

Materia: PROGRAMACIÓN III											
Apellido:						Fecha:					
Nombre:						Docente ⁽²⁾ :	Friadenrich / Lippi				
División:						Nota ⁽²⁾ :					
Legajo:						Firma ⁽²⁾ :					
Instancia ⁽¹⁾ :	PP	x	RPP		SP		RSP		FIN		

Se debe realizar una aplicación para dar de ingreso con **imagen del item**.

Se deben respetar los nombres de los archivos y de las clases.

Se debe crear **una clase en PHP** por cada entidad y los archivos PHP solo deben llamar a métodos de las clases.

1era parte

1-

A- (1 pt.) index.php: Recibe todas las peticiones que realiza el Postman y administra a qué archivo se debe incluir.

B- (1 pt.) **TiendaAlta.php:** (por POST) se ingresa Marca, Precio, Tipo ("Impresora" o "Cartucho"), Modelo, Color y Stock (unidades). Se guardan los datos en el archivo de texto **tienda.json**, tomando un **id autoincremental** como identificador (emulado). Si la **marca y tipo** ya existen, se actualiza el **precio** y se suma al **stock** existente. Completar el alta con imagen del producto, guardando la imagen con el nombre y tipo como identificación en la carpeta **/ImágenesDeProductos/2024**.

2-

(1 pt.) **ProductoConsultar.php:** (por POST) Se ingresa Marca, Tipo y Color. Si coincide con algún registro del archivo **tienda.json**, retornar **"existe"**. De lo contrario, informar si no existe el **tipo** o la **marca**. (No hay productos de la marca x / no hay productos del tipo x).

3-

A- (1 pt.) **AltaVenta.php**: (por POST) se recibe el email del usuario y la Marca, Tipo, Modelo y Stock. Si el ítem existe en **tienda.json** y **hay stock**, guardar en **ventas.json** (todos los datos recibidos junto con la fecha, número de pedido y id autoincremental). Se **debe** descontar la cantidad vendida del stock.

B- (1 pt.) Completar el alta de la venta con imagen de la venta (ej: una imagen del usuario), guardando la imagen con el nombre+tipo+modelo+email(solo usuario **hasta el @**) y fecha de la venta en la carpeta **/ImagenesDeVenta/2024**.

2da parte

4-

(2 pts.) **ConsultasVentas.php**: (por GET)

Datos a consultar:

A- La cantidad de productos vendidos en un día en particular (se envía por parámetro), si no se pasa fecha, se muestran los del día de ayer.

B- El listado de ventas de un usuario ingresado.

C- El listado de ventas por tipo de producto.

D- El listado de productos cuyo precio esté **entre dos números ingresados**.

E- El listado de ingresos (**ganancia de las ventas**) por día de una fecha ingresada. Si no se ingresa una fecha, se muestran **los ingresos de todos los días**.

F- Mostrar el **producto más vendido**.

5-

(1 pt.) **ModificarVenta.php** (por PUT)

Debe recibir el número de pedido, el email del usuario, la Marca, Tipo, Modelo y Cantidad. Si existe, se modifica; de lo contrario, informar que no existe ese número de pedido.

3ra parte

6-

(2 pts.) **AltaConjuntos.php** (por POST)

Debe permitir ingresar los detalles de la **impresora (marca, modelo)**, los detalles del **cartucho (marca, modelo)** y una **imagen** del conjunto.

Ambos productos deben existir como productos simples en el archivo **tienda.json**. Una vez revisado que los productos existan, el sistema debe combinarlos para formar un **objeto conjunto (id, marca_impresora, marca_cartucho, modelo_impresora, modelo_cartucho, precio_total, tipo="conjunto")** y almacenarlo como un único registro en el archivo **tienda.json**.

Este nuevo elemento **NO debe modificar los registros existentes, funciona como otra opción de compra**. Este objeto **NO controla el stock**, su stock depende de que haya stock de los productos que lo conforman.

Se debe poder diferenciar un producto simple de un conjunto dentro de la lógica de la aplicación a través del campo **tipo** que comparten ambas entidades.

Completar el alta con imagen del conjunto, guardando la imagen con el nombre de la impresora + nombre del cartucho como identificación en la carpeta **/ImagenesDeConjuntos/2024**.

7-

(1 pt.) **BorrarVenta.php** (por DELETE), debe recibir un número de pedido, se borra la venta (**soft-delete**, no físicamente) y la foto relacionada a esa venta debe moverse a la carpeta **/ImagenesBackupVentas/2024**.

Código obsoleto, copiado y pegado que no tenga utilidad (-1 punto).

Se pueden bajar plantillas de internet o traer código hecho, pero en ningún caso se debe incluir código obsoleto o que no cumpla ninguna función dentro del parcial.

Se tiene que sumar 7 (siete) puntos para lograr un cuatro (4).

Ocho puntos equivalen a un cinco (5),

Nueve puntos equivalen a seis (6),

Diez puntos: ocho (8),

Y si se suman los once puntos la nota será de diez (10).