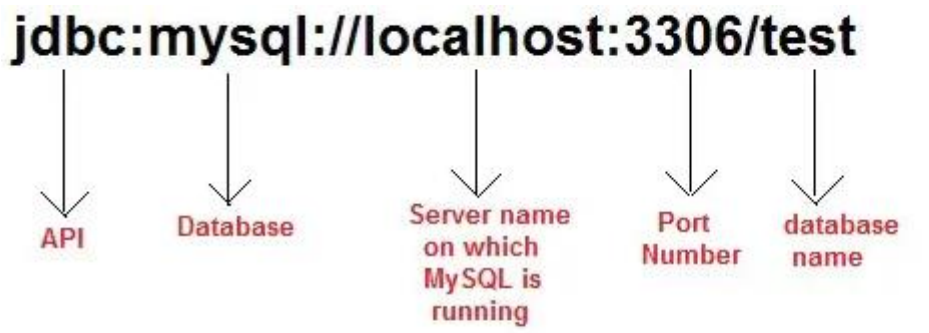
Connecting to MySQL Database using Thin Driver

To connect a Java application with MySQL database using Thin Driver. You need to follow the following steps:

Load Driver Class: The Driver Class for MySQL database is com.mysql.jdbc.Driver and Class.forName("com.mysql.jdbc.Driver") method is used to load the driver class for MySQL database.

Create Connection: For creating a connection you will need a Connection URL. The Connection URL for MySQL is:


jdbc:mysql://localhost:3306/test
API
Database
Server name on which MySQL is running
Port Number
database name

Example

1. Create a table in MySQL Database

Use the below SQL command to create a table Student into mysql database. This table has two columns: sid and name.

create table Student(sid int(10),name varchar(20));

2. Insert some record into the table

After creating the table now insert two records into table using the below sql command. Here we are inserting adam and abhi named two rows.

insert into Student values(102,'adam');

insert into Student values(103,'abhi');

Example1: Inserting , Updating, Deleting Records in Database

for "insert, update, delete" steps to follow,
1) Create a connetion
2) Create statement/query
3) Execute statament/query
4) close connection

```
import java.sql.*;
class DBConnection {
    public static void main(String[] args) throws SQLException,
ClassNotFoundException{
        Connection con = null;
        Statement stmt = null;

        try{
            // Load the MySQL JDBC Driver
            Class.forName("com.mysql.jdbc.Driver");

            // 1) Create a connetion

    con=DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db
", "root", "qwerty");

            // 2) Create Statement/Query
            stmt = con.createStatement();
            String sqlq1 = "INSERT INTO  companies VALUES(102,
'Abhishek', 'ABC', 'abc@xyz.com', 0000007890)";
            String sqlq2 = "update companies SET name='Dhruv' where
id=101";
            String sqlq3 = "delete from companies where id=102";

            // 3) Execute statament/query
            stmt.execute(sqlq2);
        }
        catch(ClassNotFoundException e){
            System.out.println("MySQL JDBC Driver not found");
            e.printStackTrace();
        }
        catch(SQLException e){
            System.out.println("Connection Failed!");
            e.printStackTrace();
        }
        finally{
            try {
                // 4) Close connection
                con.close();
                System.out.println("Connection CLosed");
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
        System.out.println("Data Updated Successfully");
    }
}
```

Example2: Accessing records from Database

for "select" steps to follow,
1) Create a connetion
2) Create statement/query
3) Execute statament/query
4) Store the result in result set
5) close connection

```
import java.sql.*;
class DBConnection {
    public static void main(String[] args) throws SQLException,
ClassNotFoundException{
        Connection con = null;
        Statement stmt = null;

        try{
            // Load the MySQL JDBC Driver
            Class.forName("com.mysql.jdbc.Driver");

            // 1) Create a connetion

    con=DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db
", "root", "qwerty");

            // 2) Create Statement/Query
            stmt = con.createStatement();
            String sqlq = "select * from companies";

            // 3) Execute statament/query
            ResultSet rs = stmt.executeQuery(sqlq);
            while(rs.next()){
                int ID = rs.getInt("id");
                String name = rs.getString("name");
                String address = rs.getString("address");
                String email = rs.getString("email");
                long phone = rs.getLong("phone");
                System.out.println(ID+" "+name+" "+address+"
"+email+" "+phone);
            }
        }
        catch(ClassNotFoundException e){
            System.out.println("MySQL JDBC Driver not found");
            e.printStackTrace();
        }
        catch(SQLException e){
            System.out.println("Connection Failed!");
            e.printStackTrace();
        }
        finally{
            try {
                // 4) Close connection
                con.close();
                System.out.println("Connection CLosed");

            } catch (Exception e) {
                e.printStackTrace();
            }
        }
        System.out.println("Data Updated Successfully");
    }
}
```