

Tarea 4 – Interfaces

Enunciado

1. Responde a las siguientes preguntas:
 - a. ¿Puede una interfaz incluir variables de instancia?
 - b. ¿Qué diferencias encontramos entre herencia e interfaz?
 - c. ¿Por qué los métodos en una interfaz no pueden ser privados?
 - d. ¿Podrían las interfaces contener varias definiciones de la misma constante?
 - e. ¿Puede una clase heredar de varias interfaces?
 - f. Si una clase hereda de varias interfaces, y esas interfaces tienen un mismo método con la misma signatura, ¿se producirá un error? ¿por qué?
2. Utilizando los valores por defecto de las interfaces y una correcta estructuración, crea una interfaz equivalente a la siguiente:

```
public interface Interfaz {  
    public double methodB();  
    public static final int aConstant = 32;  
    public static final double pi = 3.14159;  
    public void methodA( int x );  
}
```

3. ¿Es correcta la siguiente interfaz? Comenta los posibles errores que hayan.

```
public interface Interfaz {  
    public final int x = 32;  
    public double y;  
    public double addup();  
    public boolean addup();  
}
```

4. Decide qué tipo de implementación (Clase padre, hija o interfaz) sería la correcta para cada uno de los siguientes conceptos de una aplicación sabiendo que:
 - a. Bienes, es una clase con dos atributos: descripción y precio(double).
 - b. Comida: con un atributo calorías double y el método `toString`. Los objetos de comida no están sujetos a impuestos.
 - c. Juguete: con un atributo edad mínima, método `toString`. Los objetos de juguete están sujetos a impuestos.
 - d. Libro: con un atributo autor, método `toString`. Los objetos de libros están sujetos a impuestos.
 - e. Imponible: hay muchas cosas que están sujetas a impuestos que no son bienes, como los servicios o el entretenimiento. Además, no todos los bienes están sujetos a impuestos. Así que queremos tener el concepto imponible como un concepto separado, no como parte del concepto de Bienes. Imponible tiene una tasa de impuestos del 6 por ciento (double), y tiene un método de cálculo de impuestos.
 - f. No añadas getters ni setters en ninguna clase, pero si constructores.
5. Implementa las clases del ejercicio anterior y añade un `main` para mostrar los cálculos de impuestos e imprimir los objetos con ayuda del método `toString`, para ello crea objetos de las clases Bienes, Comida, Juguete y Libro.

- a. Pega tú código.
- b. ¿Podrían algunos de los 4 objetos del main mantenerse en un array?
- c. ¿Cuál sería el tipo del array? Modifica tu main y crea un array con objetos de ese tipo solo.
Copia el código.
- d. Al imprimir el array de forma:

```
System.out.println(array[0]);  
System.out.println(array[1]);
```

¿Obtendremos el mismo resultado en las

- sentencias anteriores? ¿Se llama al mismo método?
- e. Vuelve a cambiar el código, ¿hay algunos objetos que podrían haber sido declarados usando Imponible? Modifica y pega el código.
 - f. Si intentamos imprimir esos objetos con "objeto1.toString()", funcionará? ¿A qué método estaremos llamando?
 - g. Añade el método getAutor() en la clase Libro e intenta imprimir el autor del libro en el main que tenemos con objeto1.getAutor(). ¿Obtienes algún error? ¿Por qué? ¿Es correcto el error? ¿Cómo podemos corregirlo?

Entrega

- Sube un PDF con las respuestas y el código copiado.