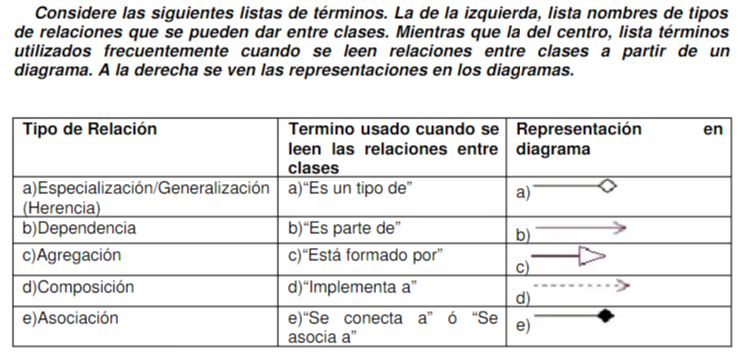
**Ejercicio 1.**





**Ejercicio 2. Empresas**

* Una aplicación necesita almacenar información sobre empresas, sus empleados y sus clientes.
* Ambos se caracterizan por su nombre y su edad.
* Los empleados tienen un salario bruto, mientras que los empleados que son directivos tienen una categoría, así como un conjunto de empleados subordinados.
* De los clientes además se necesita conocer su teléfono de contacto.
* La aplicación necesita mostrar los datos de empleados y clientes.

**Ejercicio 3. Ordenador.**

Un ordenador posee, como mínimo, los siguientes elementos:

1 Torre, 1 Teclado, 1 Monitor, 1 Ratón.

Por su parte una torre se compone de:

1 o varias unidades de disco duro, varios módulos de memoria RAM, 1 unidad óptica y 1 tarjeta gráfica.

**Ejercicio 4. Coche.**

Representar mediante un diagrama de clases un coche compuesto por 4 ruedas, un chasis y un motor.

**Ejercicio 5. Clasificación seres vivos.**

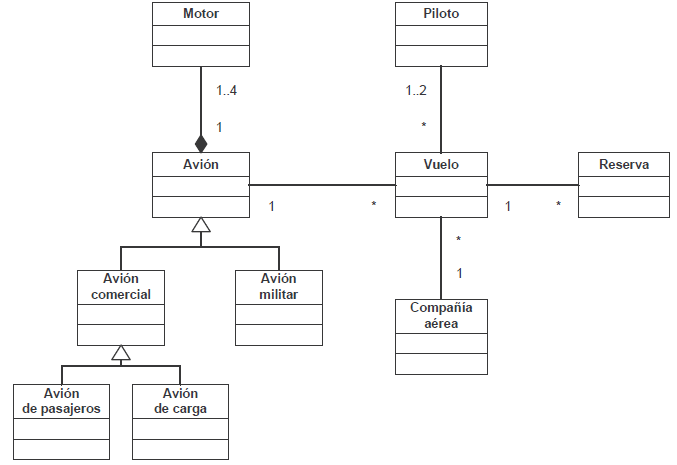
Representar mediante un diagrama de clases la clasificación de los seres vivos:

◦Los animales se deben organizar por su alimentación, esto es: Carnívoros, Herbívoros y Omnívoros

◦Las plantas se deben clasificar por su tipo: Árboles, Arbustos y Hierbas

◦Para los animales, se ponen 3 ejemplos: Conejo, León y Vaca

**Ejercicio 6. Redacta un enunciado para el siguiente diagrama de clases.**



**Ejercicio 7. Animales de la casa**

Diseñar una aplicación orientada a objetos que describa la siguiente situación:

En una **casa** viven cinco animales: una **ballena** llamada " Moby Dick", que no dice nada; un **perro** fiero llamado " Caín", que dice " Grrr"; un perro manso llamado " Abel", que dice "Guau"; un **pingüino** llamado "Adela" que no dice nada, y un **loro** que dice " Lorito bonito", "Pretty Polly" y "Viva mi dueño".

Especificar la jerarquía de herencia, las clases, los atributos y los métodos de cada clase.

**Ejercicio 8. La asociación**

Diseñar un diagrama de clases que modele el proceso de dar de alta a cada una de las personas que se apuntan a una asociación.

De cada persona interesa saber sus datos básicos: NIF, nombre completo y fecha de nacimiento. Cuando cada nuevo socio se da de alta, se le asigna un código de asociado alfanumérico y se anota la fecha de alta.

La clase Fecha se modela con tres campos (día, mes y año) de tipo entero. La clase NIF se modela con un campo de tipo entero llamado DNI y un campo de tipo carácter llamado letra.

**Ejercicio 9. El zoológico**

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.

**Ejercicio 10. Agencia de publicidad**

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, pop-up, enlace patrocinado, etc.), 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.

**Ejercicio 11. Instituto**

Se desea diseñar la base de datos de un Instituto. En la base de datos se desea guardar los datos de los profesores del Instituto (DNI, nombre, dirección y teléfono). Los profesores imparten módulos, y cada módulo tiene un código y un nombre. Cada alumno está matriculado en uno o varios módulos. De cada alumno se desea guardar el número de expediente, nombre, apellidos y fecha de nacimiento. Los profesores pueden impartir varios módulos, pero un módulo sólo puede ser impartido por un profesor.

**Ejercicio 12. Clínica “San Patrás”**

La clínica “San Patrás” necesita llevar un control informatizado de su gestión de pacientes y médicos.

De cada paciente se desea guardar el código, nombre, apellidos, dirección, población, provincia, código postal, teléfono y fecha de nacimiento.

De cada médico se desea guardar el código, nombre, apellidos, teléfono y especialidad.

Se desea llevar el control de cada uno de los ingresos que el paciente hace en el hospital.

Cada ingreso que realiza el paciente queda registrado en la base de datos. De cada ingreso se guarda el código de ingreso (que se incrementará automáticamente cada vez que el paciente realice un ingreso), el número de habitación y cama en la que el paciente realiza el ingreso y la fecha de ingreso.

Un médico puede atender varios ingresos, pero el ingreso de un paciente solo puede ser atendido por un único médico. Un paciente puede realizar varios ingresos en el hospital”.

**Ejercicio 13. Concesionario de coches**

A un concesionario de coches llegan clientes para comprar automóviles. De cada coche interesa saber la matrícula, modelo, marca y color. Un cliente puede comprar varios coches en el concesionario. Cuando un cliente compra un coche, se le hace una ficha en el concesionario con la siguiente información: DNI, nombre, apellidos, dirección y teléfono.

El concesionario también dispone de un taller en el que los mecánicos reparan los coches que llevan los clientes. Un mecánico repara varios coches a lo largo del día, y un coche puede ser reparado por varios mecánicos. Los mecánicos tienen un dni, nombre, apellidos, fecha de contratación y salario. Se desea guardar también la fecha en la que se repara cada vehículo y el número de horas que se tardado en arreglar cada automóvil”.

**Ejercicio 14. Juego de ordenador**

En un juego de ordenador existen 2 tipos de jugadores: los principiantes y los avanzados.

Todos ellos deben tener un nombre y un número de vidas. Los principiantes se desplazan andando a unas coordenadas (x,y). Los jugadores avanzados además de andar también pueden conducir un vehículo para desplazarse más rápido a unas coordenadas. Cada vehículo tiene asociada una velocidad que puede ser leída y ajustada a un valor dado, pero no puede superar una velocidad máxima dada. La velocidad máxima sólo se podrá asignar una vez y no podrá ser modificada.

Todos los atributos de las clases serán privados y tendrán métodos públicos para acceder a ellos (get/set) salvo que los requisitos indiquen lo contrario. Debe existir un método que se llame andar y otro conducir.

Crear el diagrama UML de clases para el problema propuesto y que incluya todas las

funcionalidades descritas.

**Ejercicio 15. Películas**

Representa mediante un diagrama de clases la siguiente especificación sobre las personas que participan en una película:

* De cada película se almacena el título, la sinopsis, el año, el género al que pertenece (drama, comedia, acción, terror, romance, aventura, scifi) y el país.
* Sobre las personas que participan en la película es necesario conocer el nombre, los apellidos, la fecha de nacimiento y la nacionalidad.
* Una persona puede participar en una película como actor, director, productor o guionista.
* Una película tiene al menos un director y un guionista.
* Una persona se considera actor si ha actuado al menos en una película.
* Una película puede tener asociados varios tráilers que son editados por una o más personas. No puede existir el tráiler de una película hasta que existe la película.

**Ejercicio 16. Clínica veterinaria**

Un veterinario tiene como pacientes animales y como clientes familias.

* Un cliente es un conjunto de personas que suele corresponderse con una familia.
* Cada cliente tiene un código, el primer apellido del cabeza de familia, un número de cuenta bancaria, una dirección, un teléfono y los nombres y DNI de las personas correspondientes. No existe límite en el número de personas asociadas a una entidad cliente. Además, una persona puede estar dada de alta en varios clientes (p.e.: un hombre que vive con su esposa tiene un gato y como tal pertenece a un cliente, pero también está dado de alta en el cliente asociado con el perro de sus padres)
* Los clientes pueden tener varias mascotas, cada mascota tiene un código, un alias, una especie, una raza, color de pelo, fecha de nacimiento aproximada, peso medio del animal en las últimas 10 visitas y el peso actual del animal.
* Se guardará un historial médico de cada mascota con:
  + Cada enfermedad que tuvo y la fecha en la que enfermó
  + Calendario de vacunación, en el que se registrará la fecha de cada vacuna, la enfermedad de la que se vacuna

**Ejercicio 17. Asociación “Amigos de la Fiesta”**

La asociación "Amigos de la Fiesta" desea recoger en una base de datos toda la información acerca de las corridas de todos que se celebran en España y de todos los datos relacionados con ellas.

Se desea tener información acerca de cada corrida, identificada conjuntamente por un número de orden, la feria en la que se celebra y el año de celebración (por ejemplo: orden = 2, feria = San Isidro, Año = 1999).

En una determinada corrida actúan una serie de toreros (mínimo 1 y máximo 3) de los que desea guardar su DNI, nombre, apodo y fecha en que tomó la alternativa (fecha en la que se convirtió en matador de toros).

En cada corrida un torero obtiene una serie de premios (cuántas orejas, cuántos rabos y si salió por la puerta grande o no) de los que se desea mantener información.

Cada torero puede tener un apoderado del que es protegido. A su vez, un apoderado lo puede ser de varios toreros. De él se desea saber su DNI, nombre, dirección y teléfono.

Una corrida se celebra en una plaza de toros de la que se desea saber su nombre que se supone único, localidad, dirección y aforo. En una misma plaza se pueden celebrar varias corridas de toros.

En cada corrida son estoqueados al menos 6 toros. Cada toro viene identificado por el código de la ganadería a la que pertenece, el año en que nació y un número de orden. Además, se desea mantener información acerca de su nombre y color, así como el orden en que fue toreado. Cada toro pertenece a una ganadería determinada. De cada ganadería se pretende saber su código, localidad y antigüedad (fecha de creación).

**Ejercicio 18. Gestión de pedidos.**

Se parte del siguiente enunciado:

* Dicha aplicación debe guardar la información de los pedidos que realizan sus clientes y la fecha en que se formalizan.
* Los pedidos constan habitualmente de varios productos.
* El coste total del pedido se calcula a partir de los precios individuales de cada producto, así como de la cantidad e impuestos asociados a cada uno.
* Es importante mantener información sobre las existencias de cada producto con el fin de poder informar al cliente si habrá retrasos en la entrega del pedido.
* El pedido podrá pagarse de una vez o en varios pagos.
* Las formas de pago posible son
  + Tarjeta de crédito (fecha de caducidad, número, VISA o MASTERCARD)
  + Efectivo (moneda)
  + Cheque (nombre, entidad bancaria)
* El pedido podrá estar en uno de los siguientes estados: pendiente, pagado, procesando, enviado y entregado.

**Ejercicio 19. Gestión de series de TV.**

Representa mediante un diagrama de clases la siguiente especificación relacionada con un sistema para gestionar series.

* Las series se caracterizan por su título, año de inicio, sinopsis, género al que

pertenece (acción, aventura, animación, comedia, documental, drama, horror, musical, romance, ciencia ficción) y personajes que intervienen.

* Las series se organizan en temporadas ordenadas que tienen una fecha de

producción y una fecha de estreno de televisión a nivel mundial.

* Cada temporada está a su vez formada por capítulos ordenados que tienen un título, una duración y una sinopsis.
* Un personaje en una serie concreta es interpretado por un único actor, pero un actor puede interpretar varios personajes en una misma serie.
* Un personaje interpretado por un actor puede aparecer en más de una serie.
* Además, un personaje puede no aparecer en todos los capítulos de la serie por lo que el sistema debe conocer en qué capítulos aparece un personaje.

**Ejercicio 20. Miembros comunidad universitaria.**

Se propone realizar un modelo simplificado de los distintos miembros de la comunidad

universitaria. Todos los miembros de la comunidad universitaria se caracterizan por un nombre y un D.N.I. Los miembros se dividen en estudiantes o personal de la universidad. Todos los estudiantes tienen un número de identificación asociado: el nie.

En cuanto al personal, todos tienen un salario asignado y a su vez estos pueden ser personal docente investigador (pdi) ó personal de administración y servicios (pas). Los pdi tienen asignada una asignatura que impartir (se identificará por el título) y los pas un edificio donde trabajan (se identificará por el nombre del edificio). Además de los anteriores, existen los doctorandos que son a la vez pdi y estudiantes. Los doctorandos se caracterizan por el título de la tesis doctoral sobre la que investigan.

Se pide que, utilizando herencia siempre que se pueda, se realice un diseño UML de las clases Miembro, Personal, Estudiante, Pdi, Pas y Doctorando.

**Ejercicio 21. Sistema de gestión bibliotecaria.**

Se desea realizar el modelo de un sistema de gestión bibliográfica. Para ello se consideran los siguientes conceptos:

• Hay “Bases\_de\_datos” que tienen un atributo que es su “url”, y un método “anadir\_publicación” (que devuelve un puntero a “Publicación” y tiene un parámetro que es un string).

• Las bases de datos se componen de “Publicaciones”.

• Una publicación tiene dos atributos: “nombre” (de tipo string) y “fecha” (de tipo Date –

suponerla ya implementada).

• Hay varios tipos de publicaciones: “Articulo\_de\_conferencia”, “Capitulo\_de\_libro” y “Libro”. A su vez hay un tipo de libro que son los “Proceedings”, que son libros compuestos por varios artículos de una misma conferencia. Los artículos de conferencia tienen un atributo: “num\_paginas” (int). Los capítulos de libro tienen un atributo: “num\_capitulo” (int). Los libros tienen un atributo: “ISBN” (array de 4 enteros). Los proceedings tienen un atributo: “lugar\_conferencia” (string).

• Los libros pueden contener capítulos de libro.

• Los proceedings están compuestos por artículos de conferencia.

•También hay “Autores”, que tienen tres atributos: “nombre” (string), “correo\_electronico”

(string) y “entidad” (string), y un método: “añadir\_entidad” (no devuelve nada, y tiene un

parámetro de tipo string).

• Los autores pueden escribir artículos y consultar bases de datos.

Utilizando herencia siempre que se pueda, realizar un diseño UML de clases del sistema

Indicando las clases que se consideren necesarias con sus atributos y métodos.

**Ejercicio 22. Comida rápida.**

Un restaurante de comida rápida quiere desarrollar un software para gestionar los pedidos y repartos a domicilio, así como obtener estadísticas de los mismos. Se quiere registrar la siguiente información:

● Los pedidos para consumir en el establecimiento se recogen en la caja y los de entrega a domicilio se recogen por teléfono.

● Todos los pedidos van numerados correlativamente. Por cada uno de ellos se recoge la fecha y hora del pedido.

● En los pedidos de caja se registra el número de caja.

● En los pedidos a domicilio se registra el teléfono, población y dirección del cliente.

● Se cuenta con una serie de repartidores identificados por un número, con el nombre completo, una matrícula de moto y un turno (mañana, tarde, noche).

● A cada pedido a domicilio se le asigna un repartidor.

● El establecimiento ofrece una serie de productos (hamburguesa simple, doble, alemana, Coca-Cola pequeña mediana, grande, ensalada, etcétera) identificados por un código y con un nombre y precio.

● Se ofrecen también menús especiales (combos u ofertas) compuestos por un conjunto de productos. Los menús están numerados, tienen un nombre de menú y un precio.

● Un pedido puede constar de varios menús en diferentes cantidades y/o varios productos en diferentes cantidades.

**Ejercicio 23. Alquiler de vehículos.**

Se pretende desarrollar un sistema de gestión para una empresa de alquiler de vehículos cuyas principales funcionalidades y características son las que a continuación se describen:

* La empresa se dedica al alquiler de vehículos. Dentro de los automóviles, la empresa distingue entre automóviles de lujo y básicos.
* Un cliente puede realizar reservas de automóviles desde Internet. Para ello accederá a la opción “Realizar Reservas” y el sistema le pedirá su login y password.

Si no ha sido cliente anteriormente el usuario deberá acceder a la opción “Registrarse”, en la que el sistema le solicitará sus datos personales. Una vez finalizado el registro podrá realizar la reserva. Para ello el sistema le preguntará si desea un automóvil de lujo o básico, y una vez que el cliente le ha indicado lo que desea, el sistema le pedirá las fechas iniciales y final para las que desea el alquiler, así como las ciudades de recogida y devolución del vehículo. Después le mostrará la relación de automóviles (marca, modelo, motor y precio diario) disponibles en esas fechas,

entre las que el cliente escogerá uno. La reserva queda realizada e identificada con un código del que el cliente debe tomar nota. El cliente podrá, posteriormente, acceder al sistema para consultar o cancelar su reserva, previa identificación en el sistema.

* Para recoger el vehículo el cliente debe presentarse en las oficinas de la empresa para firmar el contrato de alquiler del vehículo. Un empleado será el encargado de registrar ese contrato en el sistema, y se considera imprescindible que quede registrado qué empleado ha realizado cada contrato. El empleado accederá a la parte de “Contratos” y, si existe reserva previa (no siempre tiene por qué haberla) introducirá el código de reserva que debe de facilitarle el cliente.
* El sistema comprueba la existencia de la reserva y genera un contrato tipo para el tipo de vehículo reservado.
* El gerente de la empresa accede al sistema para realizar diversas consultas e informes sobre cuestiones como alquileres realizados, vehículos disponibles, contratos firmados por empleados, etc. Por último, existe un administrador, que es el encargado de registrar y mantener la información de los usuarios y de los vehículos de la empresa.