

Materia: PROGRAMACIÓN III

Apellido:						Fecha:	13/05/24				
Nombre:						Docente ⁽²⁾ :	Friadenrich / Lippi				
División:						Nota ⁽²⁾ :					
Legajo:						Firma ⁽²⁾ :					
Instancia ⁽¹⁾ :	Simulacro PP	x	RPP		SP		RSP		FIN		

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

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.) **HeladeriaAlta.php**: (por **POST**) se ingresa Sabor, Precio, Tipo ("**Agua**" o "**Crema**"), Vaso ("**Cucurucho**", "**Plástico**"), Stock (unidades).

Se guardan los datos en el archivo de texto **heladeria.json**, tomando un id autoincremental como identificador(emulado). Si el nombre y tipo ya existen, se actualiza el precio y se suma al stock existente. completar el alta con imagen del helado, guardando la imagen con el sabor y tipo como identificación en la carpeta **/ImágenesDeHelados/2024**.

2-

(1pt.) **HeladoConsultar.php**: (por **POST**) Se ingresa Sabor y Tipo, si coincide con algún registro del archivo **heladeria.json**, retornar "existe". De lo contrario informar si no existe el tipo o el nombre.

3-

a- (1 pts.) **AltaVenta.php**: (por **POST**) se recibe el email del usuario y el Sabor, Tipo y Stock, si el ítem existe en **heladeria.json**, y hay stock guardar en la base de datos(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 sabor+tipo+vaso+mail(solo usuario hasta el @) y fecha de la venta en la carpeta **/ImágenesDeLaVenta/2024**.

2da parte

4- (1 pts.) **ConsultasVentas.php**: (por GET)

Datos a consultar:

- a- La cantidad de **Helados** vendidos en un día en particular (se envía por parámetro), si no se pasa fecha, se muestran las del día de ayer.
- b- El listado de ventas de un usuario **ingresado**.
- c- El listado de ventas entre dos fechas ordenado por **nombre**.
- d- El listado de ventas por sabor ingresado.
- e- El listado de ventas por vaso **Cucurucho**.

5- (1 pts.) **ModificarVenta.php** (por PUT)

Debe recibir el número de pedido, el email del usuario, el nombre, tipo, vaso y cantidad, si existe se modifica, de lo contrario informar que no existe ese número de pedido.

3ra parte

6- (2 pts.) **DevolverHelado.php** (por POST),

Guardar en el archivo (devoluciones.json y cupones.json):

a- Se ingresa el número de pedido y la causa de la devolución. El número de pedido debe existir, se ingresa una foto del cliente enojado, esto debe generar un cupón de descuento (id, devolucion_id, porcentajeDescuento, estado[usado/no usado]) con el 10% de descuento para la próxima compra.

7- (1 pts.) **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 8 puntos para lograr un cuatro (4).

Nueve puntos equivalen a un cinco (5),

Diez puntos equivalen a seis (6),

Once puntos: siete (7),

Doce puntos a un ocho (8),

Trece puntos equivale a un nueve (9)

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