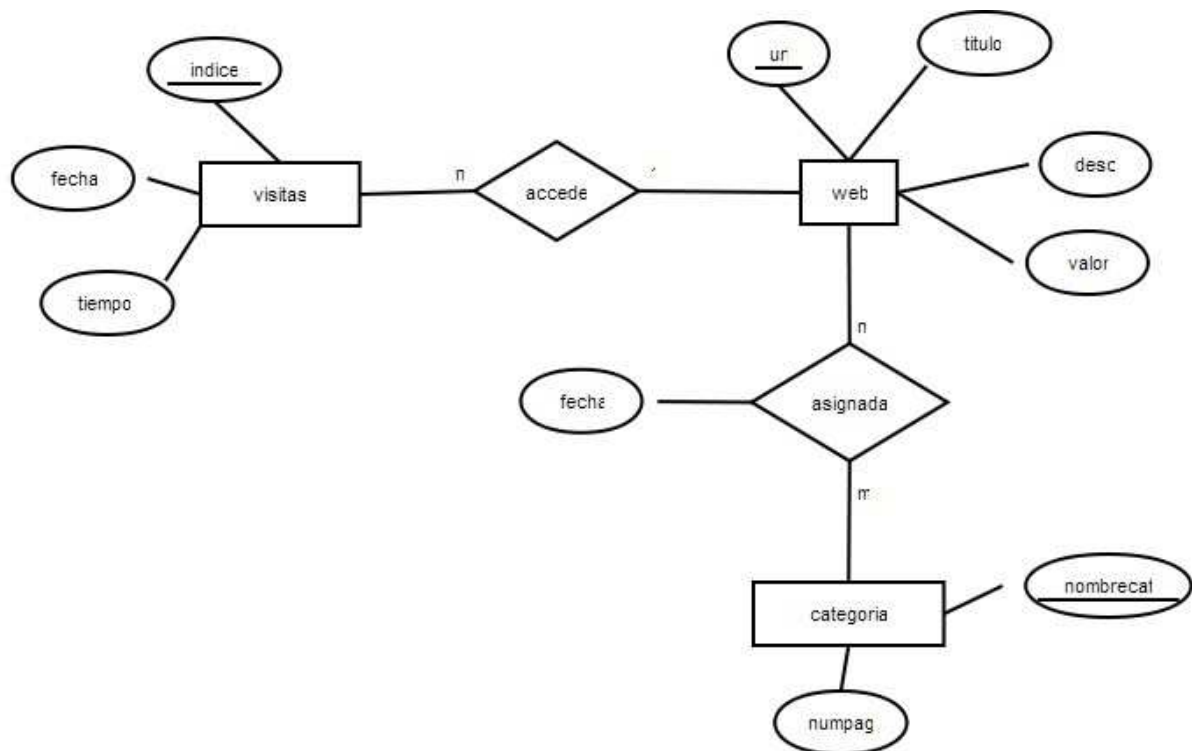
Necesitas tener una base de datos con los enlaces de las páginas web que visitas para mantener un histórico más completo que el servicio que proporcionan los Favoritos de tu navegador.

De cada página que visitas, te interesa su URL (su dirección), su nombre y una descripción breve y una valoración global.

Las páginas se van a agrupar por categorías en lugar de por carpetas, de manera que a cada página se le pueden asignar varias categorías. De cada categoría, además de su nombre, queremos saber cuántas páginas web pertenecen a ella y en qué fecha se asignó esa categoría a la página.

Para conocer el histórico de navegación, cada vez que visitemos una página web, se añadirá automáticamente a la base de datos una entrada con la fecha de consulta y el tiempo de permanencia en el sitio. Las visitas tienen asociadas un número de orden único.

Construye el esquema conceptual y el esquema lógico de la base de datos que cumpla los requerimientos expuestos.



WEB
(url, titulo, desc, valor)
CP: url

VISITAS
(indice, fecha, tiempo, url)
CP: indice
CAj: url -> WEB

CATEGORIA
(nombrecat, numpag)
CP: nombrecat

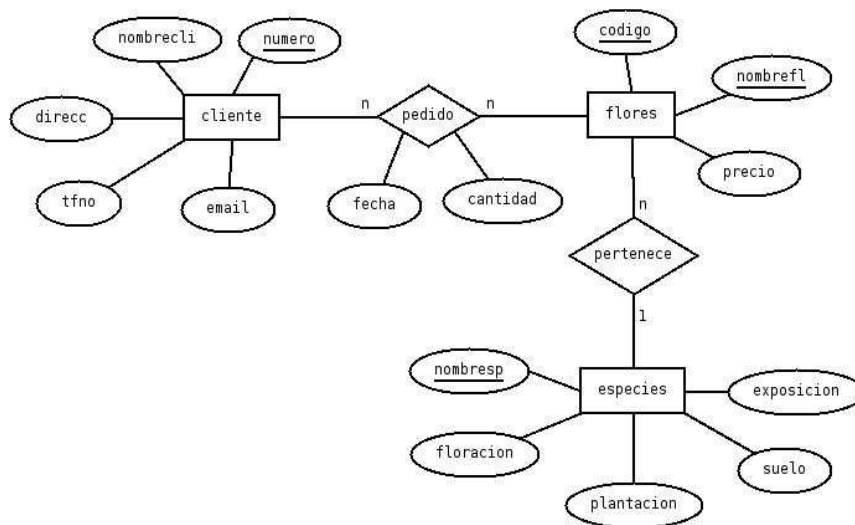
ASIGNADA
(nombrecat, url, fecha)
CP: nombrecat, url
CAj: nombrecat -> CATEGORIA
CAj: url -> WEB

Ejercicio 15

Una pequeña floristería desea ampliar su negocio y realizar ventas a través de Internet. Y para ello necesita crear una base de datos.

Cada pedido incluye un número de pedido, la fecha de venta, el importe total y una lista con las flores solicitadas y en qué cantidad. Las flores se identifican mediante un código, su nombre y el precio de venta. Las flores pertenecen a una especie determinada. Para cada especie, se almacena el nombre, la época de floración, la estación de plantación, el tipo de suelo apropiado y el tiempo de exposición recomendado.

Construye el modelo conceptual y el modelo lógico correspondientes al enunciado expuesto



FLORES

(codigo, nombrefl, precio, nombresp)

CP: codigo

CAj: nombresp -> ESPECIES

ESPECIES

(nombresp, floración, plantacion, suelo, exposicion)

CP: nombresp

PEDIDOS

(numped, fecha, importe)

CP: numped

CONTIENE

(numped, codigo, cantidad)

CP: numped, codigo\

CAj: numped -> PEDIDOS

CAj: codigo -> FLORES

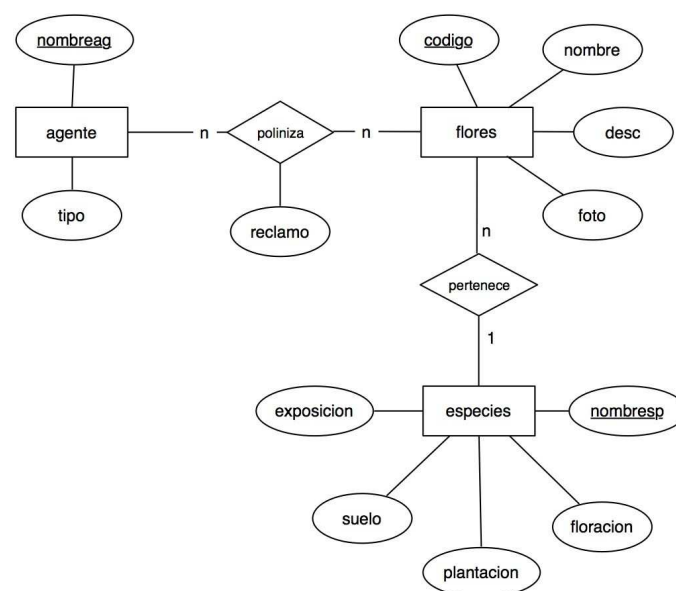
Ejercicio 16

Un centro de investigación desea recoger información sobre los procesos de polinización que se producen en una zona determinada, para lo que necesita diseñar una base de datos.

Las flores se identifican mediante un código, su nombre, una descripción y una fotografía. Las flores pertenecen a una especie determinada y para cada especie, se almacena el nombre, la época de floración, la estación de plantación, el tipo de suelo apropiado y el tiempo de exposición recomendado.

Cada flor puede ser polinizada por diversos agentes polinizadores (algunos físicos, como el viento, y otros animados, como pájaros o insectos). Para cada agente se almacena su nombre, su tipo y un subtipo. Las flores pueden usar distintos tipos de reclamo para atraer a distintos agentes (colores, olores, formas...)

Construye el modelo conceptual y el modelo lógico correspondientes al enunciado expuesto.



FLORES

(codigo, nombrefl, precio, nombresp)

CP: codigo

CAj: nombresp -> ESPECIES

ESPECIES

(nombresp, floración, plantacion, suelo, exposicion)

CP: nombresp

AGENTE

(nombreag, tipo, subtipo)

CP: nombreag

POLINIZA

(nombreag, codigo, reclamo)

CP: nombreag, codigo

CAj: nombreag -> AGENTE

CAj: codigo -> FLOREENTE

CAj: codigo -> FLORES

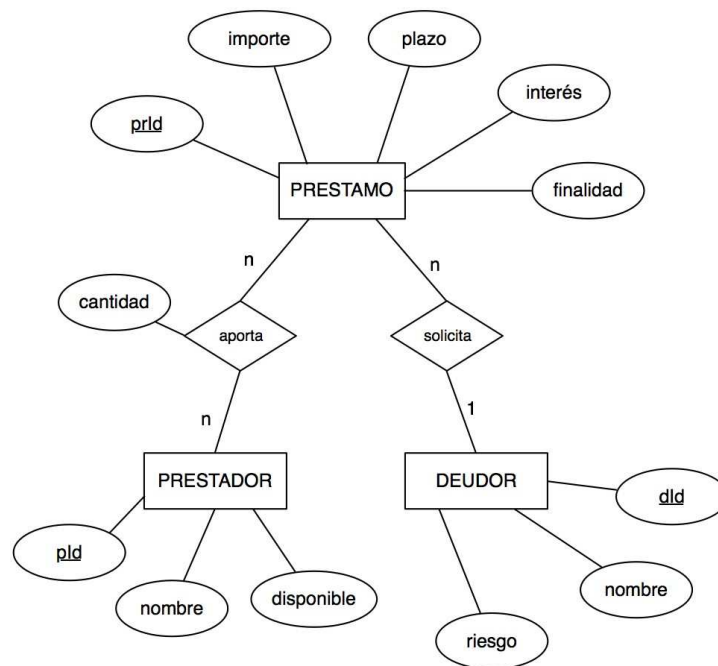
Ejercicio 17

Prestame.com es una empresa que se dedica al préstamo entre personas (p2p lending).

Los prestadores que desean prestar dinero a otros se registran con un id, su nombre y la cantidad de dinero disponible para las operaciones. Los deudores se identifican por su id y además el sistema almacena su nombre y un valor de riesgo en función de su situación personal.

Cuando el deudor solicita un préstamo, se añade un código de préstamo, el importe total, el plazo de devolución, su interés y la finalidad del mismo. Los prestadores indican qué cantidad quieren aportar a un préstamo. Un prestador puede aportar distintas cantidades parciales a varios préstamos.

Construye en el modelo conceptual y el modelo lógico correspondientes al enunciado expuesto.



DEUDOR

(dId, nombre, riesgo)

CP: dId

PRESTAMOS

(prId, importe, plazo, interés, finalidad, dId)

CP: prId

CAj: dId -> DEUDOR

PRESTADOR

(pId, nombre, disponible)

CP: numped

APORTA

(pId, prId, cantidad)

CP: pId, prId

CAj: pId -> PRESTADOR

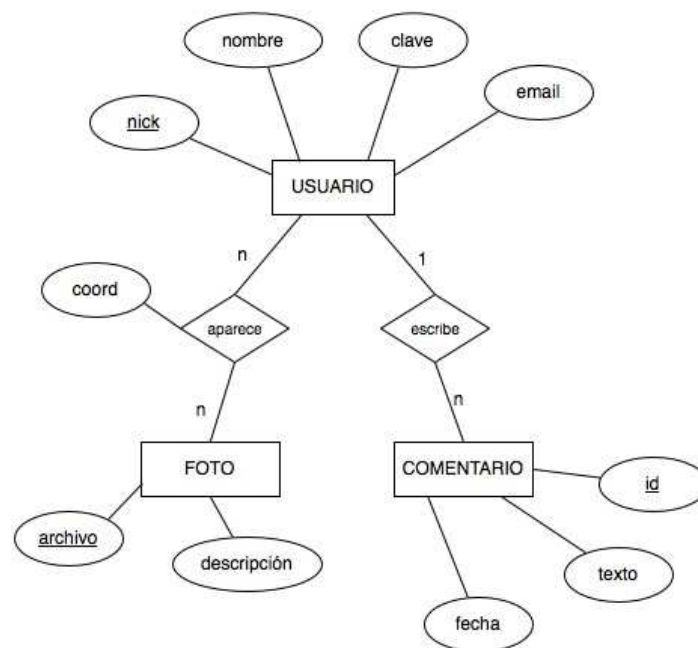
CAj: prId -> PRESTAMO

Ejercicio 18

En redes sociales como Facebook los perfiles de usuario son una herramienta fundamental. Crea una base de datos sencilla que permita almacenarlos.

Los usuarios tienen un nombre real, su nick (que es único), una clave de acceso y una cuenta de correo electrónico. Los usuarios pueden escribir comentarios, los cuales están formados por un texto y la fecha de creación. Además, los usuarios pueden aparecer en fotografías. Una fotografía tiene un nombre de archivo y una descripción opcional. Para marcar a los usuarios en las fotografías, se emplea un marco del que se guardan sus coordenadas.

Construye en el modelo conceptual y el modelo lógico correspondientes al enunciado expuesto.



USUARIO
(nick, nombre, clave, email)
CP: nick

FOTO
(archivo, descripción)
CP: archivo

COMENTARIO
(id, texto, fecha, nick)
CP: id
CAj: nick -> USUARIO

APARECE
(nick, archivo, coord)
CP: nick, archivo
CAj: nick -> USUARIO
CAj: archivo -> FOTO

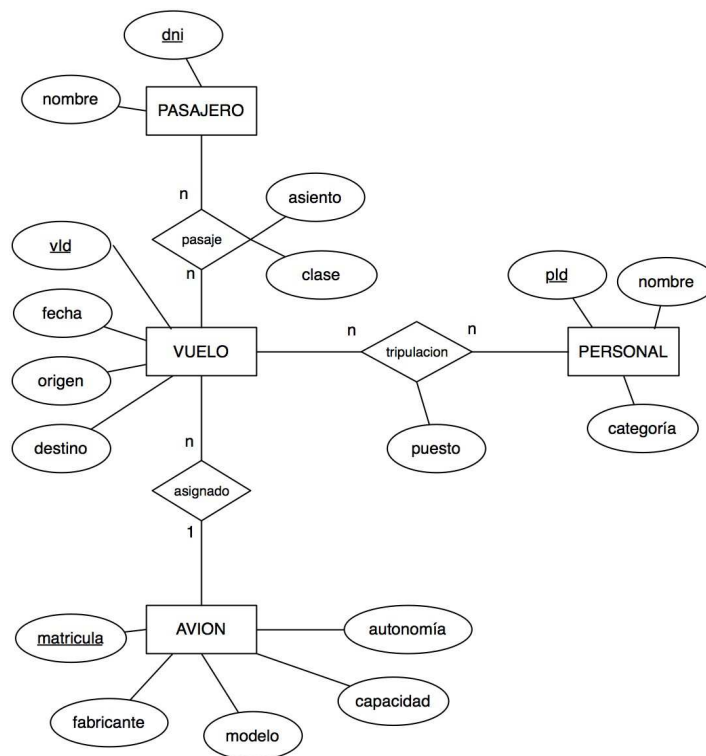
Ejercicio 19

Una compañía aérea necesita una base de datos para registrar la información de sus vuelos.

Los vuelos están caracterizados por un Id, la fecha y los aeropuertos de origen y destino. Cada vuelo es realizado por un avión. Los aviones tienen una matrícula que los identifica, el fabricante, un modelo e información sobre su capacidad (número máximo de pasajeros) y autonomía de vuelo (en horas). La tripulación asignada al vuelo está formada por el personal de la propia compañía. De cada trabajador se conoce su id, su nombre y su categoría profesional, así como el puesto que ocupa en cada vuelo en particular.

Por último, para cada vuelo, se almacena la lista completa de pasajeros, con su dni, el nombre, el asiento que ocupa y su clase (turista, primera o business).

Construye en el modelo conceptual y el modelo lógico correspondientes al enunciado expuesto.



PASAJERO
(dni, nombre)
CP: dni

VUELO
(vId, fecha, origen, destino, matricula)
CP: vId
CAj: matricula -> AVION

AVION
(matricula, fabricante, modelo, capacidad, autonomia)
CP: matricula

PERSONAL
(pId, nombre, categoria)
CP: pId

TRIPULACION
(pId, vId, puesto)
CP: pId, vId
CAj: pId -> PERSONAL
CAj: vId -> VUELO

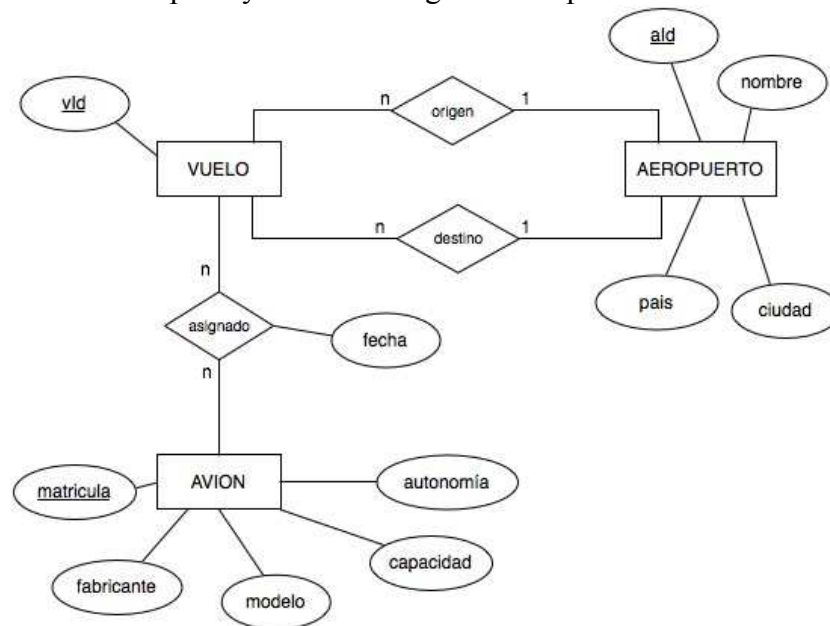
PASAJE
(vId, dni, asiento, clase)
CP: vId, asiento (también válido vId, dni)
CAj: vId -> VUELO
CAj: dni -> PASAJERO

Ejercicio 20

Una compañía aérea necesita una base de datos para registrar la información de sus vuelos.

Los vuelos tienen un identificador único. Además, cada vuelo tiene asignado un aeropuerto de origen y uno de destino (se asume que no hay escalas). Los aeropuertos están identificados por unas siglas únicas (por ejemplo: VLC-Valencia, BCN-Barcelona, MAD-Madrid). Además, de cada aeropuerto se guarda el nombre de la ciudad en la que está situado y el país. Cada vuelo es realizado por un avión. Los aviones tienen una matrícula que los identifica, el fabricante, un modelo e información sobre su capacidad (número máximo de pasajeros) y autonomía de vuelo (en horas). La asignación de aviones a vuelos no es única, así que es necesario saber la fecha en la que un avión realizó cada uno de los vuelos asignados.

Construye en el modelo conceptual y el modelo lógico correspondientes al enunciado expuesto



VUELO

(vId, origen, destino)

CP: vId

CAj: origen -> AEROPUERTO

CAj: destino -> AEROPUERTO

AVION

(matricula, fabricante, modelo, capacidad, autonomia)

CP: matricula

AEROPUERTO

(aId, nombre, ciudad, país)

CP: aId

ASIGNADO

(vId, matricula, fecha)

CP: vId, fecha (también es válido avión, fecha)

CAj: vId -> VUELO

CAj: matricula -> AVIÓN

