1. ¿Qué invonveniente identificas en la siguiente pieza de código y cómo lo mejorarías (REFACTORING)? (Escribir el código mejorado):

void imprimeFactura() {

imprimeEncabezado();

//imprime los detalles

System.out.println ("Nombre: " + nombre );

System.out.println ("Cantidad " + getCantidad());

}

Lo que noto en este problema es como obtener el nombre sin usar su setter

void imprimeFactura() {

imprimeEncabezado();

//imprime los detalles

System.out.println ("Nombre: " + getNombre());

System.out.println ("Cantidad " + getCantidad());

}

Pero yo personalmente lo haría de tal forma que el imprime factura imprima por mas de un solo valor, ya que como se tenia actualmente solo imprimia por un producto.

void imprimeFactura(HashMap<String,String> map) {

imprimeEncabezado();

//imprime los detalles

map.forEach( (k,v)->

System.out.println ("Nombre: " + k + "\nCantidad: " + v )

);

}

1. A continuación, se presenta la descripción de una tabla de amortización por método alemán, el objetivo es contruir la lógica de la tabla y una interfaz simple para la captura de información de entrada utilizando react o angular. Los datos de entrada serán el monto solicitado, la periodicidad (quincenal, mensual, etc), número de cuotas e interés.

Criterios:

* Se debe consumir vía un servicio REST.
* Se deberá enviar la liga a un repositorio de GitHub con el TEST.
* Se deberá presentar un front sencillo para la entrada de información y visualización.

Información de soporte:

<https://www.pichincha.com/portal/blog/post/tabla-amortizacion-francesa-alemana-diferencias>

<https://www.zonabancos.com/blogs/10-matematica-financiera-aplicada-al-negocio-bancario-14557-sistemas-de-amortizacion-de-prestamos-sistema-aleman.aspx>

Para comprobar el correcto funcionamiento, se adjunta plantilla en excel con la configuración de la tabla alemana.