ASSIGNMENT - 07

PROBLEM STATEMENT :-

Develop an application using JDBC, multi-threading concurrency, synchronous and asynchronous call backs, thread ports using executor service

OBJECTIVE :-

1. To understand database connectivity
2. To understand multithreading & connectivity.

OUTCOMES :-

One will be able to implement JDBC drivers
Apply multithreading in any application

THEORY :-

JDBC is a Java API to connect and execute queries from a database.
To establish connection with a database using Java, one must use JDBC drivers
Thereafter 4 types of JDBC driver in Java
type 1, 2, 3, 4

Type 1 :-
This is the oldest type of driver.
Converts JDBC calls to JDBC calls
Had poor performance due to layers.

Type - 2
Reduced layers.
NO JDBC, direct, communication
Better performance

Type - 03
Complexity written in Java
User - Client, Server and Database approach.
Known as net protocol JDBC driver

Type - 04
Includes all DB calls in a JAR file.
Known as thin JDBC driver
Portable across all platforms.

Multithreading

In this process a new thread is assigned for every new task that has to be called
Thread lifecycle is as follows.

Newborn
Running
Runnable
Blocked
Dead.

Thread Pods:

Thread pods refer to usage of fixed amount of threads repeatedly for multiple tasks

Algorithms

1) JDBC - importing package.
   → load - register drivers
   → establish connection
   → create statement
   → execute statement
   → close objects
   → accept connections
   → connect to socket
   → communicate through socket programming
   → close sockets

CONCLUSION

Thus we were able to implement a multi chat server architecture model using JDBC & concurrency.