0485.- Programación.

<u>Área personal</u> / Mis cursos / <u>Programación</u> / <u>U5 - POO. Herencia. Interfaces.</u> / <u>U5: Tarea 2 - Interfaces</u>

U5: Tarea 2 - Interfaces

Actividad 1: Implementar la interfaz Pila. Declararla e implementarla en las clases PilaTabla y PilaLista. Utilizando una referencia del tipo Pila con cualquiera de las implementaciones (es decir: *Pila p=new PilaLista()* o *Pila p=new PilaTabla()*), mostrar por pantalla, en orden inverso, 10 números introducidos por teclado.

Actividad 2: Diseñar la clase Cliente con los siguientes atributos: dni, nombre, edad y saldo. Implementar un constructor y los métodos toString() y equals() (este último basado en el dni). Implementar la interfaz Comparable con un criterio de ordenación basado también en el dni. Implementar también un comparador para hacer ordenaciones basadas en el nombre y otro basado en la edad. Crear una tabla con 5 clientes y mostrarlos ordenados por dni, por nombre y por edad.

Actividad 3: Crear la interfaz Figura, con el método: getArea()

Crear las clases PoligonoRegular y Circulo. Ambas implementan dicha interfaz.

La clase PoligonoRegular tiene dos clases hijas o subclases: Cuadrado y Triangulo. No se podrán crear objetos de la clase PoligonoRegular, pero sí de tipo Cuadrado y de tipo Triangulo.

- **PoligonoRegular** tiene como atributos uno llamado *lado* (indica lo que mide el lado en cms), y otro llamado *color*, cuyos valores pueden ser: azul, rojo, verde o amarillo.

El atributo color podrá ser visto desde cualquier clase. El atributo lado, solo podrá ser visto por clases hijas y vecinas.

La clase PoligonoRegular tendrá también un atributo llamado contadorPoligonos que cuente el número de polígonos que se van creando. Es decir, cuando se cree un Cuadrado o un Triángulo, deberá incrementarse.

- En la clase Cuadrado se implementará el método getArea(), que devolverá el valor de su área:

$$\acute{A}rea = a^2$$

siendo a un lado del cuadrado

- En la clase **Triángulo** tendremos además el atributo *altura*, y se implementará el método getArea(), que devolverá:

$$\acute{A}rea = \frac{b \cdot h}{2}$$

donde b es la base y h es la altura

- Por defecto, la forma de ordenación de objetos Triangulo es la ordenación por lado. Es decir, un triángulo se considera menor que otro si su lado es menor.
- Queremos tener también la opción de poder comparar objetos Triángulo según el orden alfabético de su color. (amarillo es el color más pequeño, luego azul,...).
- La clase **Circulo** tendrá el atributo *radio*, y su método getArea calculará el área como:

$A = \Pi \cdot R^2$

- Implementar el método necesario en las clases instanciables (Cuadrado, Triangulo, Circulo) para que los objetos de ese tipo puedan ser parámetro de System.out.Println(o).
- Cada clase instanciable tendrá un constructor que recibirá todos los valores de sus atributos como parámetro, y otro por defecto que no recibirá nada y pondrá el atributo lado y/o altura, según corresponda, a 10, y el color, a azul.

Ahora crea una clase ejecutable que realice lo siguiente:

- Crea un array de FigurasGeometricas de 5 posiciones.
- Deberás mostrar el contenido del array. (Fíjate en que se muestren los atributos correctos dependiendo de si es un Triangulo o un Cuadrado)

- Muestra ahora el valor de la variable contadorPoligonos.
- Crea después un array de objetos Triangulo, y ordena el array con su ordenación por defecto. Muestra el array.
- Ordénalo ahora por color. Muéstralo.

Estado de la entrega

Estado de la entrega	No entregado
Estado de la calificación	Sin calificar
Última modificación	-
Comentarios de la entrega	► Comentarios (0)

Agregar entrega

Todavía no has realizado una entrega

■ Lectura Interfaces

Ir a...

U5: Tarea 3 - Ampliación ►

Usted se ha identificado como Sergio Bejarano Arroyo (Cerrar sesión) Programación

Español - Internacional (es) English (en)

Español - Internacional (es)

Resumen de retención de datos

Descargar la app para dispositivos móviles