Programación Orientada a Objetos

Práctica Nº 2 – 2018

Implementar en JAVA cada uno de los siguientes problemas

1. Persona

Implementar una clase llamada Persona con las siguientes características:

Atributos: nombre, edad, fecha de nacimiento, DNI, sexo, peso y altura

Valores por defecto:

- El DNI es un valor obligatorio, no posee valor por defecto.
- En caso de la fecha de nacimiento será el 1 de enero de 2000.
- Sexo será Femenino por defecto.
- El nombre por defecto es N y el apellido es N.
- El peso y la altura son 1 por defecto.

Acorde a lo anterior se deben crear los constructores. Con DNI, con DNI nombre y apellido, con DNI nombre apellido y Fecha de nacimiento…..

La **responsabilidad** de la clase es la siguiente:

- Cálculo del índice de masa Corporal. Una persona sabe calcular cuál es su índice de masa corporal, el cual es peso / altura^2
- Saber si está en forma. Está en forma si el índice de masa corporal se encuentra entre 18,5 y 25 (esto es solo un ejemplo para probar la funcionalidad).
- Saber si está cumpliendo años.
- Saber si es mayor de edad. Una persona es mayor de edad si es mayor de 18 años
- Métodos set de cada parámetro, excepto de DNI.
- Saber si puede votar. Es necesario ser mayor de 16 años.
- Saber si es coherente. Que la fecha de nacimiento coincida con la edad.
- Mostrar toda la información del objeto. Es decir devolver un String con la información del mismo.

2. Electrodomésticos

Crear una clase Electrodoméstico con las siguientes características:

Atributos: nombre, precio base, color, consumo energético y peso.

Valores por defecto:

- El color por defecto es gris plata.
- El consumo energético 10 Kw.
- Precio base 100 pesos.
- El peso es 2 kg.

Implementar todos los constructores.

La **funcionalidad** de la clase es la siguiente:

- Todos los métodos get y set.
- Comprobar si el electrodoméstico es de bajo consumo (menor que 45 Kw)
- Calcular el balance, el mismo es el resultado de dividir el costo por el peso

- Indicar si un producto es de alta gama. En un producto de alta gama el balance es mayor que 3

3. Elementos Geométricos

Implementar la clase punto geométrico, la cual posee dos valores de coordenada X, Y. El valor por defecto de los mismos es (0,0)

La **funcionalidad** que posee la clase es la siguiente:

- Desplazar el punto en el plano. Se encarga de trasladar el punto a otra posición del plano. Para esto se incrementan los valores de X e Y en una cierta cantidad (cierto desplazamiento en X y cierto desplazamiento en Y).
- Calcular la distancia euclídea. Dado un punto es posible establecer la distancia euclídea con otro punto acorde a la siguiente fórmula:

$$Distancia^2 = (X_1 – X_2)^2 + (Y_1 – Y_2)^2$$

La clase **Math** de Java define un método **sqrt** para el cálculo de la raíz cuadrada.

Implementar la clase Rectángulo.

Para esta clase es necesario definir el rectángulo utilizando los puntos como vértices. Se trabajará con Rectángulos cuyos lados estén paralelos a los ejes.

Nota. Definir la estructura que debe poseer un rectángulo y en base a esto implementar los atributos de la clase.

La **funcionalidad** que debe proveer un rectángulo es la siguiente

- Desplazarlo en el plano. Trasladar el rectángulo acorde a ciertos valores de X e Y.
- Calcular el Área del rectángulo.
- Compararlo con otro rectángulo. Devolver 1 si el rectángulo es mayor, 0 si son iguales y -1 si es menor. Se dice que un rectángulo es mayor que otro si el área del mismo es mayor que la del otro.
- Poder invertir el rectángulo. Si se toma un vértice, invertirlo es cambiar el sentido del rectángulo, si estaba hacia arriba, ahora está hacia abajo y si estaba hacia adelante está hacia atrás. El tamaño del rectángulo debe ser el mismo.
- Determinar si el rectángulo es un cuadrado, es decir, decidir si se cumplen las propiedades para que sea un cuadrado.
- Determinar el largo del lado superior.
- Determinar si esta acostado o parado (si el alto es mayor que el ancho).

4. Series

Una Serie está formada por un conjunto de temporadas, cada una de las cuales tiene una cantidad de episodios. Cada episodio posee un título, una descripción, un flag indicando si el usuario ya vio el episodio y una calificación dada por el usuario (con valores de 1 a

5). Si el usuario no vio un episodio particular, la calificación dada será un valor negativo.

Las series poseen como atributos (además de los episodios correspondientes) un título, una descripción, un creador y un género.

Implementar en Java las clases involucradas, determinar qué clase es responsable de responder los siguientes servicios e implementarlos en Java.

- Ingresar la calificación de un episodio. Si el valor ingresado como calificación no es correcto imprimir un mensaje por pantalla y no cambiar el valor anterior.
- Obtener el total de episodios vistos de una temporada particular.
- Obtener el total de episodios vistos de una serie.
- Obtener el promedio de las calificaciones dadas por el usuario para una temporada particular.
- Obtener el promedio de las calificaciones dadas por el usuario para una serie.
- Determinar si el usuario ya vio todos los episodios de la serie.

Nota. Para calcular los promedios, tener sólo en cuenta los episodios vistos por el usuario.

5. Agenda Personal y Contactos del Celular

Implementar los ejercicios 1 y 10 del práctico 1. Identificar la funcionalidad de cada uno de los objetos. Determinar qué objetos pueden ser compartidos en ambos ejercicios y qué objetos no.

6. Clase Rectángulo

A la clase Rectángulo del ejercicio 3 implementarla nuevamente, guardando solo un vértice y las longitudes de los lados. Implementar la misma interfaz que en el ejercicio 3.

Nota. Tener en cuenta la dirección de los lados. Proveer dos constructores, uno con los 4 puntos y otro con los datos anteriores.

7. Estructuras de datos

Plantee la solución identificando objetos que intervienen, su estado y su comportamiento para las siguientes estructuras de datos:

Grafo dirigido y no dirigido, con algoritmos de recorridos BFS, DFS.

Árbol N-ario y Árbol Binario, incluir soporte para con recorridos en orden, pre-orden y postorden.

Matriz de n dimensiones.