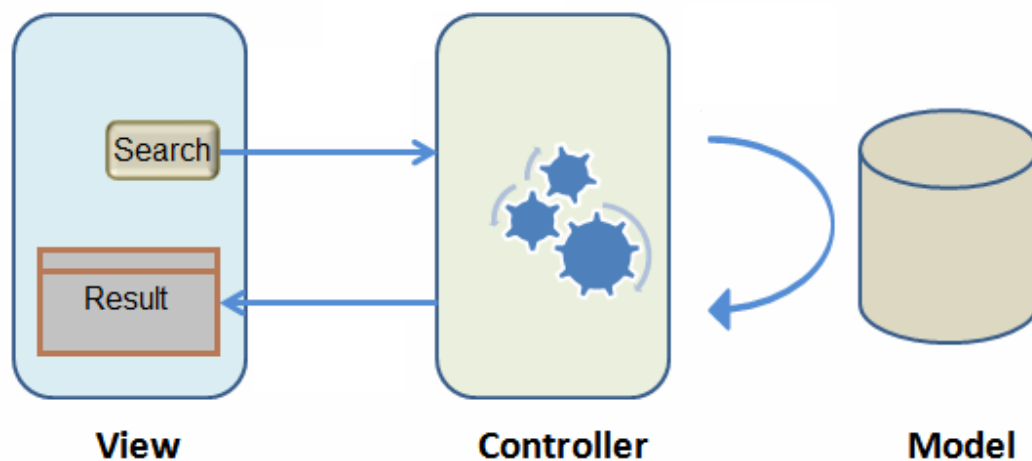


BBDD y POO - Teoría

Java no está pensado para hacer aplicaciones de menús de texto. Ni siquiera está pensado para ser ejecutado en un ordenador local, sacar cuatro ventanas y que el usuario pulse un par de botones. Esta clase de programas pueden hacerse en Java, pero rara vez se utiliza así salvo que estemos aprendiendo a programar. Cuando alguien decide usar Java para desarrollar sus aplicaciones es por dos motivos:

- **Java es Multiplataforma:** tú puedes desarrollar un programa en Java en una máquina, y mientras tengas instalada la Java Virtual Machine en otro equipo, por muy diferente que sea, el programa va a funcionar sin problemas.
- **Fácil de mantener:** si tú programas siguiendo ‘las normas’, separando bien las clases, los paquetes, etc. va a resultar muy poco costoso encontrar errores, añadir nuevas funciones o incluso modificar las ya existentes.

Cuando alguien dice coloquialmente ‘programa en Java’ se refiere normalmente a una aplicación del tipo **Cliente-Servidor**. Tú tienes una aplicación dividida en dos, uno o más Clientes instalados en una o más máquinas; y un Servidor en otra máquina. La comunicación entre un Cliente y un Servidor suele ser por una red o por internet. En estos casos es muy posible que también se refiera a que tenemos entre manos una arquitectura **Modelo-Vista-Controlador (MVC)**. Es lo mismo que antes, sólo que en este caso tenemos una Base de Datos en el Servidor.



Pero todo esto nos pilla muy lejos. Ahora nos toca pelearnos con Orientación a Objetos (POO) y aprender a acceder a una Base de Datos. Es decir, tendremos que saber cómo lanzar sentencias SQL a una BBDD y a procesar las respuestas que nos da.

XAMPP y MySQL

MySQL es un sistema de gestión de bases de datos (SGBD) de tipo relacional. Es decir, es un conjunto de programas que te permiten trabajar creando bases de datos, tablas, columnas, insertando datos, borrando otros, haciendo búsquedas, etc. Perteneció a **Oracle Corporation** y está considerada como la base de datos de código abierto más popular del mundo. Para trabajar con una BBDD de MySQL es necesario conocer **SQL**, un lenguaje de dominio específico diseñado para administrar y recuperar información de las bases de datos relacionales. Un ejemplo de estas sentencias es:

```
SELECT * FROM T_ALUMNOS
```

Por otro lado, **XAMPP** es un programa de software libre, que consiste principalmente en un MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. Sin entrar en detalles, se usa principalmente para desarrollar aplicaciones web, dado que el XAMPP tiene todo el software necesario para implementar la parte Servidor de estas aplicaciones (el Controlador y el Modelo).

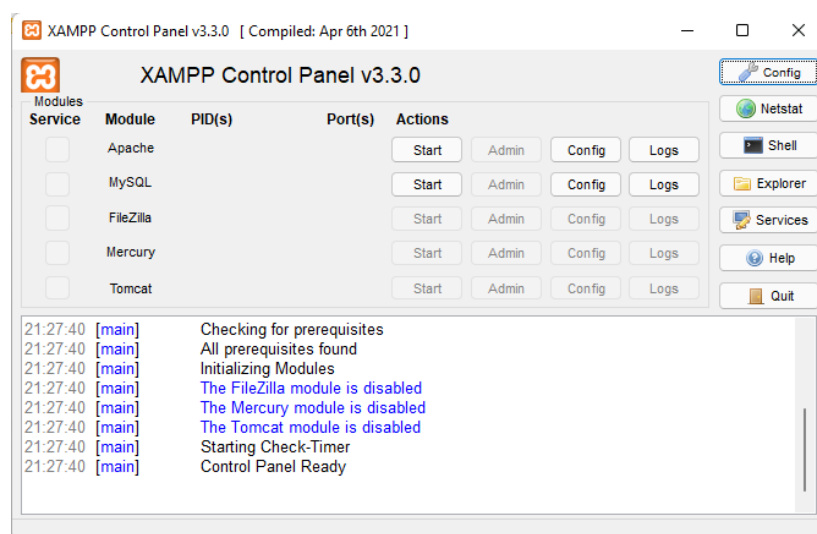
Lo que nos interesa del XAMPP es que, una vez correctamente instalado, nuestros equipos tendrán una **MySQL** a la que podremos conectarnos desde los programas Java que desarrollemos en Eclipse.

Instalar XAMPP

Para instalar XAMPP, acude al enlace siguiente y descarga la última versión:

<https://www.apachefriends.org/download.html>

Una vez descargado, en las opciones de instalación, no selecciones ninguna que no sea el Apache y el MySQL. No importa si lo haces, pero estarás ocupando espacio de disco para aplicaciones que no vamos a utilizar. Una vez completada la instalación, ve a la carpeta **c:\xampp**. Allí verás el programa **xampp-control.exe**, que es el que inicia el programa. Notarás que te aparece abajo a la derecha junto al reloj de Windows.



Debes de encender primero el servidor **Apache**. Después, enciende el **MySQL**. Tendrás que esperar a que te digan que todo marcha bien. Cuando desees apagar el XAMPP, haz el proceso inverso. Cerrar la ventana del Panel de Control no detendrá el Apache ni el MySQL.

Las Bases de Datos y los Servidores tipo Tomcat no tienen interfaz per se. Eso significa que no puedes ‘ver’ lo que están haciendo de la forma habitual. Para trabajar con ellos, tendrás que conectarte a ellos a través de un Navegador Web y usar las siguientes direcciones:

- **http://localhost/dashboard/** - Esta dirección nos lleva a la página por defecto del servidor Apache. No nos interesa para nada. No obstante, si el Apache está apagado o no funciona, esta página no se mostrará.
- **http://localhost/phpmyadmin/** - Esta dirección nos muestra la consola de administración de MySQL. De nuevo, si no te sale, es que no está arrancada la BBDD o funciona mal. Aquí podemos crear nuestras bases de datos, tablas, etc.

Si te pide una **clave de acceso**, el usuario es ‘root’, y la clave vacía.



Cómo usar el MySQL

El MySQL es bastante intuitivo, aunque un poco pesado de usar. Primero, pulsa en **Nueva** en el menú de la izquierda para crearte una nueva Base de Datos. Las que vienen por defecto son para que las use el propio MySQL, no hay que tocarlas. En la imagen, se ve que se ha creado nueva la **basedatosprueba**. Ves que tiene una única tabla creada llamada **t_alumno**. Se crean dándole al Nueva que hay dentro de la Base de Datos.

El menú de creación de una tabla es algo confuso: quizás tengas que probar un par de veces hasta que lo hagas bien. Las claves primarias tienen que ser todas ellas enteros (INT) no-nulas (descomenta el checkbox nulo), primarias (por Índice a Primary) y auto-incrementables (comenta el checkbox A_I). El resto, a tu gusto.

Si pulsas el botón **Previsualizar SQL** podrás ver la sentencia SQL de creación que se va a lanzar, por si no tienes claro lo que estás haciendo. Si le das al botón **Guardar**, se creará la tabla. Es posible también crear las tablas ejecutando sentencias SQL escritas a machete.

Si seleccionas una tabla, verás que se muestran varias pestañas. Las más importantes son:

- **Examinar:** Te muestra la tabla, con todas sus filas. También te permite hacer cambios directamente sobre la tabla sin lanzar SQL. Recuerda que hay que refrescar la ventana de vez en cuando para ver los cambios sobre la tabla.
- **Estructura:** Para modificar la estructura misma de la tabla: añadir columnas, etc.
- **SQL:** Permite lanzar sentencias sobre la tabla. Ten cuidado, porque todas las sentencias excepto el Select modifican los datos de las tablas. Pero no hay nada que te impida hacer lo que quieras mientras la SQL sea correcta.
- **Insertar:** Para insertar a machete filas en la tabla.

A la hora de trabajar con MySQL, recuerda que las Tablas sólo pueden relacionarse con Tablas de una misma Base de Datos; pero no hay límites en cuanto a las Tablas que quieras utilizar o las relaciones que hagas entre ellas.

Hasta aquí todo lo que tienes que saber de XAMPP y MySQL para la asignatura de Programación.

Bases de Datos desde Java

Tienes un ejemplo completo de código Java subido en Moodle que realiza consultas contra una BBDD. Pégale un vistazo antes de seguir.

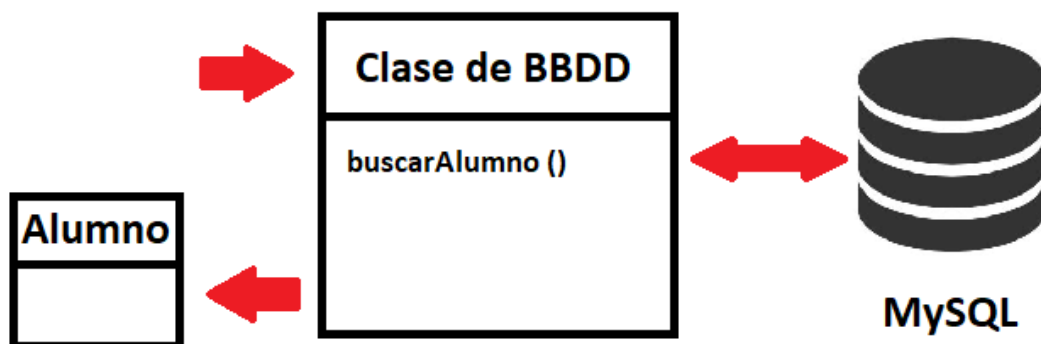
Todo lo que existe en el mundo mundial para Java son Clases y Objetos; y una Base de Datos no iba a ser diferente. Para poder **conectarse** a una Base de Datos, necesitaremos de una Clase que contenga el código necesario para establecer la conexión, unos métodos que lancen las **consultas** (sentencias) SQL e interpreten los resultados, y unas clases que **contengan** esa información en Java (POJOs).

Simplificando mucho, lo más habitual es que tengas una serie de clases con los métodos de acceso a BBDD y otras que describan cada una de las filas de la tabla. En nuestro ejemplo, tenemos una tabla **t_alumno**. Esta tabla tiene una serie de datos: id, nombre, apellidos y edad. Necesitaremos por tanto crear una clase que describa esa tabla.

t_alumno	
COLUMNA	TIPO
Id	Int
Nombre	Varchar (10)
Apellidos	Varchar (30)
Edad	Int

Alumno (POJO)	
ATRIBUTO	TIPO
Id	Int
Nombre	String
Apellidos	String
Edad	Int

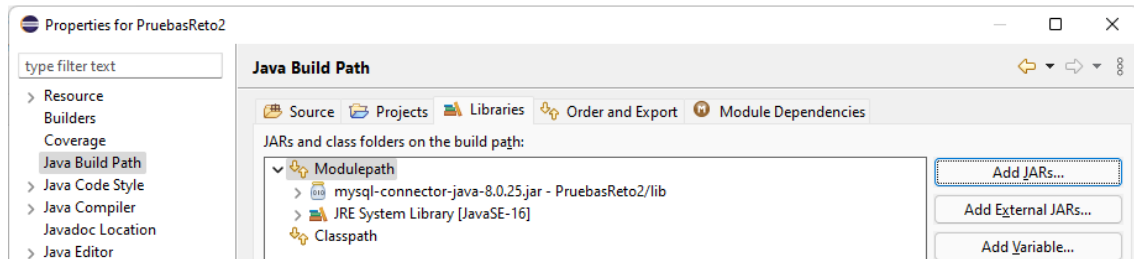
La clase **Alumno** describe una fila de **t_alumno**. POJO significa “Plain Old Java Object”, que es una forma de llamar a un Objeto de una Clase que sólo tiene **Atributos** y **Getters & Setters**. Cuando haces una consulta a la BBDD desde Java, lo que tienes de respuesta no puede usarse directamente. Para poder trabajar con ella, se hace lo siguiente: se extrae la columna nombre de una fila, y se guarda en un Objeto Alumnos mediante `alumno.setNombre (...)`. A continuación, se saca otra columna y se repite el proceso. De esta manera, tendremos un Objeto Alumno por cada una de las filas devueltas por la BBDD.



Configurar el acceso a BBDD

Java (Eclipse) necesita las siguientes cosas para poder conectarse a una BBDD:

- **El conector:** Es una librería que proporciona las Clases y métodos necesarios para establecer la conexión. Las librerías son código Java hecho por terceros que, añadidos a nuestro proyecto, nos permite reutilizar su código. La librería tiene que añadirse a tu proyecto, no basta con copiarlas en la carpeta lib. Para ello, dale a **Propiedades del Proyecto\Build Path** y **Add Jars**. Selecciona la librería, y verás algo así:



- **Datos de conexión:** La forma más sencilla es crear una clase de configuración llamada **DBUtils**. Aquí colocaremos la URL a la BBDD, el Driver que tenemos que usar, el usuario y el password de la Base de Datos.

```
// Configuraciones necesarias para la Base de Datos de MySQL
public class DBUtils {

    // La URL donde esta la Base de Datos. Se descompone en:
    // driver : bdd : // IP : Puerto / Schema
    public static final String URL = "jdbc:mysql://localhost:3306/test";

    // El Driver que vamos a usar
    public static final String DRIVER = "com.mysql.cj.jdbc.Driver";

    // Nombre y Pass de acceso a la Base de Datos
    public static final String USER = "root";
    public static final String PASS = "";

}
```

Métodos de acceso a BBDD

Para acceder a BBDD, se usan métodos que utilizan la librería importada anteriormente. Por regla general, lo más conveniente es que estos métodos reciban y retornen datos en forma de Clases (POJO), aunque no es obligatorio. Cada vez que hagamos un acceso, será necesario indicar el **driver** que se va a usar, **abrir** una conexión con la BBDD, abrir el **statement** con la sentencia SQL, y si es necesario abrir un **resultSet** (Select) para procesar las respuestas. Es muy importante **cerrar** de nuevo todo lo abierto en orden inverso a su apertura, o la BBDD puede colapsarse y no retornar conexiones.