# Java Audio Player Application Setup Guide

The application consists of different elements connecting together and finally displaying an audio player form which manages not only to authenticate a user based on a MySQL database, but also to stream specific audio files directly from the form.

To setup the project in different machines a sequence of steps is required:

Load the files

After downloading the files needed for the project to run, we must first load them into Netbeans or any other related program that supports JForms.
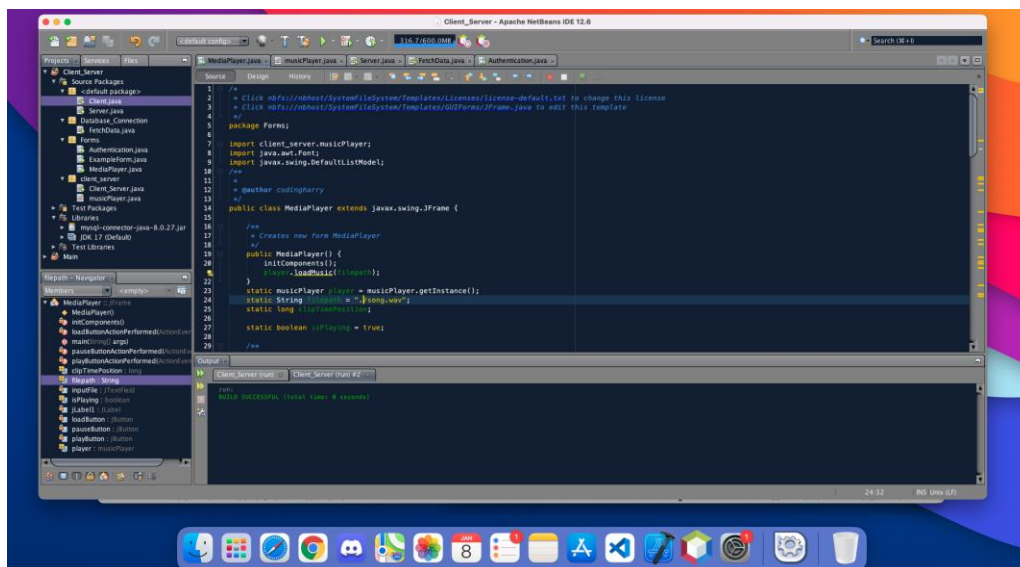

Establish a .jar connection

After loading the files, we must then connect the MySQL connector file (.jar) directly from the file system to our project, in order for it to successfully connect to the database. To achieve that, we navigate to the "Projects" tab at the top left pane of Netbeans as shown below.

Then we click the dropdown arrow named "Libraries" and then we right click on it, selecting "Add .jar/Folder".
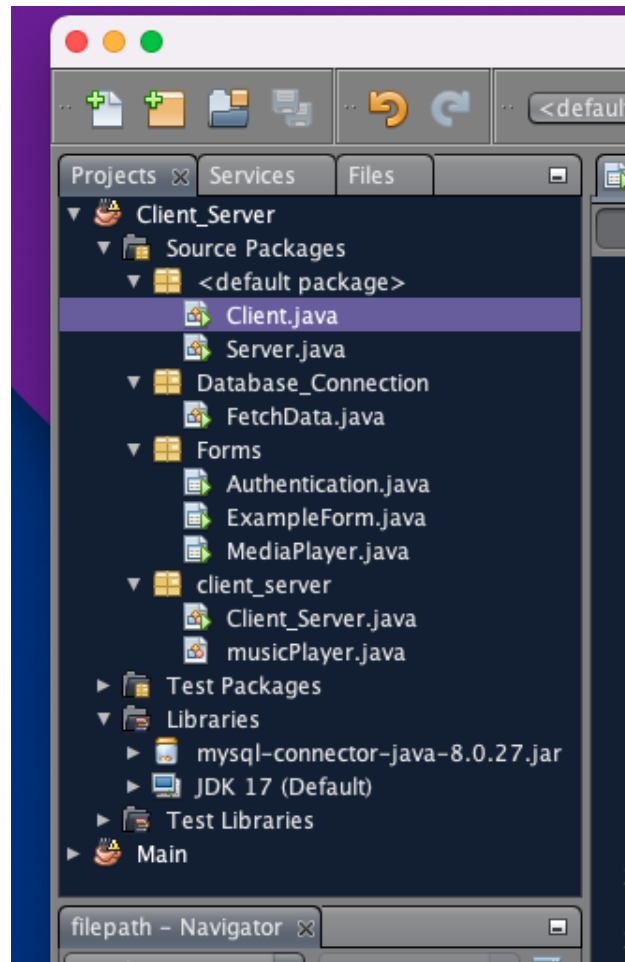
From the modal box that appears, we select the .jar file ("mysql-connector-java.jar") from our file system and finally click "Open" from below.
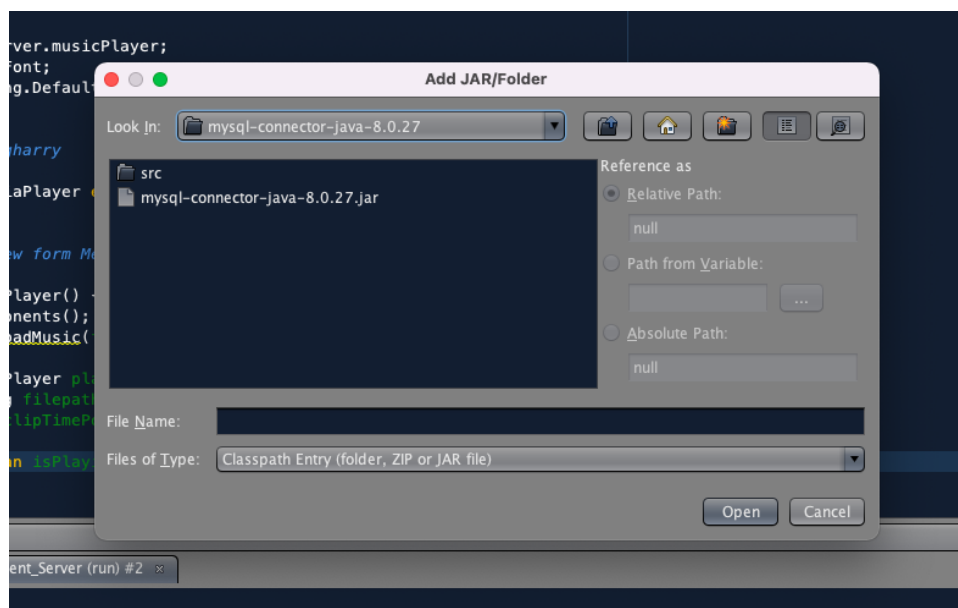

After successfully connecting the .jar file with our application we should also then navigate to the "Services" tab on the top left pane of our application. Selecting "Databases" and then "Drivers" should display a variety of libraries inserted in our application. Right click on "Drivers"and then "New

Driver" should also complete the addition of the .jar file within the application. Choose the same file that you chose previously from the "Libraries" tab.

Database Connection

Within our application there is a functioning database code that helps connecting the database with our application together. After successfully connecting our database, we can open the XAMPP application and by enabling MySQL and opening phpMyAdmin we should then navigate to create a new database with the name of "audioplayer".
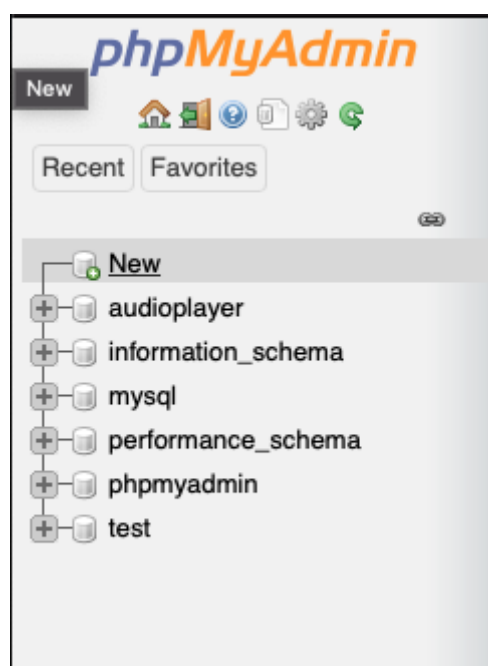

After achieving the creation of the database, named "audioplayer", we should then navigate to our application files and open the "audioplayer.sql" file that contains the working SQL code of our application, copy and then paste it inside the SQL section of the database.
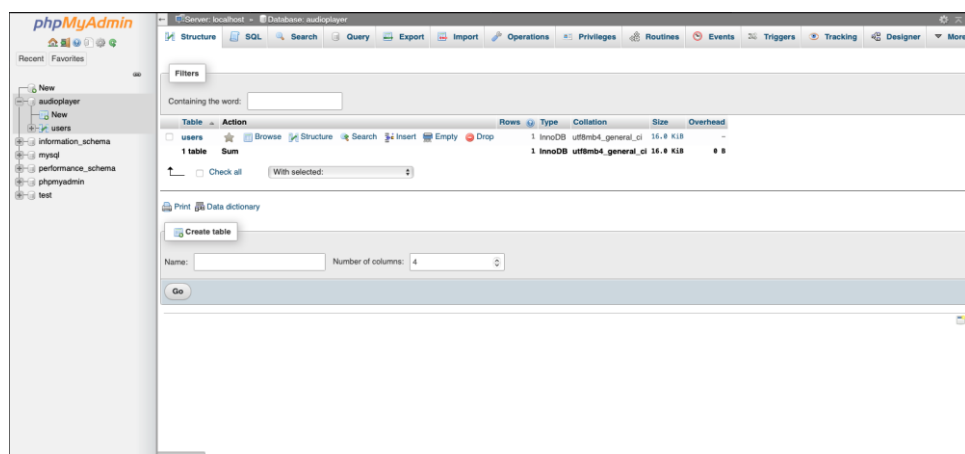
We can then click on "GO" and our code will be successfully compiled and ready for direct in-app use.

Running the application

Within the application itself, there are two basic things that need to be done in order to run it. Firstly, we should run the "Server.java" file in order for our Server to be able to run. Secondly, the "Client.java" file should be run, in order for an available client to be able to sign into the application with specific credentials according to our database and then stream music.
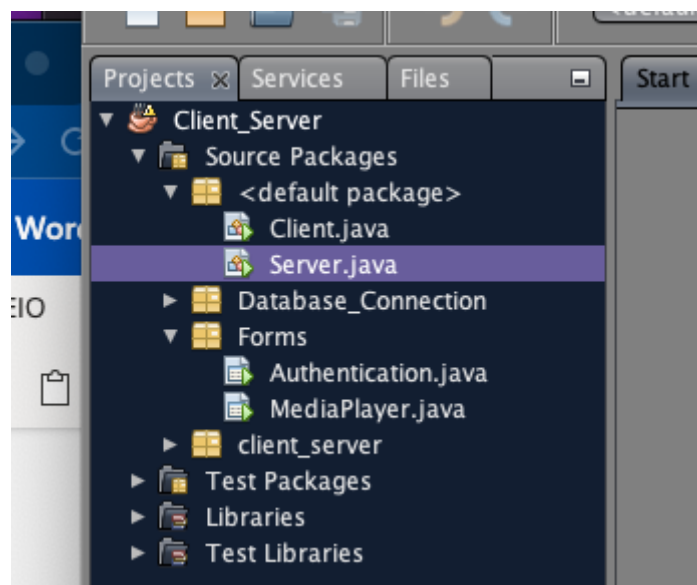
The application runs with four different songs, just by typing the name of the following files in the load section:

- song.wav
- roads.wav
- endoftheroad.wav
- Maria.wav

```
233                    gainControl.setValue(-10.0f);
234                    System.out.println("You've reached the limit!");
235                    System.out.println("Volume: " + gainControl.getValue());
236                    JFrame jFrame = new JFrame();
237                    JOptionPane.showMessageDialog(jFrame, """
238                                                          You have reached the volume limit!
239                                                                        gainControl.ge
240                    }
241                }
242            }
243
245    private void
246        try {
249            Stri
250            File
251            Syst
252            Syst
254            File
254            JOpt
255        }
        catch (E
257            Syst
258        }
259    }
260
261    /**
262     * @param args the command line arguments
```

roads.wav   LOAD

PLAY

PAUSE

Volume

+   −

Download