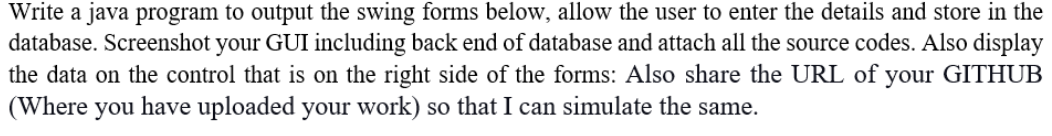
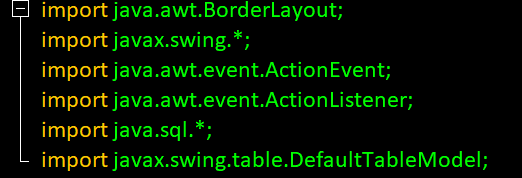
**JAVA CAT 2**

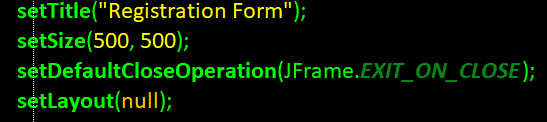


First we start by importing all the required imports for our program.



Those imports will allow us implement borders, use swing GUI, implement ActionEvent and ActionListener, use SQL for java and lastly help implement table models.

We then go ahead and create the jframes for the first form i.e. registration form.



We continue adding all the required components in a GUI including the following;

* **Two fields that accept in data that is name and mobile field.**

****

* **Two radio buttons to capture the individuals gender.**

****

* **A drop down to capture DOB.**

****

* **A field to capture the address.**

****

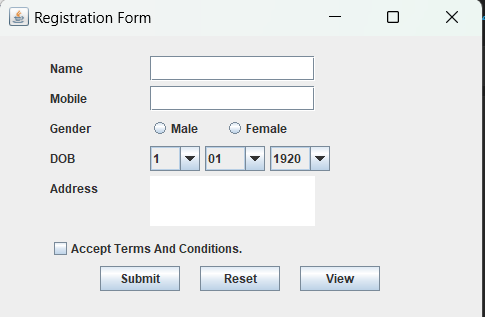
* **A check box to accept terms and conditions.**

****

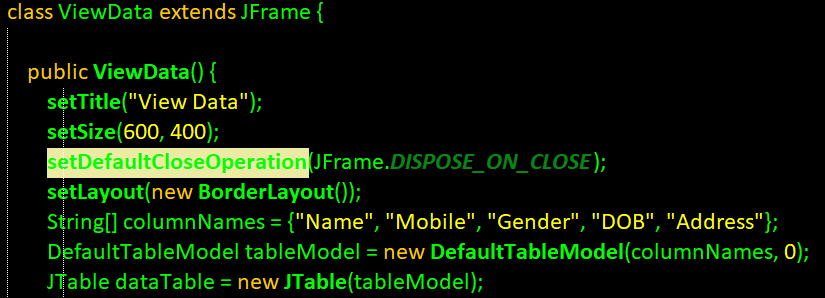
* **Three buttons, submit, reset and view for submitting keyed data, resetting the fields to null and viewing the data keyed in respectively.**

****

Generally, the registration form looks like this

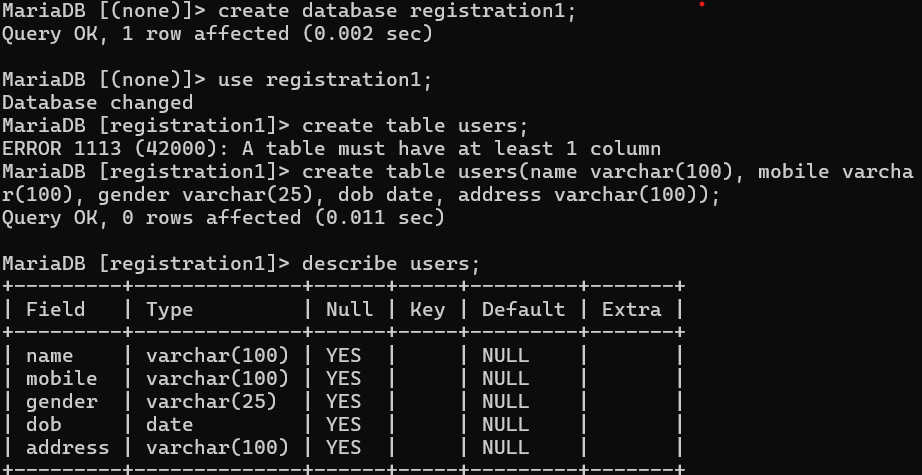


We then come up with a code that will create and display the data form so that when one clicks view he’ll be able to see the recently keyed in data.

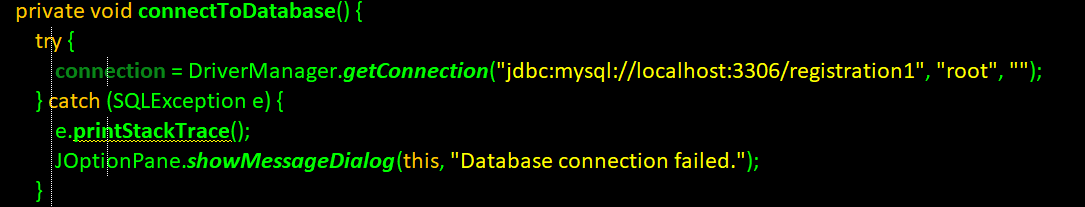
After coming up with the forms, we then move to creating a database connection so that it can be able to capture the entered data, store it and finally retrieve it.

I opted for MySQL for its database because of its simplicity and because its widely used.

We therefore moved on and created a database named registration1 and all of its necessary columns as shown below



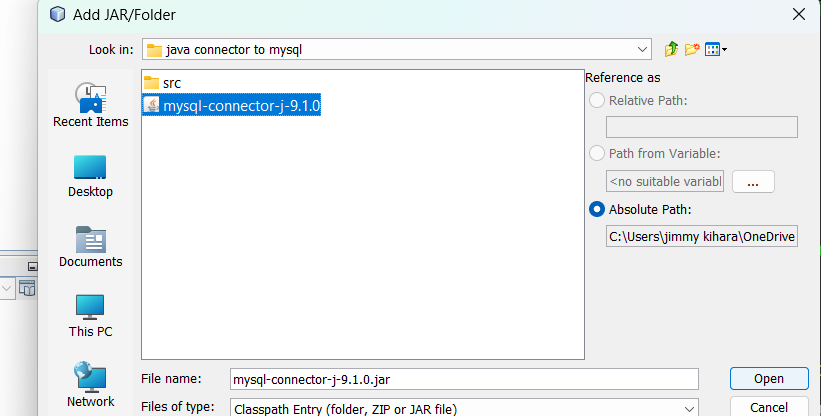
Since the database is already available we create a connection via the code to port 3306 local host which is for MySQL as shown below



And since it can’t connect directly we also add a driver for MySQL connector in the libraries to make it even more efficient

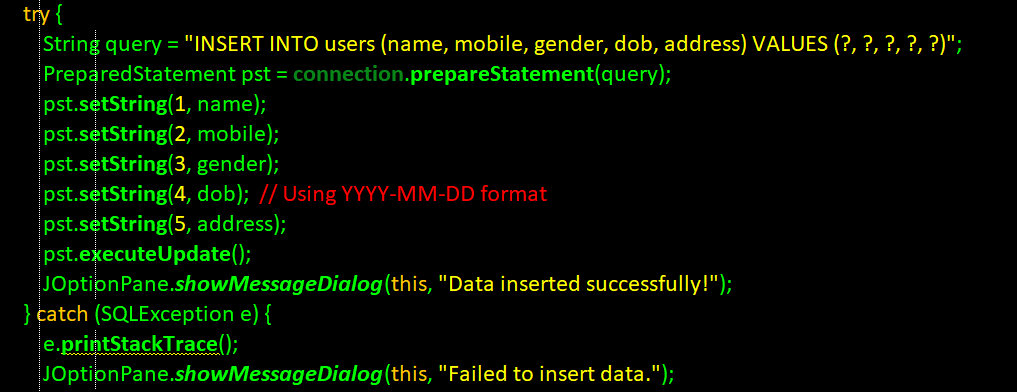
This really does make the jdbc get the connection from the port whether you are using xamp or the MySQL shell.

The attached media shows the downloaded MySQL connector being loaded to the libraries of the project.

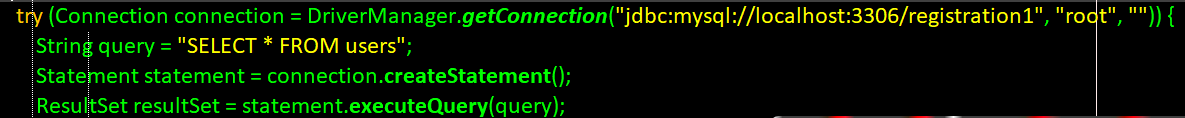


After creating all this, we modify the code so that after the user has keyed in data when registering it automatically inserts them to the SQL database we created earlier.

It captures them and stores the respectively.



Again not forgetting we had a second GUI to display the registered data of the individual we therefore add few lines that will be able to fetch the same data entered from our database and display it also, the following picture shows how the code looks like.

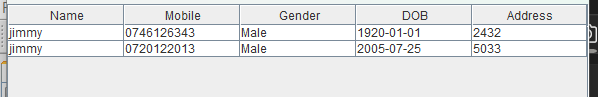


After retrieving data, we get this in our registration form

1. **The data we entered**



1. **The data we see**



We see two rows of data because we entered them.

**THUS THE REGISTRATION FORM WITH JAVA AND ITS DATABASE CONNECTION IS SUCCESSFUL, BOTH POSTING AND RETRIEVING DATA.**