

Materia: PROGRAMACIÓN III

Apellido:						Fecha:					
Nombre:						Docente ⁽²⁾ :	Madou / 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 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.) **HamburguesaCarga.php**: (por POST) se ingresa Nombre, Precio, Tipo ("simple" o "doble"), Cantidad(de unidades). Se guardan los datos en el archivo de texto **Hamburguesas.json**, tomando un id autoincremental como identificador(emulado) .Sí el nombre y tipo ya existen , se actualiza el precio y se suma al stock existente. completar el alta con imagen de la hamburguesa, guardando la imagen con el tipo y el nombre como identificación en la carpeta **/ImágenesDeHamburguesas**.

2-

(1pt.) **HamburguesaConsultar.php**: (por POST) Se ingresa Nombre, Tipo, si coincide con algún registro del archivo **Hamburguesas.json**, retornar "Si Hay". 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 nombre, tipo y cantidad ,si el ítem existe en **Hamburguesas.json**, y hay stock guardar en la base de datos(con la fecha, número de pedido y id autoincremental) y se debe descontar la cantidad vendida del stock .

b- (1 pt) Completar el alta con imagen de la venta , guardando la imagen con el tipo+nombre+mail (solo usuario hasta el @) y fecha de la venta en la carpeta **/ImágenesDeLaVenta**.

2da parte

4- (1 pts.) **ConsultasVentas.php**: necesito saber :

- a- La cantidad de **Hamburguesas** vendidas en un día en particular, si no se pasa fecha, se muestran las del día de ayer.
- b- El listado de ventas entre dos fechas ordenado por nombre.
- c- El listado de ventas de un usuario ingresado.
- d- El listado de ventas de un tipo ingresado.

5- (1 pts.) **ModificarVenta.php** (por PUT), debe recibir el número de pedido, el email del usuario, el nombre, tipo y cantidad, si existe se modifica , de lo contrario informar.

3ra parte

6- (1 pts.) **borrarVenta.php** (por DELETE), debe recibir un número de pedido, se borra la venta y la foto se mueve a la carpeta /BACKUPVENTAS.

7- (2 pts.) **DevolverHamburguesa.php 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 con el 10% de descuento para la próxima compra.

4ta parte

8- (2 pts.) **AltaVenta.php**, ... (continuación) Todo lo anterior más...

- a- Debe recibir el cupón de descuento (si existe) y guardar el importe final y el descuento aplicado en el archivo.
- b- Debe marcarse el cupón como ya usado

9- (2 pts.) **ConsultasDevoluciones.php**:-

- a- Listar las devoluciones con cupones.
- b- Listar solo los cupones y su estado
- c- Listar devoluciones y sus cupones y si fueron usados

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).