

ANTEPROYECTO

Videoclub:

ArrayList (estructuras de datos):

Mi proyecto consistiría en un videoclub en el que se almacenan productos que son vídeos, audios, videojuegos y revistas mediante arrays.

Enumeraciones:

Enumeración ContenidoVideo: películas, series, cortos.

Enumeración ContenidoAudio: música, narraciones de libros, CDs para aprender idiomas.

Enumeración ContenidoVideojuego: acción, aventura, de baile.

Enumeración ContenidoRevista: videojuegos, cine, música, comic.

Enumeración Director:

Los vídeos pueden tener distintos directores

Enumeración Videojuegos:

Los videojuegos pueden haber sido creados por distintas empresas.

Enumeración TituloRevista:

Los distintos títulos de revistas.

Enumeración DiaEntregaAudio:

Los días que se puede alquilar los audios.

Enumeración DiaEntregaVideo:

Los días que se puede alquilar los vídeos.

Enumeración DiaEntregaVideojuego:

Los días que se pueden alquilar los videojuegos.

Herencia:

Clase Padre: Producto.

Clases Hijas: Audio, Video, Videojuego, Revista

Los Productos tienen un código, un nombre, y un precio.

Los Videos tiene: fecha de alquiler, enumeración Director.

Los Audios tienen: fecha de alquiler y atributo autor.

Los Videojuegos tienen: fecha de alquiler y enumeración Empresa.

Las Revistas tienen: atributo ticket.

Funcionalidad:

Los vídeos, audios y videojuegos pueden ser alquilados pero el precio varía dependiendo de si son audios, videos o videojuegos.

Las revistas se pueden comprar, aparece un ticket cuando se hace una compra.

Interfaces:

Se implementará la interfaz Serializable,

Rentable: Para asignar un precio, afecta a todas las clases hijas.

Descontable: Para asignar un descuento dependiendo de la fecha de entrega, afecta a audio, video y videojuego.

FechaAlquilable: Asigna una fecha, afecta a audio, video y videojuego.

Ficheros:

Cada vez que se utilice el programa se podrá crear un fichero, abrir uno nuevo, guardar, guardar como, etc..

Excepciones:

VacioException: Controla que cuando se acceda al almacen no este vacio.

NoEscritoException: Controla que no se introduzcan valores vacios.

AudioExisteException: Controla que el audio exista.

VideoExisteException: Controla que el video exista.

VideojuegoExisteException: Controla que el videojuego exista.

Expresiones regulares:

Cada disco tiene un código para almacenarlo en el almacén ordenadamente en el que podrá ponerse del 1 al 9 y una letra que equivaldría a la estantería en el que está.

Las revistas tienen un código de compra que sería 4 letras en Mayúscula 1 número del 0-9 y una letra de la X-Z, un ej: ABCDOX

Fechas:

Cada vez que se alquile un producto se anotará la fecha del alquiler y la fecha de entrega, además se podrá elegir la fecha en la que se devolverá el disco, dependiendo de la tardanza y de que tipo de disco sea, el precio varía.

GUI:

Tendrá un menú en el que se podrá añadir audios, videos, videojuegos y revistas, eliminarlos, mostrar los que están en la tienda y mostrar el número de productos que están en la tienda. Luego tendrá un menú acciones en el que se podrá alquilar audios, videos, videojuegos, comprar revistas, mostrar los que estan alquilados, mostrar el número de discos alquilados y devolver audios, videos y videojuegos. Y por último el menú ayuda.