

EJERCICIO 1

Se quiere hacer el diseño de un robot modular. El robot estará compuesto por varios módulos entre los que se encuentran: rotación, extensión, helicoidal, cámara. Los módulos podrán ser dinámicos (capaces de moverse: rotación, extensión, helicoidal) o estáticos (no se pueden mover: cámara).

Los módulos tendrán un identificador (1-255) y unas dimensiones (largo, ancho y alto, entre 1 y 200mm). Los módulos estarán compuestos de un sistema de control y un sistema de comunicación. Los módulos dinámicos tendrán:

- motores (1 ó 2).
- Un parámetro que es el tipo de movimiento que pueden realizar.
- Una función que es moverse (con parámetro el tipo de movimiento).

Los módulos estáticos podrán tener sensores (de 0 a 5).

El sistema de control utiliza el sistema de mensajes para comunicarse. Los módulos pueden enviar y recibir mensajes de/hacia el usuario y otros módulos, con un parámetro que es un array de datos a mandar o recibir. También utiliza los motores para moverse y los sensores para captar información del medio.

Se pide que utilizando herencia siempre que se pueda, se realice un diseño UML de las clases necesarias para representar todas las entidades del sistema, indicando atributos y métodos, así como las relaciones existentes entre las clases.

EJERCICIO 2

Se trata de realizar un diagrama de clase para representar las relaciones entre empresa, empleados y clientes. Utilizaremos asociaciones de composición y generalización en el diagrama. Los requisitos son los siguientes:

- La empresa se compone de clientes y de empleados. Utilizaremos para estas relaciones asociaciones de composición.
- Los datos de la empresa son: CIF, razón social, dirección y teléfono.
- Los datos de los clientes son: código del cliente, nombre, fecha de nacimiento, teléfono, empresa para la que trabaja y comisión.
- Los datos de los empleados son: código de empleado, nombre, fecha de nacimiento, teléfono, fecha de alta en la empresa y salario.
- Como los atributos nombre, fecha de nacimiento y teléfono son comunes para clientes y empleados se creará una clase persona para esos atributos, y las clases cliente y empleado heredarán la clase persona.
- Un empleado puede ser director de varios empleados. De este director se necesita saber también la categoría y la fecha de alta como director. El director heredará de empleado y además tendrá una asociación [1 ..*] con empleado.
- Para todas las clases crearemos dos métodos, uno para asignar datos a los atributos (como si fuese el constructor), este método entrará tantos parámetros como atributos, y otro que se llame obtener que devolverá un objeto de la misma clase.

EJERCICIO 3

Se desea realizar el análisis de un sistema de gestión informática de una pequeña agencia de viajes que oferta viajes a sus clientes. El sistema debe proporcionar una ventana inicial con una serie de menús que abrirán paso al resto de ventanas de la aplicación que permitirán realizar las siguientes acciones :

- La gestión de las reservas de viajes para realizar reservas , modificar reservas, consultar reservas , borrar reservas y generar e imprimir facturas.
- El mantenimiento de datos de clientes, para mantener actualizados los datos se realizarán operaciones de consulta, altas, bajas y modificaciones de datos de clientes, y además debe permitir generar listados de clientes.
- Mantenimiento de datos de viajes, para mantener actualizados los datos se realizarán operaciones de consultas , altas , bajas , modificaciones e informes de viajes.

Disponemos de una base de datos donde están almacenados los datos de los clientes, los viajes, las reservas , las fechas de viaje, los datos son los siguientes:

- Datos de clientes son: código-cliente, nombre , teléfono y dirección.
- Datos de viajes son: código, nombre , plazas y precio.
- Datos de las reservas son: número de reservas y estado de la reserva.
- Un cliente puede realizar muchas reservas , y una reserva es de un cliente.
- Igualmente , de un viaje se pueden realizar muchas reservas , y una reserva pertenecerá a un viaje.
- Los viajes se ofertan en varias fechas de viaje, de estas fechas se necesita saber la fecha de comienzo y la fecha de fin . Estas fechas pueden ser compartidas por varios viajes.
- También se cuenta con la información de un catálogo de viajes, datos del catálogo son: código, destino, procedencia, temporada, precio. Los viajes se crean a partir del catálogo.

EJERCICIO 4

Se desea realizar el análisis de un sistema de gestión informática de una pequeña academia dedicada a organizar cursos de formación. El sistema debe proporcionar una ventana inicial con una serie de menús que abrirá paso al resto de ventanas de la aplicación . Las ventanas deben permitir realizar las siguientes acciones:

- La gestión de la matrícula de alumnos en los cursos, para dar de alta, eliminar, modificar o consultar datos de matriculación.
- La gestión de cursos, para mantener actualizados los datos de cursos se realizarán operaciones de consultas, altas, bajas y modificaciones , y además generar listados de cursos, y alumnos de cursos.

Disponemos de una base de datos donde están almacenados los datos de los cursos, los alumnos, y las matrículas. Los datos son los siguientes:

- Los datos de curso son: código, curso, denominación, horas y precio.
- Los datos de alumnos son: código alumno, nombre, fecha de nacimiento y nivel.
- Los datos matrícula son: el número de matrícula y la fecha.
- Un alumno se puede matricular en varios cursos es decir podrá tener varias matrículas.
- Los cursos son de dos tipos: de idiomas y de tecnologías. De los cursos de idiomas necesitamos saber datos de especialidad y laboratorio. De los de tecnología necesitamos saber el equipamiento y el presupuesto.

Se pide:

- Identificar las clases de diseño y sus relaciones: clases del tipo Entidad, de Control e Interfaz.
- Identificar los atributos y las operaciones, incluir solo los atributos y operaciones que se especifican en el enunciado.
- Identificar los paquetes: un paquete para las clases Entidad, otro para las clases de Control y otro para las clases Interfaz.
- Realizar el diagrama de clases y de paquetes .
- Añadir la relación de dependencia entre los paquetes.
- Obtener la estructura del código. (ingeniería inversa).

EJERCICIO 5

Especificar un diagrama de clases que describa redes de ordenadores. Los elementos que se pueden incluir en la red son:

- Servidor, PC, Impresora.
- Hub, Cable de red.

Los PCs pueden conectarse con un único Hub, los servidores con uno o varios.

Los Servidores y PCs pueden generar mensajes, con una cierta longitud.

Los Hubs tienen un número de puertos, algunos de los cuales puede usarse para conectar con otros Hubs. Tienen cierta probabilidad de “perder” mensajes.

Las impresoras pueden averiarse, con cierta probabilidad, durante cierto tiempo.

EJERCICIO 6

Una biblioteca tiene copias de libros. Estos últimos se caracterizan por su nombre, tipo (ingeniería, literatura, informática, historia ...), editorial, año y autor.

Los autores se caracterizan por su nombre, nacionalidad y fecha de nacimiento.

Cada copia tiene un identificador, y puede estar en la biblioteca, prestada, con retraso o en reparación.

Los lectores pueden tener un máximo de 3 libros en préstamo.

Cada libro se presta un máximo de 30 días, y por cada día de retraso, se impone una “multa” de dos días sin posibilidad de coger un nuevo libro.

Realiza un diagrama de clases y añade los métodos necesarios para realizar el préstamo y devolución de libros.

Realiza un diagrama de casos de usos.

EJERCICIO 7

Representa mediante un diagrama de clases la siguiente especificación.

Una aplicación necesita almacenar información sobre empresas, sus empleados y sus clientes. Ambos (empleados y sus clientes) se caracterizan por su nombre y edad. Los empleados tienen un sueldo bruto, 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.