

Acceso a MySQL desde Java

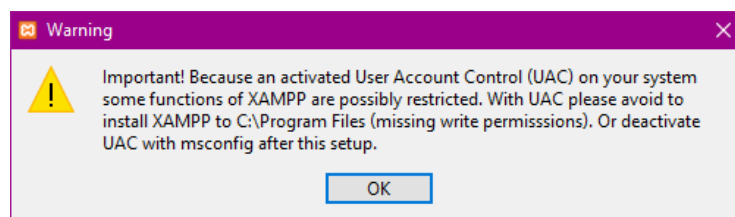
Introducción

Vamos a acceder a una base de datos MySQL desde Java. Vamos a instalar y configurar un servidor MySQL al que nos conectaremos desde NetBeans utilizando Java. Necesitamos un servidor MySQL que permita ser atacado desde Java y una base de datos MySQL.

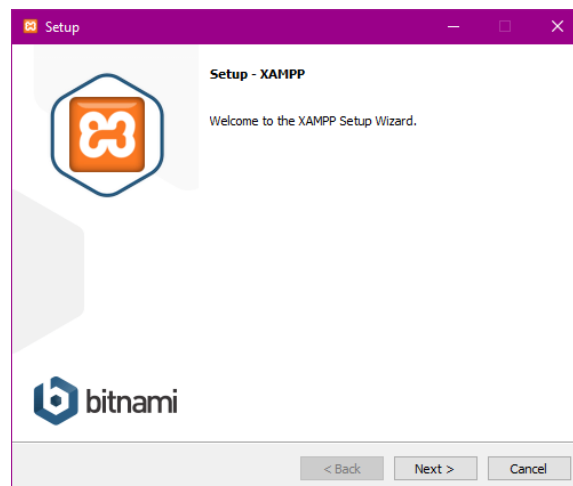
Instalación de XAMPP bajo Windows

Accedemos a <https://www.apachefriends.org/index.html> y descargamos la última versión disponible e instalamos.

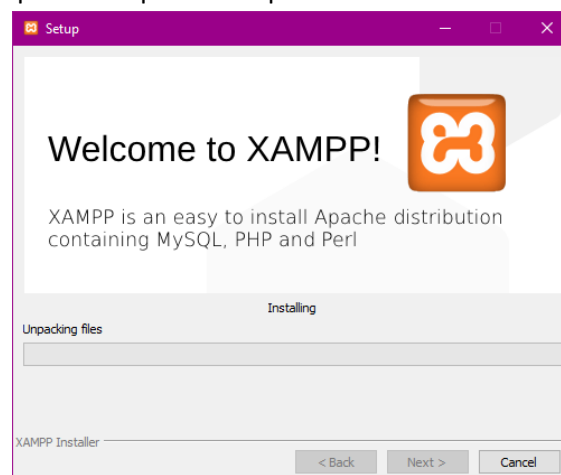
Si en la instalación aparece la siguiente ventana:



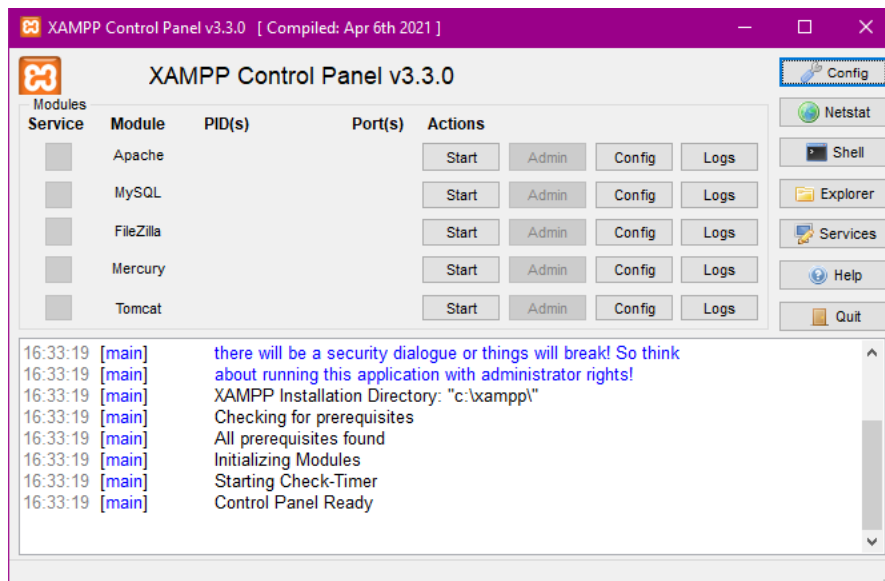
Pulsamos OK.



Pulsar Next en todas las pantallas que van a apareciendo hasta...



Hasta llegar a la última pantalla que pulsamos Finish y aparece el Panel de control:



Si tenéis curiosidad, en [este enlace](#) podéis ver una guía rápida de XAMPP. Podemos ver todos los servicios disponibles con XAMPP.

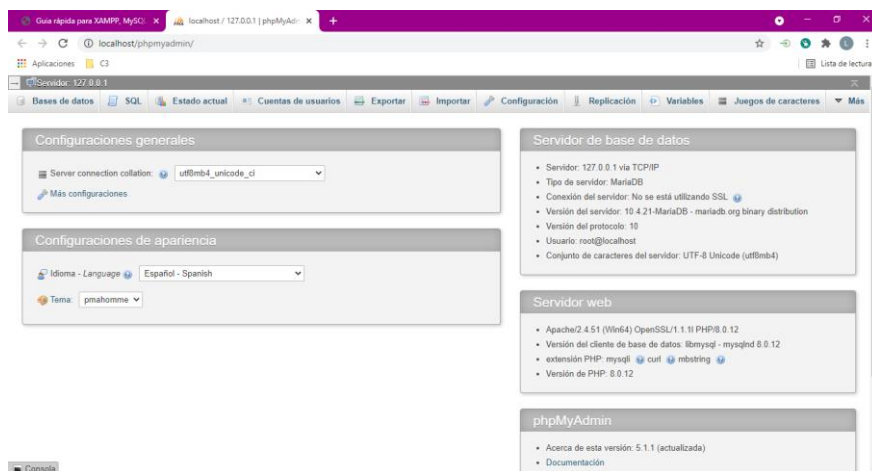
Configuración de MySQL

En primer lugar, vamos a arrancar los servicios que necesitamos:

1. Arrancar Apache (Start)
2. Arrancar El sistema gestor de bases de datos (Start de MySQL)
3. Entrar en phpMyAdmin (Admin de MySQL).

Nos pide usuario y contraseña (accedemos con root y sin contraseña) pero **cuidado** que somos súper usuario.

Vamos a establecer contraseña para root:



Pulsamos en Cuentas de usuarios. En la ventana que aparece, pulsamos en Editar privilegios de root/localhost. Seleccionamos Change password y establecemos la contraseña y paramos MySQL y Apache en el panel de control.

Con los servicios parados y phpMyAdmin cerrado. En C:\xampp\phpMyAdmin\config.inc.php

Cambiamos

```
/* Authentication type and info */
$cfg['Servers'][$i]['auth_type'] = 'config';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = '';
$cfg['Servers'][$i]['extension'] = 'mysqli';
$cfg['Servers'][$i]['AllowNoPassword'] = true;
$cfg['Lang'] = '';
```

Por

```
/* Authentication type and info */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
$cfg['Servers'][$i]['extension'] = 'mysqli';
$cfg['Servers'][$i]['AllowNoPassword'] = true;
$cfg['Lang'] = '';
```

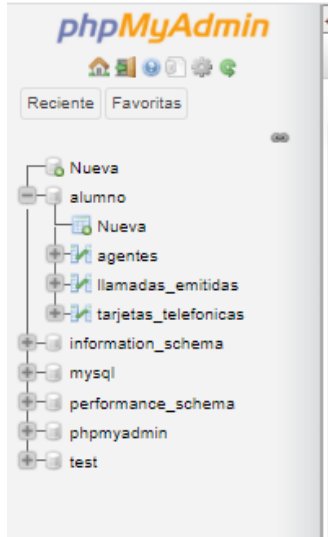
Ahora vamos a crear un usuario nuevo. Accedemos de nuevo a phpMyAdmin (pedirá usuario y contraseña) y creamos un usuario y contraseña (**alumno/alumno**) en Cuentas de usuarios → Agregar cuenta de usuario:

Y pulsamos en Continuar.

Creación de tablas

Arrancamos el servicio y accedemos con el usuario **alumno**.

Seleccionamos y copiamos el contenido de agentes.sql y en la pestaña SQL lo pegamos y pulsamos en Continuar:

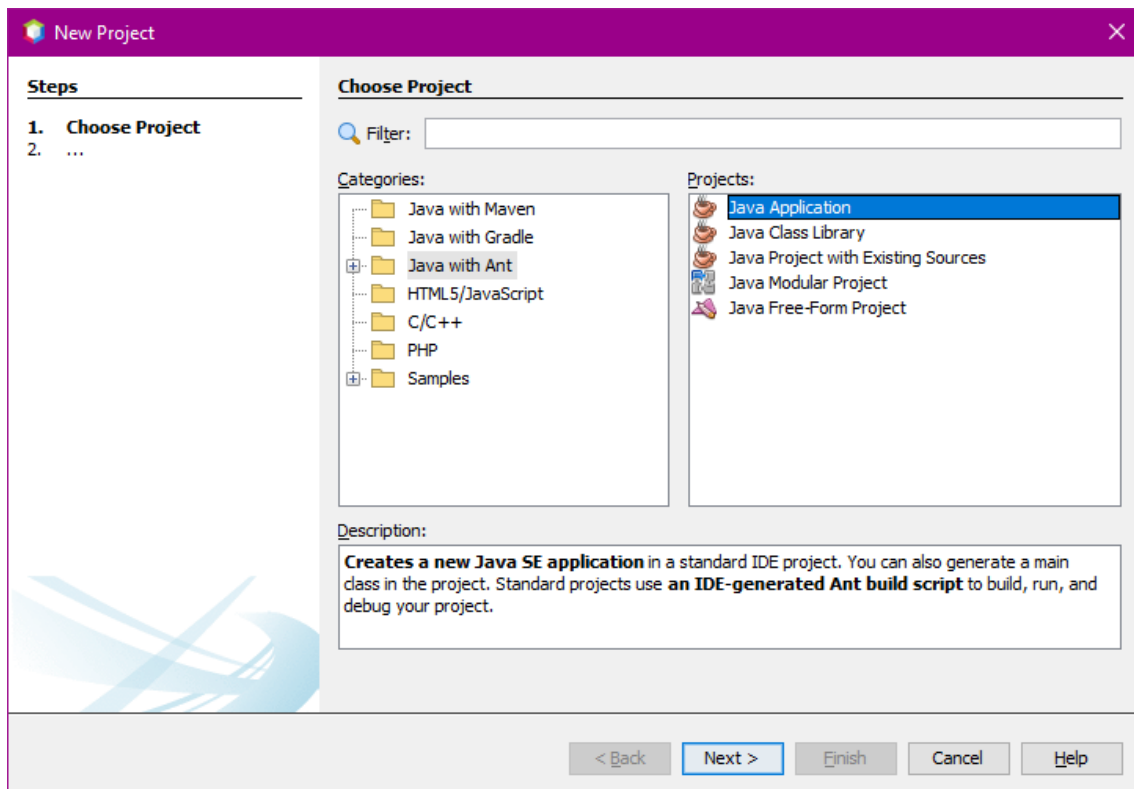


Cerramos el navegador.

Acceso a MySQL desde Netbeans/Java

El servidor MySQL debe estar activo (no es necesario Apache). Necesitamos JDBC. Lo descargamos de <https://dev.mysql.com/downloads/connector/j/8.0.html> (o de aulavirtual).

Debemos incluir la librería en todos los proyectos:

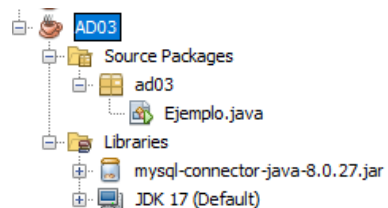
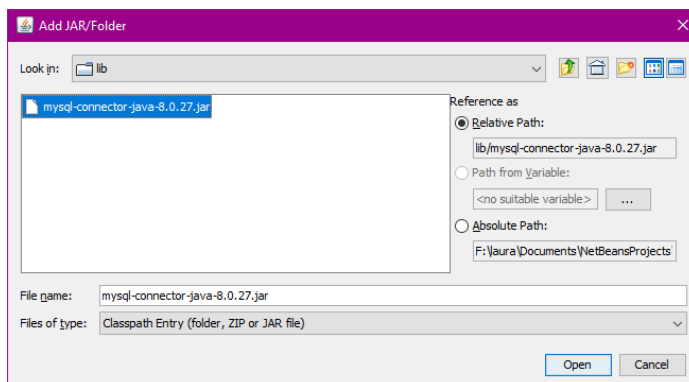


Le damos nombre en la siguiente pantalla.

Ahora buscamos dónde están los proyectos en NetBeans y copiamos el fichero jar.

| Nombre | Fecha de modificación | Tip | Tamaño |
|-------------|-----------------------|---------------------|--------|
| lib | 28/10/2021 18:23 | Carpeta de archivos | |
| nbproject | 28/10/2021 18:19 | Carpeta de archivos | |
| src | 28/10/2021 18:19 | Carpeta de archivos | |
| build.xml | 28/10/2021 18:19 | Archivo XML | 4 KB |
| manifest.mf | 28/10/2021 18:19 | Archivo MF | 1 KB |

Yo lo he hecho en una carpeta lib dentro del proyecto. En NetBeans pulsamos con el botón derecho sobre el proyecto > Properties > Libraries : Add Jar/Folder:



Ejemplo de proyecto java que accede a MySQL

El paquete `java.sql.*`; es el que proporciona acceso a la base de datos. Debemos incluirlo en nuestra clase.

1. Crear una conexión con la base de datos

Primero nos aseguramos de que MySQL está activo.(Start del XAMPP).

Ahora creamos la conexión con la base de datos a través del correspondiente JDBC. Para ello creamos un objeto llamado `conexion` de la clase `Connection` y le damos los valores necesarios. Esos valores son:

- Quién hace la conexión y hacia dónde se hace: `jdbc:mysql`
- Dónde está el servidor de base de datos, por qué puerto está escuchando y cuál es la base de datos: `//localhost:3306/alumno`
- Qué usuario es y qué contraseña tiene: `alumno/alumno`

```
Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/alumno","alumno","alumno");
```

2. Realizar una consulta

```
public static void main(String[] args) {
    try {
        Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/alumno","alumno","alumno");
        Statement query = (Statement) conexion.createStatement();
        ResultSet resultSet = query.executeQuery("select nombre_agente, frase_clave from agentes");
        while (resultSet.next()) {
            String nombre_agente = resultSet.getString("nombre_agente");
            String frase_clave = resultSet.getString("frase_clave");
            System.out.println(nombre_agente+" - "+frase_clave);
        }
        resultSet.close();
        query.close();
        conexion.close();
    } catch (SQLException ex) {
        System.err.println(ex);
    }
}
```

3. Insertar nueva fila

Inserta un agente nuevo:

- NOMBRE_AGENTE: Austin Powers
- FRASE_SECRETA: Peligro es mi segundo nombre

Para el código ejecuta la siguiente consulta:

```
SELECT max(CODIGO_AGENTE) as maximo FROM AGENTES;
```

Y asigna máximo +1.

Accedemos a phpMyAdmin y comprobamos que se ha insertado.

4. Crear una tabla nueva

Desarrolla el código necesario para crear una nueva tabla MISIONES. Lo que se necesita es almacenar las misiones que los distintos agentes han llevado a cabo. Así, esa tabla MISIONES tendrá los campos:

- Código de misión, de tipo INT, que será el campo clave de la tabla
- Nombre de la misión, de tipo VARCHAR
- Agente encargado de la misión, de tipo INT, siendo clave ajena a la tabla AGENTES

En SQL:

```
CREATE TABLE MISIONES(  
    CODIGO_MISION INT NOT NULL,  
    NOMBRE_MISION VARCHAR (80) NOT NULL,  
    COD_AGENTE INT NOT NULL,  
    CONSTRAINT PK_MISIONES PRIMARY KEY (CODIGO_MISION),  
    CONSTRAINT FK_AGENTE FOREIGN KEY (COD_AGENTE) REFERENCES AGENTES  
    (CODIGO_AGENTE)  
);
```