

Práctico 4 - OOP

Introducción:

Se solicita al estudiante realizar programas que respeten la Teoría de la Programación Orientada a Objetos POO.

El código fuente debe estar comentado para indicar cuál es la acción o intencionalidad que se ha querido realizar en alguna parte del código fuente.

Los programas deben reflejar el siguiente manejo de temas:

- ⇒ Creación de clase principal con atributos: públicos, protegidos y privados.
- ⇒ Creación de Clases hijas.
- ⇒ Herencia de la clase padre a las clases hijas.
- ⇒ Creación de un constructor bien realizado y coherente.
- ⇒ Creación de métodos.
- ⇒ Manejo de sobrecarga de métodos.
- ⇒ Ingreso de datos por el usuario desde el teclado.
- ⇒ Información por pantalla fluida y clara para el usuario.
- ⇒ Se deben guardar con el formato apellido_nombre_tpnn_ejnn.extension

1. ELECTRODOMÉSTICOS

Contexto del Problema: se trata de una empresa comercial que vende productos electrodomésticos y necesita en su sistema informático llevar la información de los televisores y celulares. En el programa el operador debe consultar información y características generales y propias de cada producto. Luego de leer y analizar lo que se pide, diseñe el diagrama de clases correspondiente.

Crear una clase padre llamada **Producto** con los atributos: *nombre*, *precio*, *nro_garantia*, *nro_fabrica* y *stock*.

Los atributos deben ser:

- nombre (*público*)
- precio (*público*)
- nro_garantia (*público*)
- nro_fabrica (*protegido*)
- stock (*protegido*)

Y los métodos: *get_nro_fabrica()* que sirve para leer el numero de nro_fabrica, *get_stock()* que sirve para leer el numero de stock.

Crear una sobrecarga de métodos llamada: *get_info()* que muestre la siguiente información: clase.nombre - clase.precio - clase.nro_garantia - clase.nro_fabrica - clase.stock

Crear una clase hija: llamada **Televisor** (*que herede de la clase Producto*), pero agregar como atributo propio el atributo llamado pulgadas, el resto de los atributos los hereda de la clase padre Producto. Para la clase Televisor se debe crear el método llamado *get_info()*:

cuyo objetivo es obtener información adicional del televisor.

Deberá mostrar la siguiente información:

clasehija.nombre - clasehija.precio - clasehija.nro_garantia - clasehija.nro_fabrica - clasehija.stock - clasehija.pulgadas

Crear una clase hija: llamada **Celular** (*herede las características de la clase principal Producto*). Los atributos de la clase Celular son los mismos que la clase padre, salvo que a la clase Celular se le agrega el atributo de nombre "marca". Para la clase Celular se debe crear el método llamado `get_info()`: cuyo objetivo es obtener información adicional del celular.

Deberá mostrar la siguiente información:

clasehija.nombre - clasehija.precio - clasehija.nro_garantia - clasehija.nro_fabrica - clasehija.stock - clasehija.marca

2. BIBLIOTECA

Contexto del Problema: Se sitúa en una biblioteca de una institución.

Consigna: Crea un programa en Python que modele una biblioteca. La biblioteca debe tener dos clases: Libro y Biblioteca. Luego de leer y analizar lo que se pide, diseñe el diagrama de clases correspondiente.

La clase Libro debe tener los siguientes atributos: título, autor, año de publicación y número de páginas.

La clase Biblioteca debe tener una lista de libros y los siguientes métodos:

agregar_libro(libro), *eliminar_libro(libro)*, *buscar_libro(titulo)*, y *mostrar_libros()*.

El método *agregar_libro(libro)* debe agregar un objeto de la clase Libro a la lista de libros de la biblioteca.

El método *eliminar_libro(libro)* debe eliminar un objeto de la clase Libro de la lista de libros de la biblioteca. El método *buscar_libro(titulo)* debe buscar un libro en la lista de libros de la biblioteca por título y devolver el objeto de la clase Libro correspondiente. El método *mostrar_libros()* debe mostrar por pantalla todos los libros de la biblioteca con su información completa (*título, autor, año de publicación y número de páginas*).

3. TIENDA ONLINE

Contexto del Problema: Se debe crear un sistema informático para una tienda online que vende productos físicos y productos digitales (*como por ej. software, etc.*). Luego de leer y analizar lo que se pide, diseñe el diagrama de clases correspondiente.

Consigna: Se debe crear un programa en Python que modele una tienda en línea. La tienda debe tener dos clases: Producto y Tienda.

La clase **Producto** debe tener los siguientes atributos: *nombre, precio, descripcion y cantidad*.

Además, se debe crear una jerarquía de clases con una clase principal y dos clases hijas. La clase principal es Producto y las dos clases hijas son ProductoDigital y ProductoFisico. La clase **ProductoDigital** debe tener un atributo adicional llamado *formato* y

sobrescribir el método *precio_total()* de la clase **Producto** para aplicar un descuento del 10 % en el precio total.

La clase **ProductoFisico** debe tener un atributo adicional llamado *peso* y sobrescribir el método *precio_total()* de la clase **Producto** para agregar un costo de envío de \$5 por cada 100 gramos de peso.

La clase **Tienda** debe tener una lista de productos y cuatro métodos: *agregar_producto(producto)*, *eliminar_producto(producto)*, *buscar_producto(nombre)* y *mostrar_productos()*.

El método *agregar_producto(producto)* debe agregar un objeto de la clase **Producto**, **ProductoDigital** o **ProductoFisico** a la lista de productos de la tienda. El método *eliminar_producto(producto)* debe eliminar un objeto de la clase **Producto**, **ProductoDigital** o **ProductoFisico** de la lista de productos de la tienda.

El método *buscar_producto(nombre)* debe buscar un producto en la lista de productos de la tienda por nombre y devolver el objeto de la clase **Producto**, **ProductoDigital** o **ProductoFisico** correspondiente. El método *mostrar_productos()* debe mostrar por pantalla todos los productos de la tienda con su información completa (*nombre*, *precio*, *descripción*, *cantidad* y *atributos adicionales de cada clase*).

Además, se debe crear al menos un objeto de cada subclase y agregarlos a la tienda. Finalmente, se debe buscar un producto en la tienda por nombre, eliminar un producto de la tienda y mostrar todos los productos de la tienda después de eliminar un producto.

4. GESTION DE ALUMNOS Y PROFESORES

Contexto del Problema: Se debe crear un sistema informático para la gestión de profesores y estudiantes de un Instituto Educativo. Luego de leer y analizar lo que se pide, diseñe el diagrama de clases correspondiente.

Consigna: Crear la clase principal **Persona** con los atributos públicos *nombre*, protegidos *edad* y privados *direccion* y *telefono*.

Crear un método llamado *imprimir_info* cuyo objetivo es imprimir la información de la persona.

Crear una clase hija llamada **Estudiante** que hereda de la clase **Persona**. Dicha clase hija contiene todos los atributos de la clase padre, pero se le debe agregar un atributo específico de tipo publico llamado: *carrera*

Crear una clase hija llamada **Profesor** que también hereda de la clase **Persona**. Dicha clase hija contiene todos los atributos de la clase padre, pero se le debe agregar un atributo específico de tipo publico llamado: *especialidad*

En ambas clases hijas se deben agregar los métodos *imprimir_info()* respectivamente. Agregar una sobrecarga de métodos que tenga como objetivo llamar al método *imprimir_info()* de la clase **Persona** desde las clases hijas y agregar la información adicional.