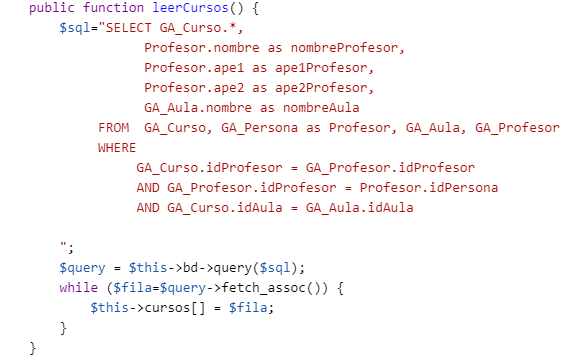
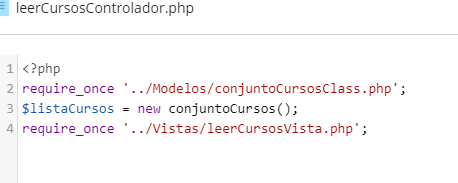
**Enlace a la aplicación** : <https://nascor04.md360.es/CRUD_GA_BDD>

**Metodo leerCursos dentro de conjuntoCursosClass:**

Creamos el método leerCursos donde le pasaremos la SQL con los datos del profesor, el aula y el curso y los guardaremos en la BBDD

**leerCursosControlador:**

****

En este caso llamaremos a la clase conjuntoCursos y crearemos la variable listaCursos con todos los cursos que tengamos e iremos a la vista para mostrarlos…

**leerCursosVista:**

****

En primer lugar haremos un condicional por tal de que si el array de cursos está vacío nos muestre el mensaje de que no hay cursos disponibles. En caso de que el Array tenga contenido, entrará por el else…

Haremos un foreach con el array de cursos que obtendremos en lista cursos y mostraremos cada curso que tengamos uno a uno, con los datos que pasamos en los echo (nombreCurso, nombre y apellidos de profesor, nombrel aula donde se imparte).

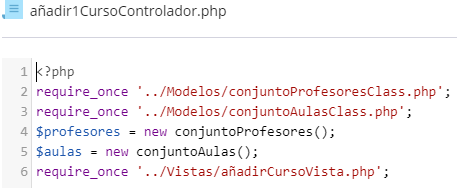
Añadiremos 2 botones donde pondremos el ID curso para utilizarlo mas tarde en el método GET para obtener la posición del curso y poder modificarlo/borrarlo.

**Método añadirCurso dentro de conjuntoCursosClass**:

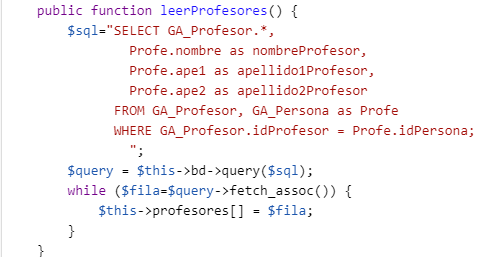


Crearemos dentro de la clase de conjuntoCursos el método añadirCurso con los parámetros que acabará recibiendo por $\_POST en el controlador para añadir los datos a la base de datos.

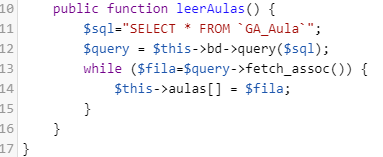
Le pasaremos la condición de que si está recibiendo un valor de la $fechaInicio y $fechaFinal, los coja, sino que lo ponga como NULL en la SQL.

**añadir1CursoControlador:**

Para añadir el curso llamamos a las clases conjunto profesores y conjunto aulas y creamos variables respectivamente con la lista de profesores disponibles y la lista de aulas, esto permitirá que al añadir un curso obtengamos ambos desplegables y podamos elegir donde y quién imparte el curso. Para eso necesitamos los métodos leerProfesores y leerCursos en sus respectivas clases.

**Método leerProfesores en conjuntoProfesoresClass:**Creamos la Query para obtener los datos de los profesores donde tendremos que acceder también a la tabla Persona mediante el idProfesor=idPersona y los almacenaremos en un array profesores[].

**Metodo leerAulas en conjuntoAulasClass:**

Aquí bastará con coger todos los datos de aula ya que no depende de ninguna foreign Key y las almacenaremos en el array aulas[].

**añadirCursoVista:**

****

Crearemos un formulario para añadir los datos del curso y los dos desplegables para elegir aula y profesor.

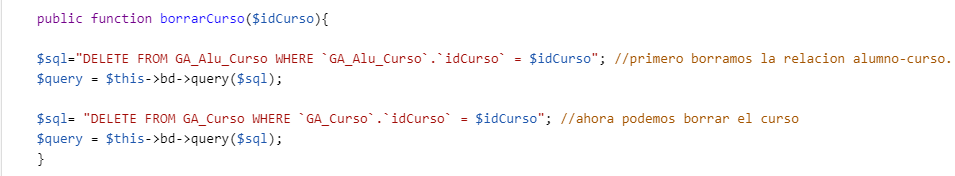
**añadir2CursoControlador**



Llamaremos al modelo conjuntoCursosClass y le diremos primero los condicionales, si el formulario nos devuelve para fecha inicio o para fecha final un valor vacío, le pondremos a la variable $fechaInicio valor NULL. Lo mismo haremos con $fechaFinal en caso de ser un valor vacío. En caso de que haya contenido, pondremos en $fechaFinal o $fechaInicio el valor introducido en el formulario. Eso nos permite ingresar tanto cursos sin fechas programadas como fechas programadas incluso cursos con fecha de inicio pero sin fecha final:

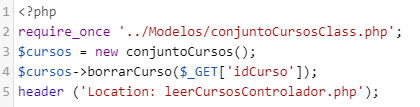


**Método borarCurso en conjuntoCursosClass:**

****

Aquí tenemos una relación del idCurso con el idAluCurso por lo que primero tenemos que borrar esta relación antes de borrar el curso en sí ya que idCurso es una foreign Key en GA\_Alu\_Curso.

**BorrarCursoControlador:**

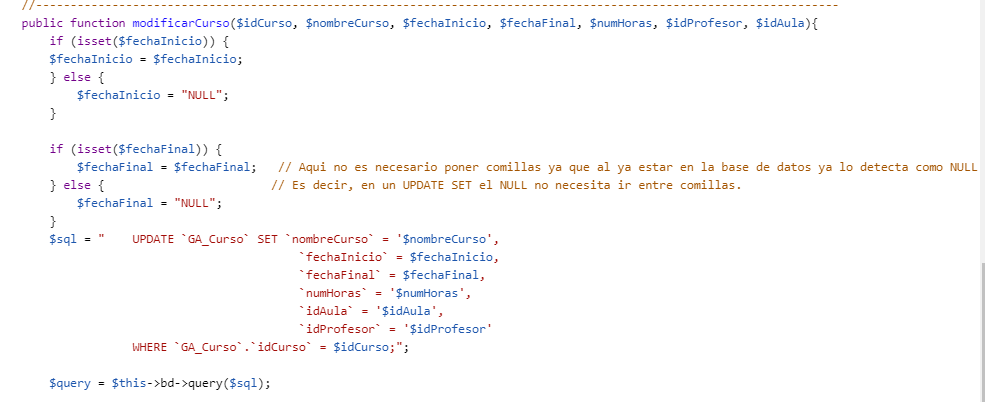
****

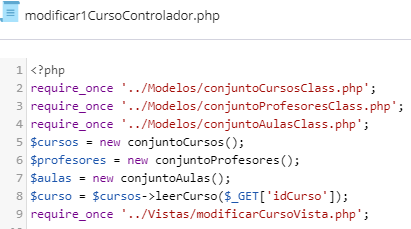
Llamamos a conjuntoCursosClass y cogemos la lista de cursos, aplicamos la función de borrarCurso donde recogemos el idCurso que queremos borrar y volvemos a leerCursos.

**Método modificarCurso:**

En el método le pasamos los parámetros con los datos a actualizar para modificar los datos del curso.

**modificar1CursoControlador**:





Llamaremos a los modelos conjuntoCursos, conjuntoProfesores, conjuntoAulas por tal de obtener el listado con los cursos, profesores y aulas (los dos últimos para poder obtenerlos en desplegables a la hora de modificar los datos).

En $curso cogeremos el curso a modificar con el método $\_GET e iremos a modificarCursoVista.

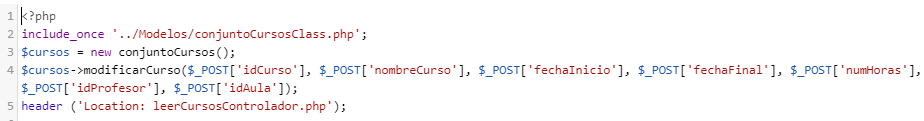
**modificarCursoVista**

****

Aquí pintamos un formulario con los datos del curso a modificar adquiridos por el id del curso que modificamos.

Mostraremos un foreach para mostrar tanto el array profesores como para el array aulas.

Pasaremos también un input type hidden con el valor del idCurso e iremos al hacer el submit a **modificar2CursoControlador**:

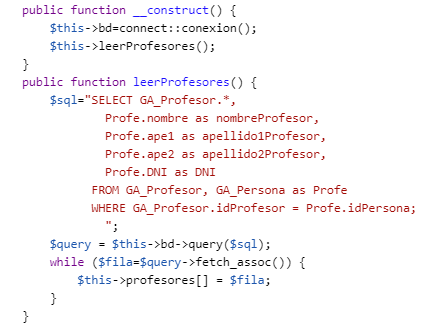


Volvemos a coger la lista de todos los cursos y ejecutamos el método modificarCurso donde le pasaremos en los parámetros los datos recibidos por el formulario (incluido el idCurso para que se actualice este mismo curso).

Volveremos a leerCursosControlador donde podremos ver el curso modificado.

**PROFESORES**:

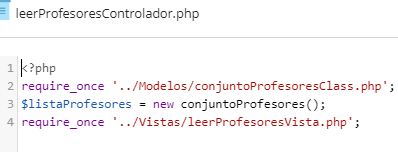
**Método leerProfesores** de la clase conjuntoProfesores



Hacemos la query con un select y los datos del profesor y de la persona relacionada entre idProfesor e idPersona.

Incluimos el método leer profesores dentro del construct para que al instanciar a la clase ya se haga automáticamente.

**leerProfesoresControlador:**



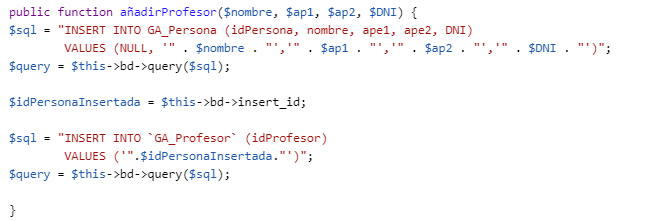
Llamamos a la clase conjuntoProfesores, al hacer una instancia de la clase conjuntoProfesores invocamos al método leer profesores y vamos a la vista:

**leerProfesoresVista**:

Haremos un foreach para recorrer el array de profesores profesor a profesor y mostrar sus datos obtenidos de la BBDD.

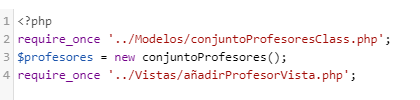
Antes de ese foreach tenemos un **condicional** también donde le diremos que si el array de profesores está vacio (no hay profesores) que muestre por pantalla un mensaje que diga que No hay profesores actualmente.  
**Esto es importante a la hora de la funcionalidad puesto a que si no hacemos este condicional y nos quedamos sin profesores, saldrá un error ya que no se puede hacer un foreach de un array vacío.**

Añadir profesor:

**Método añadirProfesor**

Un insert en SQL pasando como parámetros del método las variables que recibiremos en el $\_POST del formulario para rellenar los datos del nuevo profesor.

**añadir1ProfesorControlador**

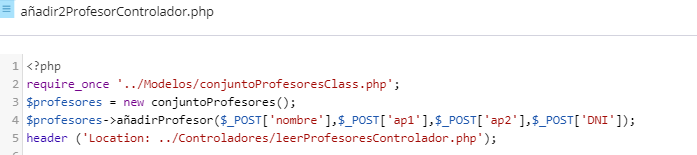
****

Simplemente se tiene que coger la clase conjuntoProfesores y el listado de todos los profesores e ir a añadirProfesorVista…

**añadirProfesorVista:**

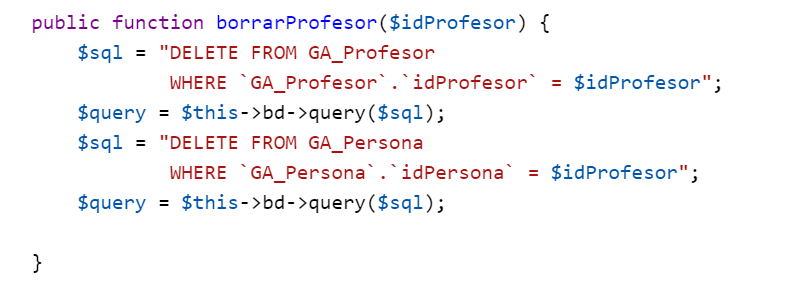


Este formulario es más sencillo ya que solo recogeremos los datos del profesor que añadimos y sus datos personales e iremos a **añadir2ProfesorControlador**:



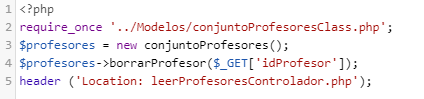
Simplemente llamaremos de nuevo a conjuntoProfesores y a la lista de profesores y llamaremos a la función de añadirProfesor con los datos recibidos en el formulario como parámetros.  
Finalmente iremos a leerProfesoresControlador.php donde veremos el nuevo profesor añadido.

**Método borrarProfesor** en la clase conjuntoProfesores:

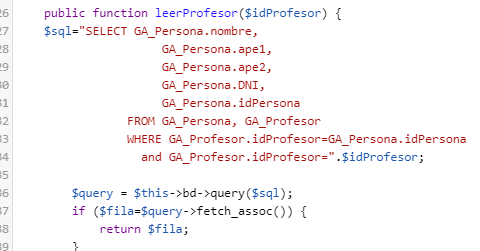


Hay que pasar 2 queries puesto que el profesor está relacionado con la persona(aun que también con el curso), es decir, no se puede borrar un profesor sin antes borrar el curso en el que está asignado. Por lo tanto, en la primera query borramos el profesor con el ID profesor igual al del parámetro y luego borramos el la persona con el idPersona igual al del parámetro.

**borrarProfesorControlador:**

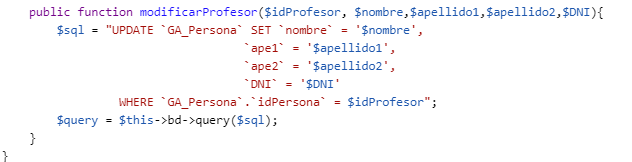
****Simplemente llamamos cogemos la lista del conjunto de profesores y le borramos la posición que nos llega por el enlace href con idProfesor y volvemos a leerPrfofesores donde ahora podremos ver como se ha eliminado dicho profesor.

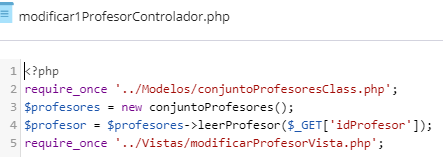
**Método leerProfesor** primero para poder modificar…

Seleccionamos en la query las tablas que queremos coger en FROM, en este caso persona y profesor, ya que están relacionados.   
Como solo queremos 1 profesor concreto, le especificaremos que deben de coincidir en el WHERE el idProfesor con el idPersona y el idProfesor con el idProfesor que le pasaremos en el parámetro.

Finalmente le diremos que nos seleccione los datos que queremos en el SELECT.

Método **modificarProfesor** en conjuntoProfesores:

Introducimos en la función los parámetros a modificar, el id aunque no se cambie también tenemos que pasarlo para coger el profesor del cual queremos cambiar los datos. Cogeremos la tabla donde se cambiaran los datos (en este caso Persona) y haremos un SET, donde haremos los cambios correspondientes en todos los valores pasados por el parámetro que luego en el modificar2Controlador al llamar a la función recibiremos por $\_POST.

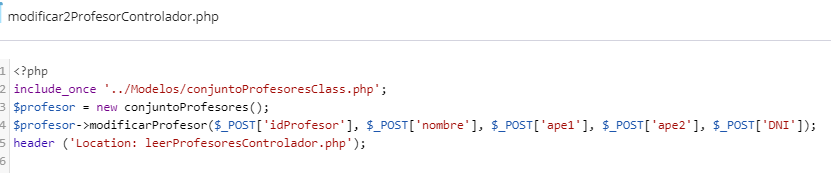
**modificar1ProfesorControlador**:

Llamamos a conjuntoProfesores y cogemos la lista de profesores que tenemos, llamamos a leerProfesor con el $\_GET para seleccionar el profe del que cambiamos los datos y vamos a la vista.

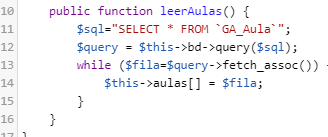
**modificarProfesorVista:**

****

Pintamos un formulario donde nos saldrán los actuales datos del profesor a cambiar gracias a la función leerProfesor. Al enviar vamos al modificar2.

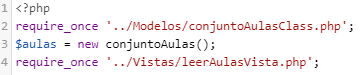
**modificar2ProfesorControlador  
**

Cogemos de nuevo la lista de profesores y usamos la función modificarProfesor donde le pasamos en los parámetros como dije anteriormente los datos pasados por $\_POST. En este caso el idProfesor lo hemos pasado por $\_Post en un input Hidden en el formulario.

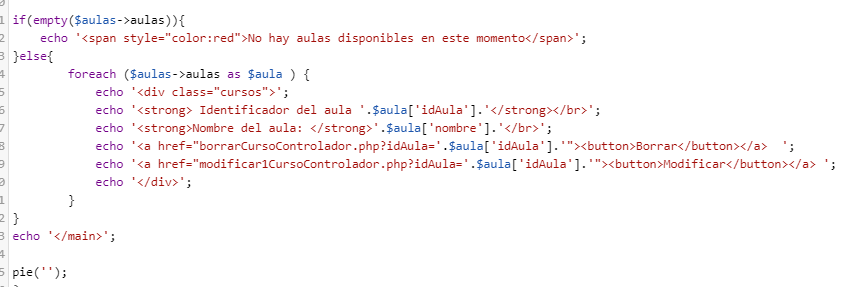
**leerAulas en conjuntoAulasClass**

Metodo leerAulas seleccionando todas las columnas de la tabla ya que solo tiene ID y Nombre, lo guardamos en un array aulas[]

**leerAulasControlador:**

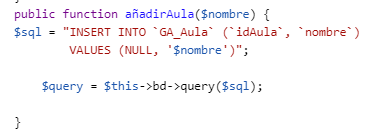
Cogemos la lista con todas las aulas y vamos a la vista para mostrarlas.

**leerAulasVista:**

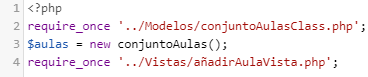
****

Mostramos en un formulario todas las aulas que tenemos en el array aulas[] mediante un foreach donde a cada aula le llamamos $aula y mostramos sus propiedades. Hacemos todo esto en un condicional donde le decimos primeramente que si $aulas->aulas (es decir el array aulas[]) está vacío que no de ningún error y en su lugar muestre un mensaje donde nos dirá que no hay aulas disponibles en este momento.

**añadirAula en conjuntoAulasClass:**

método añadir aula donde pasamos el $nombre como parámetro únicamente ya que el id es autoincrementado.

**Añadir1AulaControlador:**

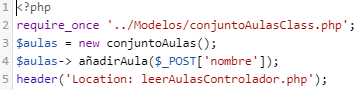
Cogemos la lista de aulas e iremos a la vista.

**añadirAulaVista:**

****

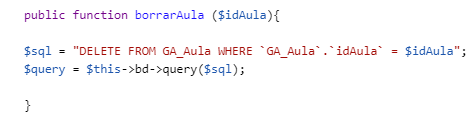
Pintamos un formulario simple donde tendremos un input donde ingresaremos el nombre de la nueva aula.

**Añadir2AulaControlador:**

****

Cogeremos la lista de aulas y dentro de ella añadiremos el nuevo aula con el nombre obtenido en el formulario.

**borrarAula en conjuntoAulasClass:**

Haremos el método borrarAula. En este caso la query es muy simple ya que no queremos que se borre el aula si tiene un curso asignado. Es decir primero deberíamos modificar el curso y cambiarlo de aula o borrar el curso.

**borrarAulaControlador:**

****

Aquí deberemos hacer un **session\_start** para mostrar el **mensaje error** en caso de que se trate de eliminar un aula que contenga algún curso asignado.

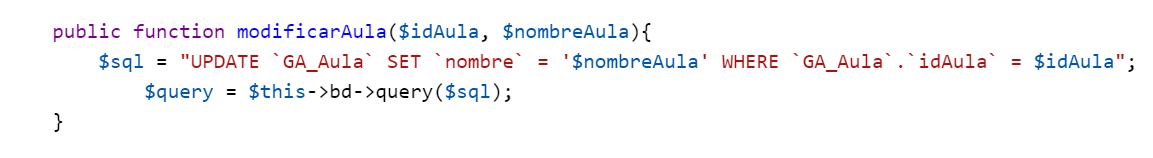
Cogeremos la lista de aulas y los cursos que tiene el aula en concreto que queremos eliminar.

Hacemos un **condicional** donde le diremos que **si cursoAula no está vacío** (es decir el aula contiene algún curso), muestre el **mensaje error** en la vista. En caso de estar la variable **cursoAula vacia** (no hay cursos en el aula), aplicamos el **método borrarAula** con el id obtenido por el enlace del href.

**Ahora hay que configurar la vista con el session\_start() para que muestre el mensaje si existe:**

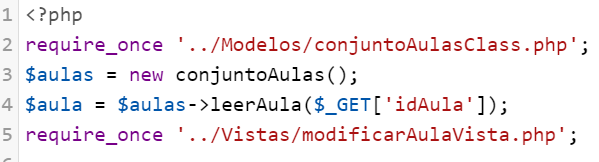
**leerAulasVista modificado después del borrado:  
**

Ahora modificamos el archivo leerAulasVista con un **session\_start()** al inicio y haremos un **if** donde le diremos que si existe el **$\_SESSION[‘mensajeError]** lo muestre. Esto hará que se muestre solamente cuando intentamos borrar un aula que tiene un curso asignado. Luego hacemos un **unset** para que se limpie después de mostarla.

**Método modificarAula en conjuntoAulasClass:**

Cogemos los datos a modificar con los parámetros que necesitaremos, en este caso el idAula que recibiremos por el get para modificar el aula en concreto y el $nombreAula que será el nombre que recibiremos por $\_POST.

**modificar1AulaControlador:**

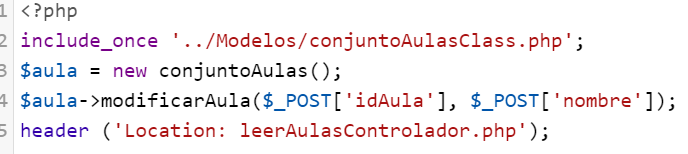
****

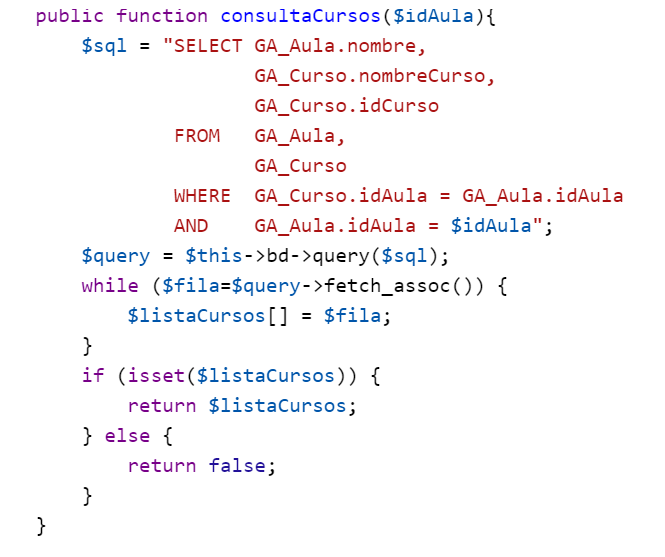
Cogemos el listado de aulas y hacemos el método leerAula para coger el aula que queremos modificar con el $\_GET que pasaremos por el href.

**modificarAulaVista:  
**

Pintamos el formulario en la vista donde recogeremos los datos (en este caso solamente el nombre) del aula que queremos modificar para que veamos cual es su valor actual antes de modificarlo y pasaremos el id del aula a modificar mediante un input hidden. Al cambiar el valor vamos al…

**modificar2AulaControlador:**

Cogemos el listado de aulas para hacer el método modificar los datos del aula en concreto (esto se sabe mediante el input hidden) que pasamos por $POST.

**CONSULTAR CURSOS QUE SE IMPARTEN EN UN AULA:**

Cogemos mediante el **FROM** la tabla **GA\_Aula** y la tabla **GA\_Curso**.

Vamos al **WHERE** y especificamos qué aula queremos y le ponemos que queremos ver los datos donde coincidan el idAula del curso y el idAula del aula para que salgan todos los cursos que hay con ese id aula o lo que es lo mismo, los cursos que se imparten en ese aula y con un AND le diremos que sea donde el idAula sea igual al idAula que pasamos por el parámetro de la función.  
Entonces pasamos al SELECT que será los valores que cogeremos, el nombre del aula, del curso y el id del curso.

Finalmente guardaremos todas las filas que recojamos que en definitiva serán los cursos asignados y los guardaremos en un array en $listaCursos.