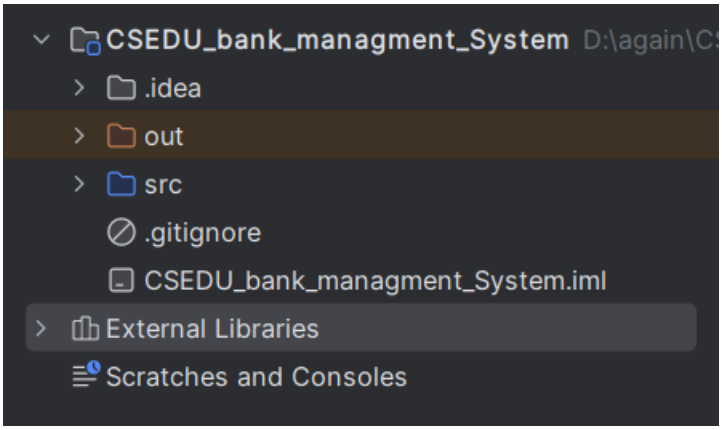
Project Setup:

The project will run in any machine after following the bold instructions. I will explain here the way to setup in IntelliJ.

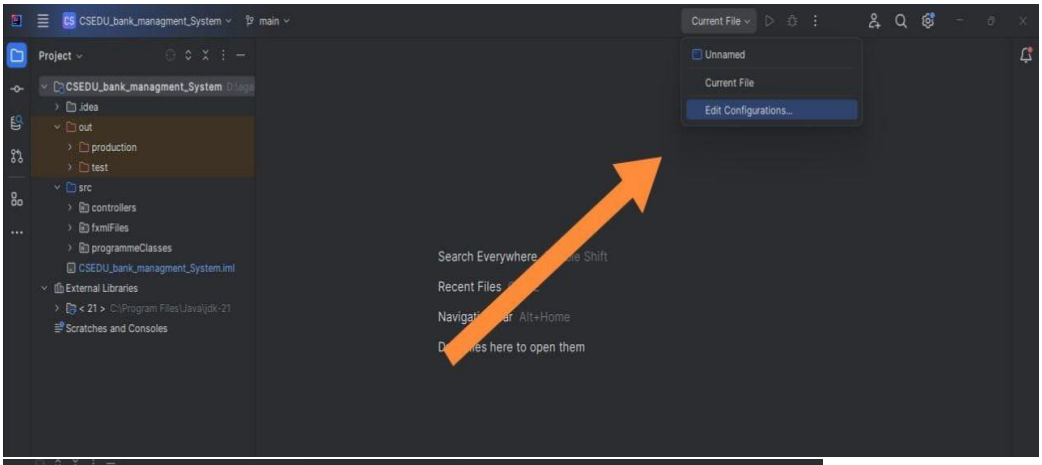
1. **Download the repository from GitHub link or extract folder.**
2. **After opening the folder "CSEDU_BANK_MANAGMENT_SYSTEM" in machine, you need to add three libraries in project structure/ project path. They are Java-JDK_21, two application library 'JAVAFX' and 'MYSQL-Connector'.**
   IntelliJ should look like this –
   

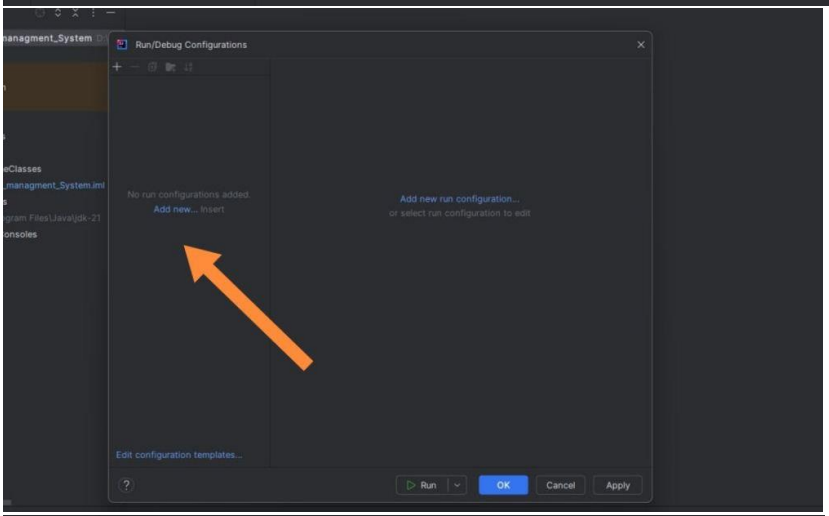
3. **Setup SDK**
   Download JDK21 from JDK-21 Download
   After download and setup JDK in your device add this JDK to project module. Way to do this: File -> Project Structure -> Project -> add JDK -> select the setup location of JDK-21 (in my case: [C:\Program Files\Java\jdk-21]) -> apply -> ok.
4. **Setup JAVAFX**
   Download JavaFX from javaFX downlad
   After extracting the downloaded file there will be some jar files [openjfx-22.0.1_windows-x64_bin-sdk\javafx-sdk-22.0.1\lib] in this location. You need to add those to our project library. Way to do this: File -> Project Structure -> Libraries -> Add new (+) -> java -> select extracted folder jar files (there will be 8 jar files) [path (in

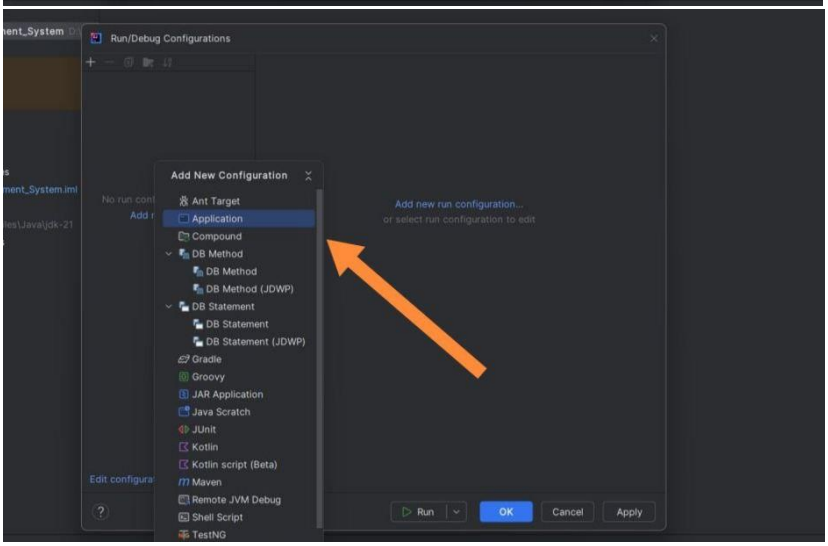
my case E:\openjfx-22.0.1_windows-x64_bin-sdk\javafx-sdk-22.0.1\lib)] -> ok -> apply -> ok.
For more information visit openjfx

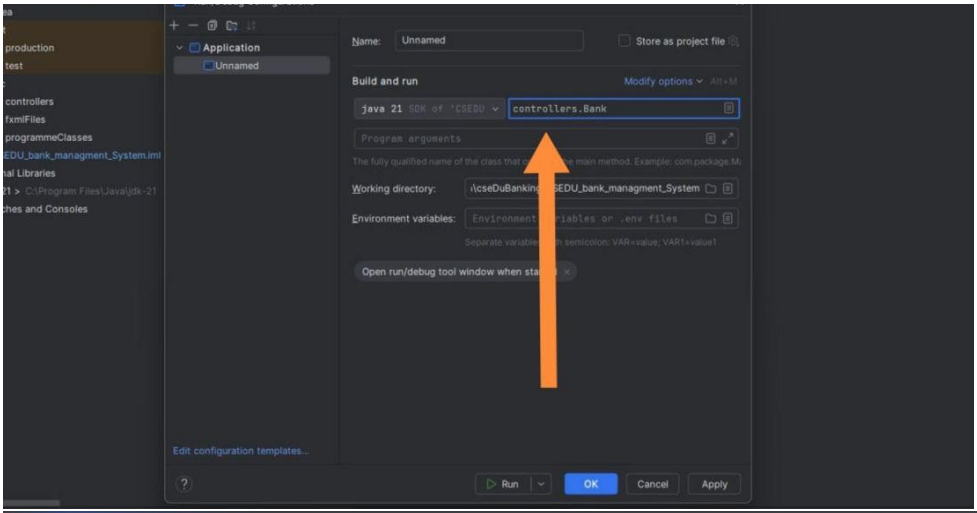
5. **Setup MYSQL-Connector**
   Download connector from mySQL connector download
   After extracting the downloaded connector file (there will be a jar file), way to do this in IntelliJ->
   File -> Project Structure -> Libraries -> Add new (+) -> java -> select extracted folder jar file [path (in my case: D:\sql\mysql-connector-j-8.3.0\mysql-connector-j-8.3.0\mysql-connector-j-8.3.0.jar)] -> ok -> apply -> ok.
6. **Configure VM**
   To run the program, you need to add JavaFX in VM option in configurations. Here is how you can do this in IntelliJ:
   Run/Debug configuration -> Edit Configurations -> Add new -> Application -> setup main function location (in this case controllers.Bank) -> modify options -> add VM option -> VM required command -> apply -> ok.
   VM required command:
   --module-path "your javafx-sdk-22.0.1\lib folder path" --add-modules javafx.controls,javafx.fxml
   Example in my case:
   --module-path "D:\ openjfx-22.0.1_windows-x64_bin-sdk***javafx-sdk-22.0.1\lib***" --add-modules javafx.controls,javafx.fxml

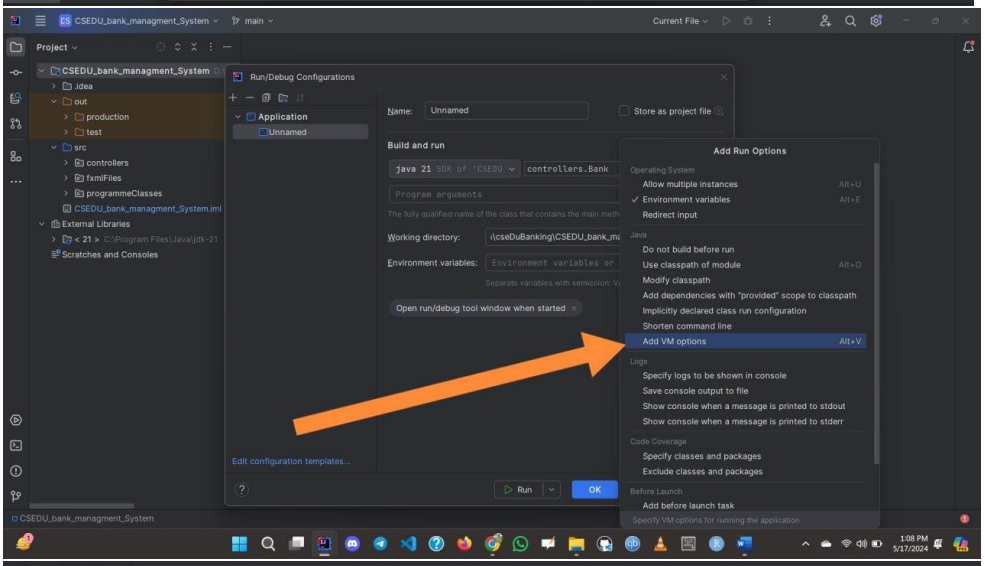
   For more clarifications, I added the steps below.

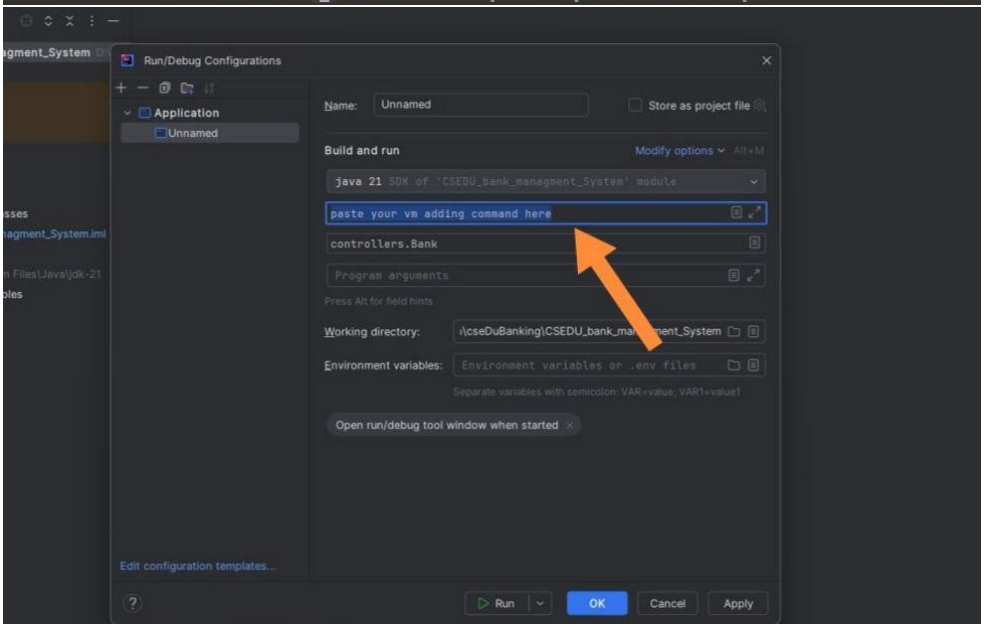

CSEDU_bank_managment_System
main
Current File
Unnamed
Current File
Edit Configurations...
Project
CSEDU_bank_managment_System
.idea
out
production
test
src
controllers
fxmlFiles
programmeClasses
CSEDU_bank_managment_System.iml
External Libraries
< 21 > C:\Program Files\Java\jdk-21
Scratches and Consoles
Search Everywhere
Recent Files
Navigation Bar Alt+Home
Drop files here to open them


Run/Debug Configurations
No run configurations added.
Add new... Insert
Add new run configuration...
or select run configuration to edit
Edit configuration templates...
Run
OK
Cancel
Apply

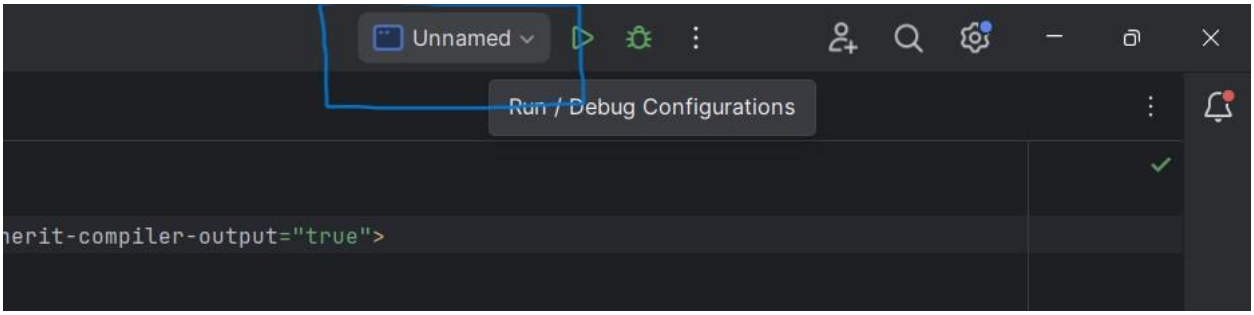

Run/Debug Configurations
Add New Configuration
Ant Target
Application
Compound
DB Method
DB Method
DB Method (JDWP)
DB Statement
DB Statement
DB Statement (JDWP)
Gradle
Groovy
JAR Application
Java Scratch
JUnit
Kotlin
Kotlin script (Beta)
Maven
Remote JVM Debug
Shell Script
TestNG
Add new run configuration...
or select run configuration to edit
Run
OK
Cancel
Apply


Application
Unnamed
Name: Unnamed
Store as project file
Build and run
Modify options Alt+M
java 21 SDK of 'CSEDU
controllers.Bank
Program arguments
Working directory:
Environment variables:
Edit configuration templates...
Run
OK
Cancel
Apply


CSEDU_bank_managment_System
main
Current File
Project
Run/Debug Configurations
Application
Unnamed
Name: Unnamed
Store as project file
Build and run
java 21 SDK of 'CSEDU
controllers.Bank
Program arguments
Working directory:
Environment variables:
Open run/debug tool window when started
Add Run Options
Operating System
Allow multiple instances Alt+U
Environment variables Alt+E
Redirect input
Java
Do not build before run
Use classpath of module Alt+O
Modify classpath
Add dependencies with "provided" scope to classpath
Implicitly declared class run configuration
Shorten command line
Add VM options Alt+V
Logs
Specify logs to be shown in console
Save console output to file
Show console when a message is printed to stdout
Show console when a message is printed to stderr
Code Coverage
Specify classes and packages
Exclude classes and packages
Before Launch
Add before launch task
Specify VM options for running the application
Edit configuration templates...
Run
OK
1:08 PM
5/17/2024


Run/Debug Configurations
Application
Unnamed
Name: Unnamed
Store as project file
Build and run
Modify options Alt+M
java 21 SDK of 'CSEDU_bank_managment_System' module
paste your vm adding command here
controllers.Bank
Program arguments
Press Alt for field hints
Working directory:
Environment variables:
Environment variables or .env files
Separate variables with semicolon: VAR=value; VAR1=value1
Open run/debug tool window when started
Edit configuration templates...
Run
OK
Cancel
Apply

7. **Run the attached SQL file in MySQL query for creating database and required tables.**
8. **Configure username and password in src/controllers/Bank class:**
    a. At the 19th line of Bank class replace with your info – connectDB("jdbc:mysql://localhost:3306/CSEDU_BANK", "*your_my_sql_connection_username (example: root)*", "*password_of_the_connection (example: 12345678)*");
9. **Run the application**
    a. In IntelliJ you can simply click the unnamed (You can add Bank name when configuring VM. Else it will be 'unnamed' by default)/Bank option above of the screen.



If there are any issues let me know.

==========END==========