El objetivo de la práctica es realizar un CRUD (Create, Read, Update, Delete) a partir de la AEV4.

Tendremos como base un archivo csv donde estén los datos de las empresas a introducir en la cartera.

Para realizar este archivo csv puedes utilizar el código en data.php

**PARTE 2. Añadiendo funcionalidad para editar**

En esta parte vamos a añadir la funcionalidad para editar una empresa de la cartera.

Para ello utilizaremos el elemento (en mi caso la imagen del lapiz) que en la primera parte insertamos, de tal forma que al hacer clic en ella, se abra otra página en la que se muestren los campos de esa empresa, y al modificarlos poder guardar los cambios.

Deberemos de realizar los siguientes pasos:

-En la generación del html de drawList de Cartera, añadiremos una referencia a un archivo llamado update.php, que crearemos posteriormente. Esta referencia irá acompañada del identificador del cliente, para posteriormente pasárselo por parámetro mediante GET a la llamada del método update.

$output .= "<td><a href='edit.php?id=" . $client->getId() . "'><img src='img/edit\_icon.png' width='25'></a></td>";

-Añade un archivo llamado update.php, el cual será llamado en la línea anterior y le habremos pasado la id del cliente. Con esta id del cliente mostraremos los campos de ese cliente en el archivo update.php. Si os sirve de ayuda, yo he añadido un método en Cartera llamado getClient($id), el cual devolverá el cliente que coincide con el $id que le paso.

-En este formulario se podrán editar los campos y guardar mediante un botón. Para ello crea un método update en Cartera.

Este formulario contendrá una linea algo así:

<form id="form\_x" class="class\_x" method="post" action="edit.php?id=<?=$id?>">

De esta manera le decimos que el método utilizado va a ser post y la acción abrir el formulario con la id que hemos recogido previamente con GET al darle al botón desde el html de la lista de empresas (list.php en mi caso). Para ello existirán unas líneas antes tal que así:

$cartera = new Cartera("data.csv");

$id = isset($\_GET['id']) ? $\_GET['id'] : null;

$client = $cartera->getClient($id);

Además para poder enviar los datos una vez actualizados, deberemos de utilizar el siguiente código:

if (count($\_POST) > 0) {

$cartera->update($\_POST);

header("location: list.php");

}

Esto llamará al método update de la clase cartera, al cual le pasamos los datos con post.

Estos datos se gestionarán en el método como si de un array se tratara, con los nombres de cada elemento tal y como le hayamos puesto en el input del formulario.

Por ejemplo si en el formulario hemos escrito para uno de los campos:

.......

Cuanto tratemos esto en el método update de cartera lo trataremos así:

public function update($datos)

{

foreach ($this->clients as $client) {

if ($client->getId() == $datos["id"]) {

$client->setCompany($datos["company"]);

....