***Alexandria University***

***Faculty of Engineering***

***Computer and Systems Engineering Dept. CS:***

***PROGRAMMING 2***

---

# Assignment 4: JDBC API

Names:

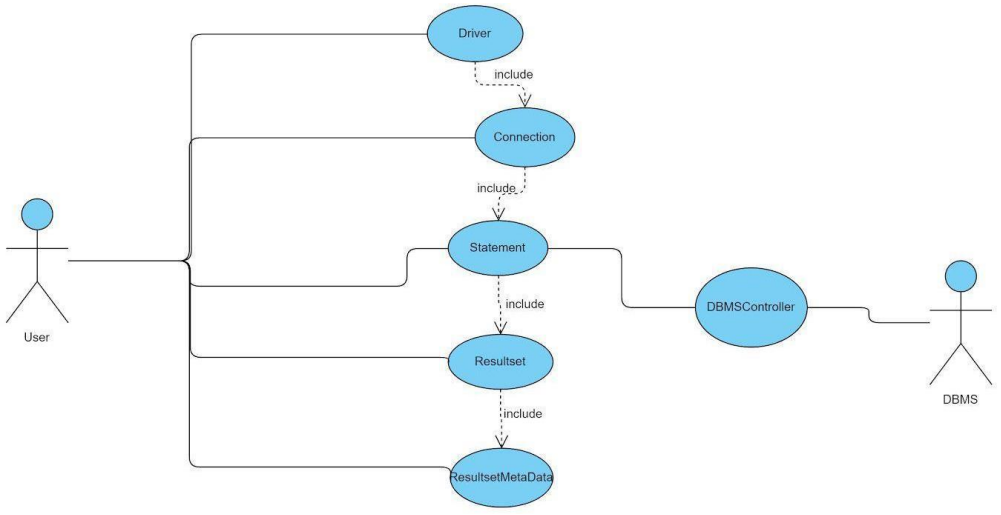
1. Aya Gamal (1)
2. Ahmed Talaat (5)
3. Salma Ragab (32)
4. Linh Ahmed (50)

# Description:

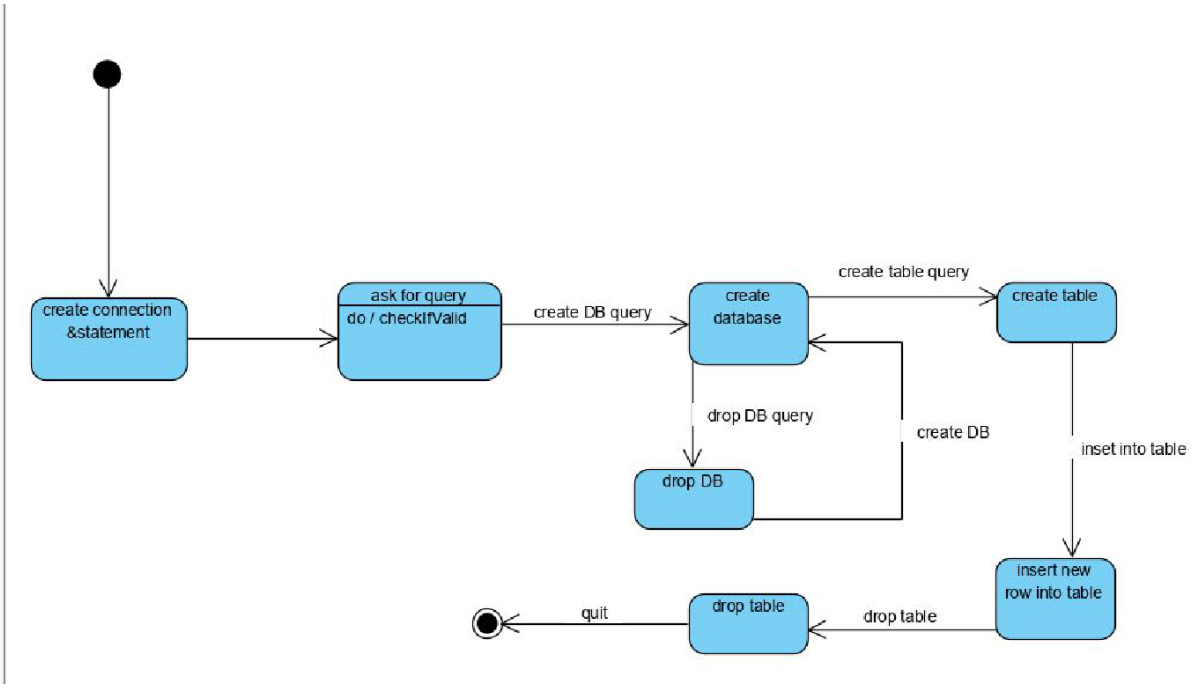
Java Database Connectivity (JDBC) is an application programming interface (API) which allows the programmer to connect and interact with databases. It provides methods to query and update data in the database through update statements like SQL's CREATE, UPDATE, DELETE and INSERT and query statements such as SELECT.

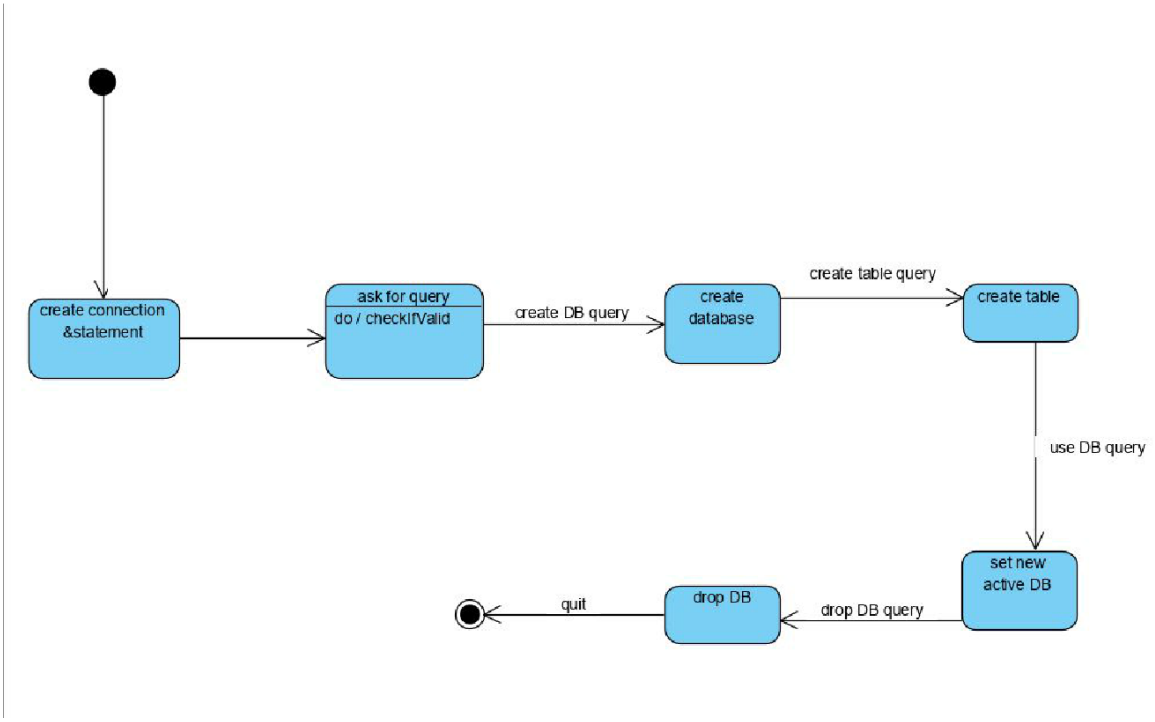
# UML Diagram:

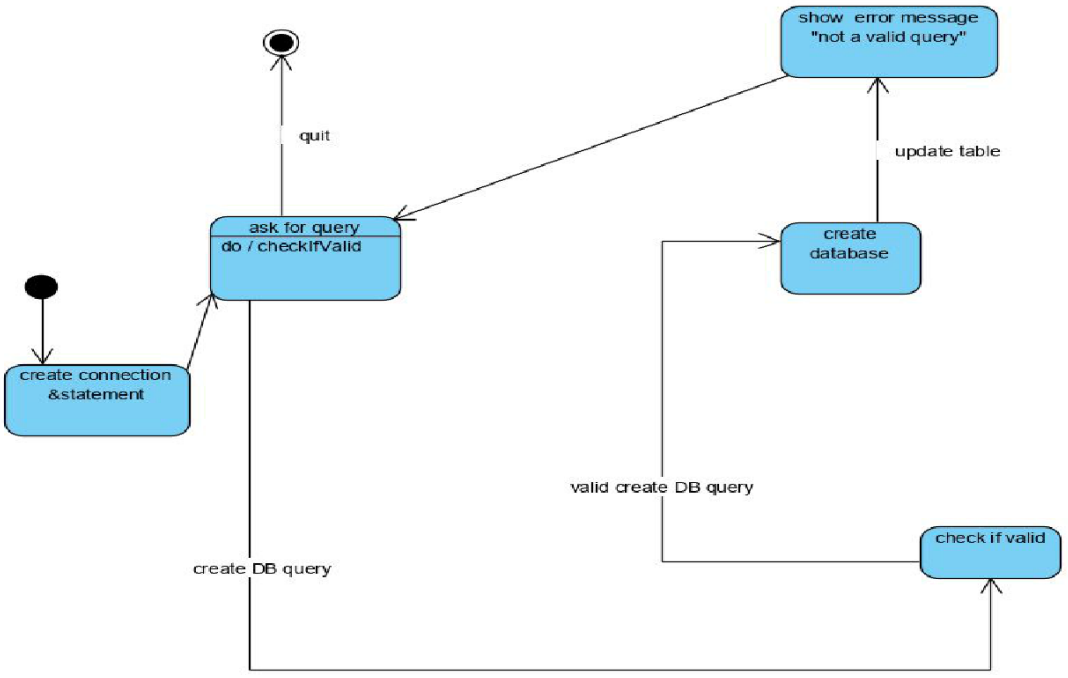
## Use Case:

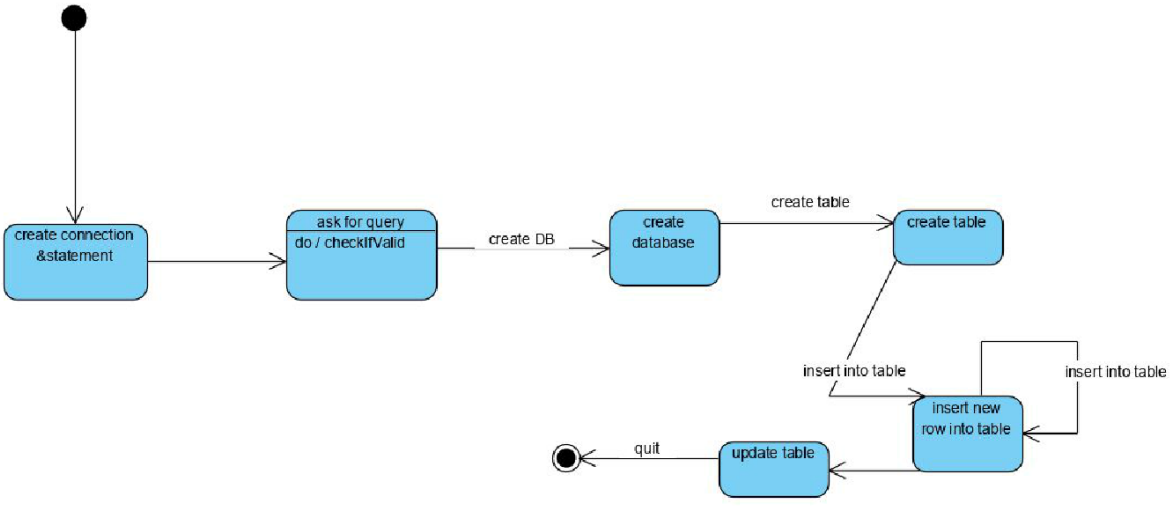


# State Diagram for 3 scenarios:

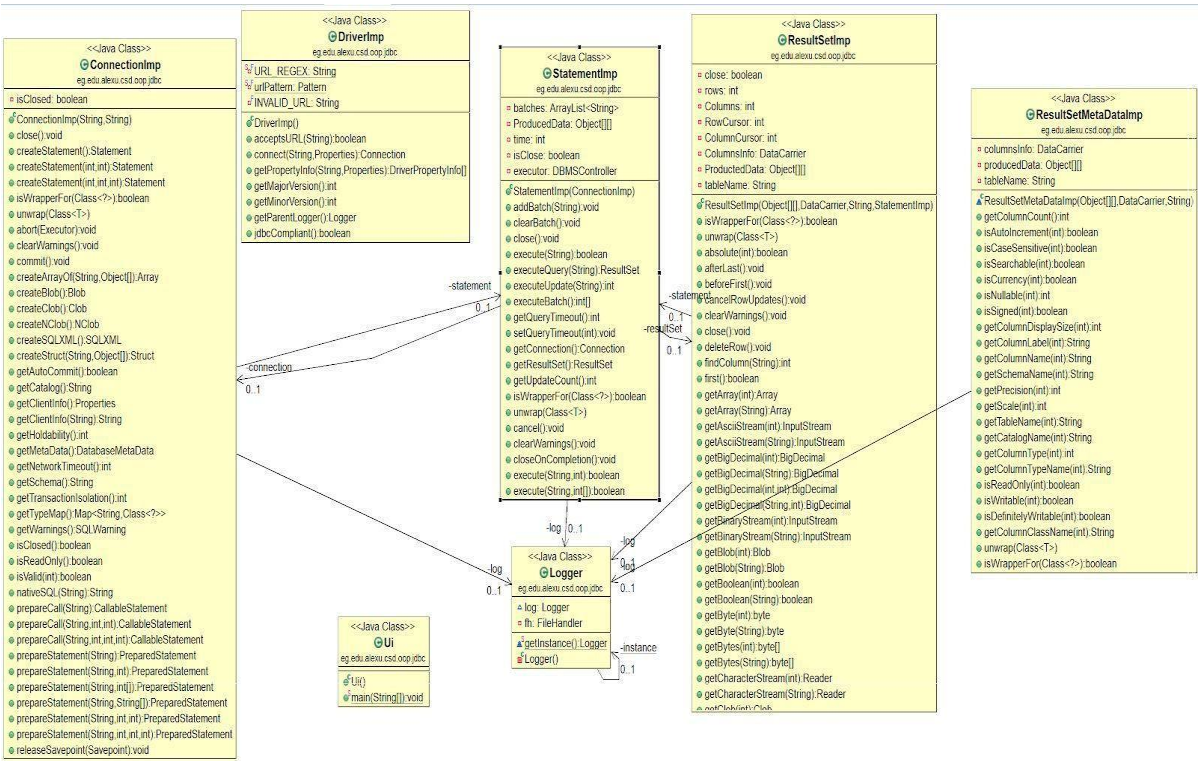


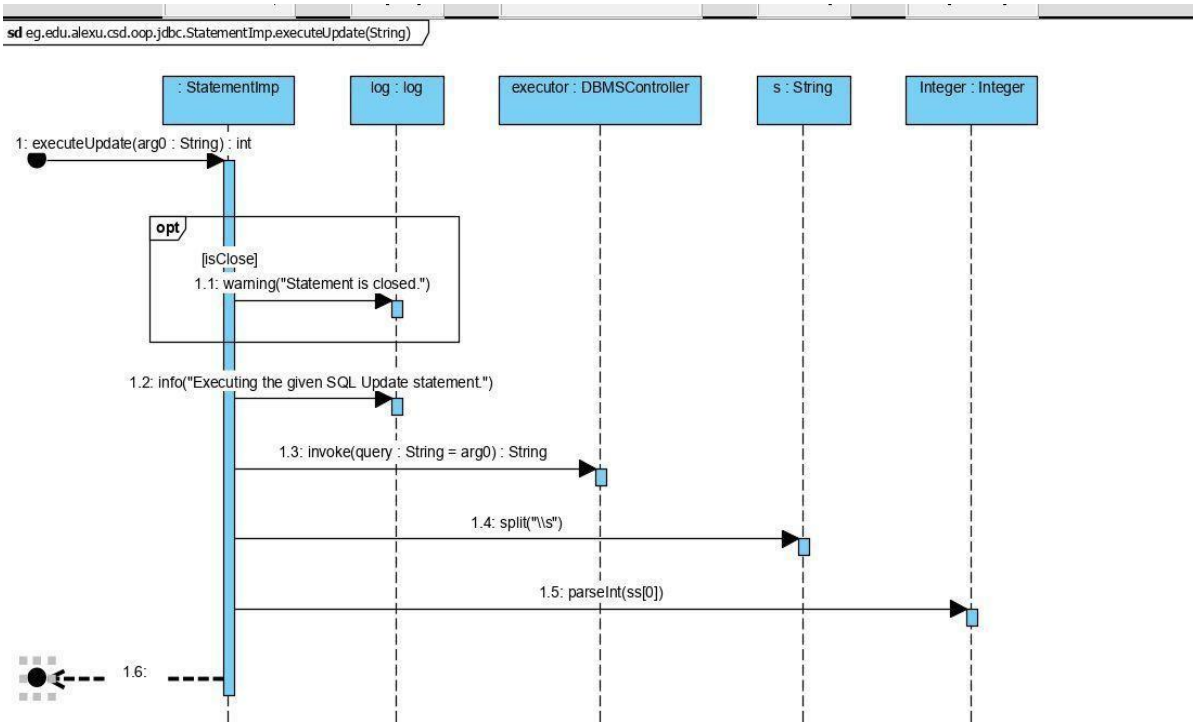

show error message
"not a valid query"
quit
update table
ask for query
do / checkIfValid
create
database
create connection
&statement
valid create DB query
check if valid
create DB query

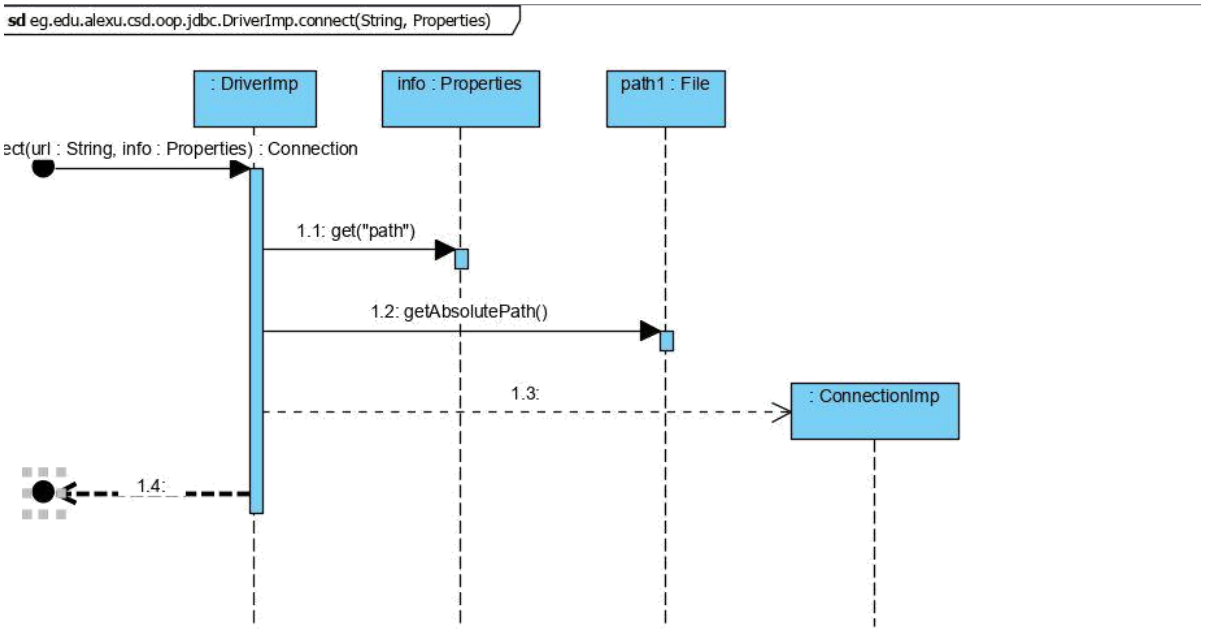

create connection
&statement
ask for query
do / checkIfValid
create DB
create
database
create table
create table
insert into table
insert into table
insert new
row into table
quit
update table

# Detailed class Diagram:




# Sequence Diagram:






# Design Patterns:

## Singleton Design Pattern:

Used in logger

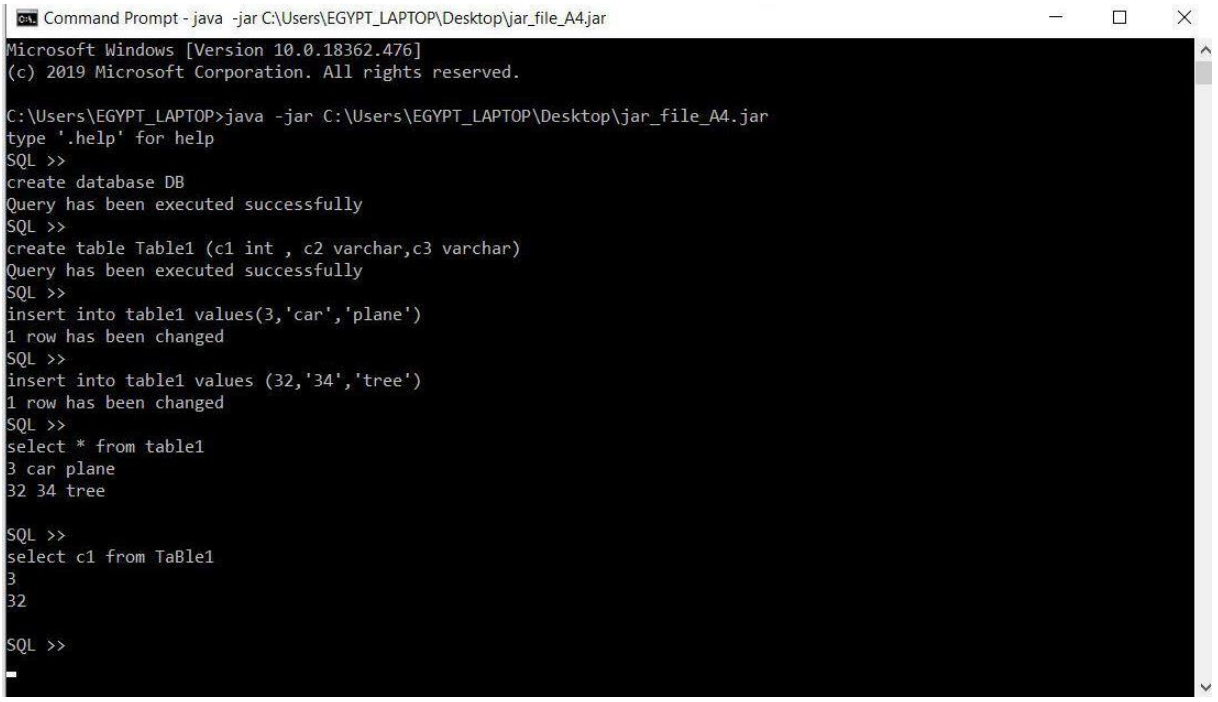
## Object Pool Design Pattern:

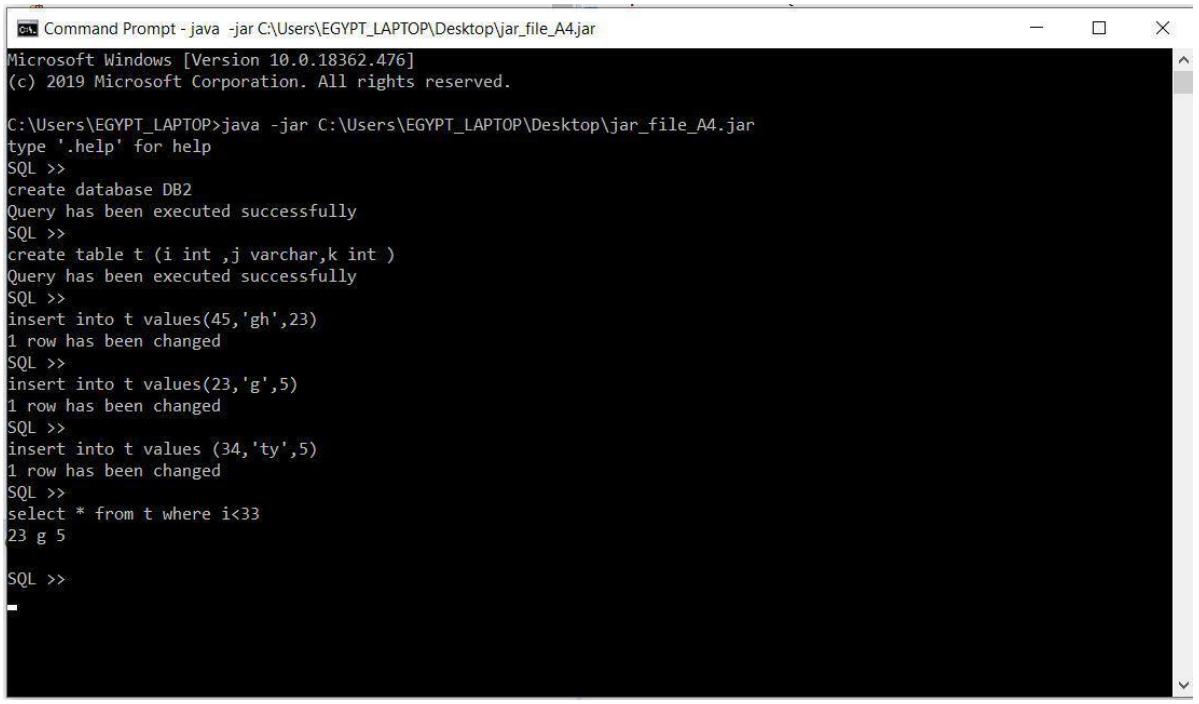
Used in Connection Pool.

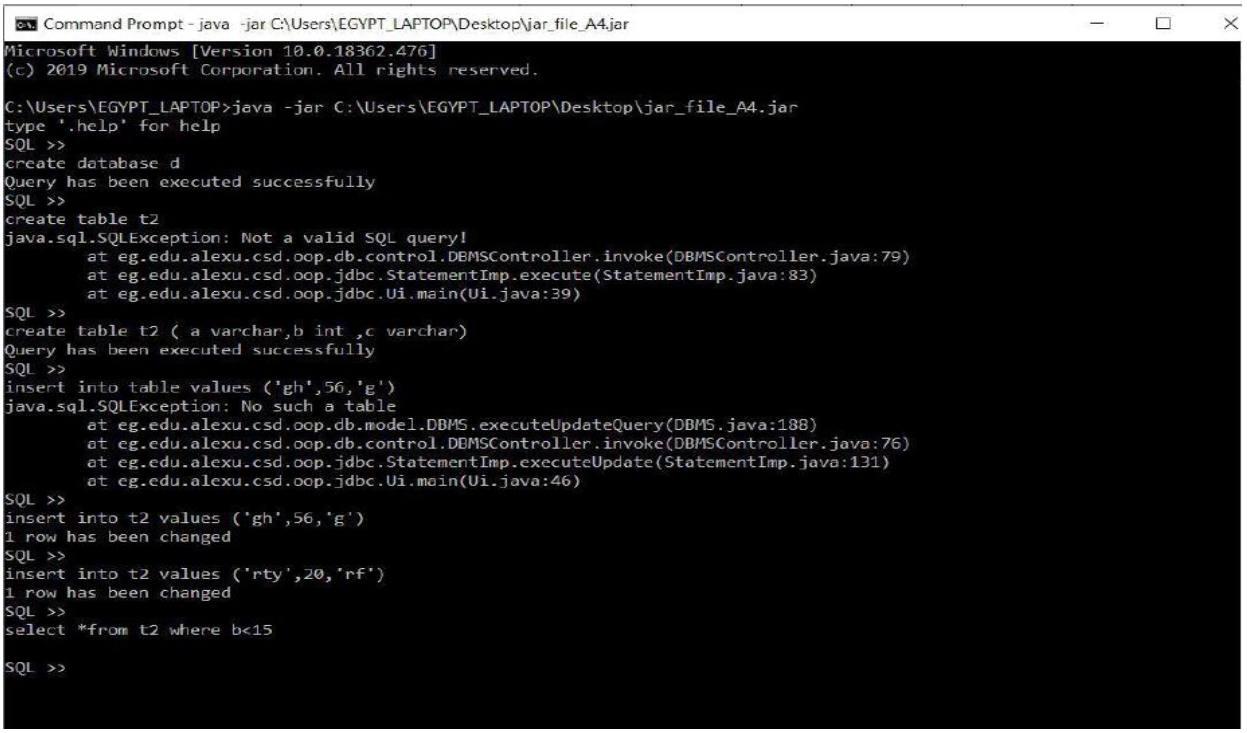
# Design decisions and Assumptions:

- DriverImp:
  Get the connection to database after accepting valid URL and throws exception in case of not valid ones.
- ConnectionImp:
  Creates statement object which should execute queries.
- StatementImp:
  Executes queries and get ResultSet.
- ResultsetImp:
  It contains selected data from current table.
- ResultSetMetaDataImp:
  Get information about data in ResultSet.
- Logger:
  writing log messages during the execution of a program to a central place. Those messages include warning messages as well as info messages so that the messages can later be retrieved and analyzed.

# Snapshots:


Command Prompt - java -jar C:\Users\EGYPT_LAPTOP\Desktop\jar_file_A4.jar
Microsoft Windows [Version 10.0.18362.476]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\EGYPT_LAPTOP>java -jar C:\Users\EGYPT_LAPTOP\Desktop\jar_file_A4.jar
type '.help' for help
SQL >>
create database DB
Query has been executed successfully
SQL >>
create table Table1 (c1 int , c2 varchar,c3 varchar)
Query has been executed successfully
SQL >>
insert into table1 values(3,'car','plane')
1 row has been changed
SQL >>
insert into table1 values (32,'34','tree')
1 row has been changed
SQL >>
select * from table1
3 car plane
32 34 tree
SQL >>
select c1 from TaBle1
3
32
SQL >>


Command Prompt - java -jar C:\Users\EGYPT_LAPTOP\Desktop\jar_file_A4.jar
Microsoft Windows [Version 10.0.18362.476]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\EGYPT_LAPTOP>java -jar C:\Users\EGYPT_LAPTOP\Desktop\jar_file_A4.jar
type '.help' for help
SQL >>
create database DB2
Query has been executed successfully
SQL >>
create table t (i int ,j varchar,k int )
Query has been executed successfully
SQL >>
insert into t values(45,'gh',23)
1 row has been changed
SQL >>
insert into t values(23,'g',5)
1 row has been changed
SQL >>
insert into t values (34,'ty',5)
1 row has been changed
SQL >>
select * from t where i<33
23 g 5
SQL >>


Command Prompt - java -jar C:\Users\EGYPT_LAPTOP\Desktop\jar_file_A4.jar
Microsoft Windows [Version 10.0.18362.476]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\EGYPT_LAPTOP>java -jar C:\Users\EGYPT_LAPTOP\Desktop\jar_file_A4.jar
type '.help' for help
SQL >>
create database d
Query has been executed successfully
SQL >>
create table t2
java.sql.SQLException: Not a valid SQL query!
at eg.edu.alexu.csd.oop.db.control.DBMSController.invoke(DBMSController.java:79)
at eg.edu.alexu.csd.oop.jdbc.StatementImp.execute(StatementImp.java:83)
at eg.edu.alexu.csd.oop.jdbc.Ui.main(Ui.java:39)
SQL >>
create table t2 ( a varchar,b int ,c varchar)
Query has been executed successfully
SQL >>
insert into table values ('gh',56,'g')
java.sql.SQLException: No such a table
at eg.edu.alexu.csd.oop.db.model.DBMS.executeUpdateQuery(DBMS.java:188)
at eg.edu.alexu.csd.oop.db.control.DBMSController.invoke(DBMSController.java:76)
at eg.edu.alexu.csd.oop.jdbc.StatementImp.executeUpdate(StatementImp.java:131)
at eg.edu.alexu.csd.oop.jdbc.Ui.main(Ui.java:46)
SQL >>
insert into t2 values ('gh',56,'g')
1 row has been changed
SQL >>
insert into t2 values ('rty',20,'rf')
1 row has been changed
SQL >>
select *from t2 where b<15
SQL >>

# User Guide:

## Running the application:

- It is a console application.
- In the cmd, type this command: Java -jar (path of the jar file).
- The application should run inside your cmd and you will see “SQL >>”.
- Then type any of the supported SQL queries.

## Supported SQL Queries:

1. Create database
2. Create table
3. Insert into table
4. Delete from table
5. Drop database
6. Drop table
7. Select from table
8. Update table
9. Use Database (bonus).

## SQL Queries References:

1. W3 Schools: https://www.w3schools.com/sql/
2. Oracle: https://docs.oracle.com/javase/tutorial/jdbc/