

Tasca M8

Tabla de contenidos:

Tasca M8	1
Tabla de contenidos:	1
Descripció	1
Nivell 1	1
- Exercici 1	1
Nivell 2	2
- Exercici 2	2
Nivell 3	2
- Exercici 4	2
Recursos	2
RESPUESTA/SOLUCION:	3
Funcionamiento del aplicativo:	3
Funcionamiento interno:	4

Descripció

Ha arribat l'hora de posar en pràctica una teoria tan important com la de l'orientació a objectes en PHP. Desenvoluparem en tres fases(que s'acumularan) un projecte bàsic que representarà un inventari de productes que bé podrien ser les compres de la nostra llar. Després, desenvoluparem opcions relacionades amb els productes que comprem, de tal forma que aquest projecte bé podria ser la llavor d'una eina més complexa que ens permetrà obtenir dades rellevants sobre les nostres compres domèstiques.

Quasi res !

Es valorarà com repartiu responsabilitats entre diferents objectes. Sabem que tot el que us proposarem en aquest exercici es pot fer tranquilament en un únic arxiu .php, però la idea és que apliqueu POO ;).

No es valorarà presentació en aquest exercici, però ja sabeu que la vida és molt més bonica amb colors i estils :)

Nivell 1

- Exercici 1

Necessitarem crear una base de dades amb una taula "Compra" que tingui com a atributs:

- **Nom** del producte comprat.
- **Preu** unitari del producte.
- **Quantitat** adquirida del producte.
- **Id** de la fila en la taula.

Podeu crear-al al vostre gust o fer servir l'arxiu adjunt "php_m8_n1.sql".

El fluxe d'execució principal vindrà donat a l'arxiu **index.php** que en aquesta primera fase haurà de :

- **Connectarse** a una base de dades(pots crear una classe per a aïllar aquesta lògica del fluxe principal d'execució).
- **Inserir** al menys 3 productes a la taula "producte" de la base de dades.
- **Crear una classe** Taula, que ens servirà per a invocar els mètodes que hi implementem per dibuixar una taula amb els productes registrats a la base de dades d'una manera semblant(que no té perquè igual) a aquesta:

Nom Producte	Preu Unitat	Quantitat	Total
Sal	0.9	2	1.8
Xocolata	2	2	4
Total	5.8		

- **Desconnectarse** de la base de dades.

Nivell 2

- Exercici 2

Afegeix a sobre de la taula un botó "Nou Producte" que condueixi a la pàgina **insereix.php** on haurem de crear un formulari que ens permeti crear un producte a inserir a la base de dades.

- Exercici 3

Afegeix a cada fila de la Taula de productes comprats una columna que inclogui opcions per modificar i eliminar cadascun dels productes. Podeu indicar les opcions com vulgueu(mitjançant icones, botons...) de tal forma que:

- La opció de modificar ens porti a una pàgina **modificar.php** que inclogui un formulari per editar dades d'aquell producte.
- La opció d'eliminar ens permetrà esborrar aquell producte de la base de dades. Lògica que implementarem al nivell 3.

Nivell 3

- Exercici 4

Implementar les lògiques d'inserció, esborrat i modificació dels productes.

Recursos

Revisa la informació i els continguts que trobaràs dins del mòdul.

RESPUESTA/SOLUCION:

Funcionamiento del aplicativo:

He creado una aplicación para la gestión de los productos tal como se indica:

Adjunto un pantallazo de la aplicación:

INFORMACION DE PRODUCTOS:

#	Nom. Producto:	Cantidad:	Precio:	SubTotal:	#	#
46	Pan	1	24.71	24.71		
47	Arroz	1	10.1	10.1		
48	Harina	2	11.1	22.2		
49	Sal	9	12.2	109.8		
63	Pasta	2	25.3	50.6		
64	Macarrones	2	10.1	20.2		
65	Sal gorda	3	11.2	33.6		

Total = 271.21

Añadir producto:

Nombre:

Cantidad:

Precio:

Añadir...

Formulario (modo añadir).

La aplicación consta de una única pantalla. La pantalla consta de dos partes la parte **Izquierda** muestra la tabla con los datos ya guardados, con información de los productos: su id, nombre producto, cantidad(unidades), precio unitario y sub total (cantidad x precio). Tambien en la parte inferior se muestra la suma de todos los subtotales, en color azul para resaltarlo.

En la parte **derecha** tenemos el formulario de inserción de datos para añadir nuevos productos. Se permite poner nombre, una cantidad de productos que tiene que ser un número entre 1 y 100. El precio unitario tiene que ser un número con 2 decimales como máximo.

Si queremos editar/o borra un registro pulsaremos sobre el boton/icono edit/borrar en la parte derecha de la tabla.

49	Sal	9	12.2	109.8		
63	Pasta	2	25.3	50.6		
64	Macarrones	2	10.1	20.2		

Detalle de la tabla.

El icono editar nos abrirá el formulario de la parte derecha en modo editar, nos mostrará los datos para editar. Este formulario es ligeramente diferente al de añadir productos, ya que nos muestra los datos a editar del producto seleccionado.

Editar producto: 46

Nombre:

Cantidad:

Precio:

Modificar...

[regresar a añadir registro](#)

Detalle modo editar.

Para guardar los cambios pulsaremos sobre el “boton modificar...”. Por último si queremos regresar al modo añadir producto, en la parte inferior aparece un link para regresar al modo añadir un registro.

Si pulsamos sobre el icono de la papelera directamente se borra el registro.

Funcionamiento interno:

Hemos generado los siguientes ficheros .php

index.php :es la página principal con la tabla que muestra todos los datos guardados en la bbdd y dos formularios: **añadir** (por defecto visible), **modificar** (oculto por defecto). Se cambia entre ellos por medio de los datos que realizamos a este formulario, si no llega ningún dato se muestra el formulario añadir, y si llega el dato id se abre en modo update.

dbAccess.php :contiene los datos de la conexión a la bbdd ip, usuario bbdd, pss bbdd, nombre de la bbdd, este fichero también contiene la conexión y la desconexión a la bbdd.

tareas.php :contiene las objetos que interactúan con la bbdd **list** (para listar todos los registros), **create** (para añadir un registro), **show** (obtener datos de un registro por id) **update** (para actualizar datos de un registro), **delete** (para borrar un registro).

createRequest.php :obtiene los datos para realizar un insert de los datos llamando al objeto **create** para añadir un registro.

deleteRequest.php :obtiene el id y realiza la llamada al objeto **delete** para borrar el objeto.

editRequest.php : obtiene el id y realiza la llamada a la página index con id para que se abra en modo edición de un registro.

editRequest1.php :obtiene los datos modificados y los envía a dbAccess para realizar la modificación, con la llamada al objeto **update**.

Sql: adjunto el fichero **M8POO.sql** de creación del esquema, la creación de la tabla y los registros que ya tengo añadidos para test.