**JDBC**

**Steps:**

1. **Load and register JDBC driver class.**

This can be done by using either Class.forName() or DriverManager.registerDriver().

Class.forName("com.mysql.cj.jdbc.Driver");

OR

Driver jdbcDriver = new com.mysql.cj.jdbc.Driver ();

DriverManager.registerDriver(jdbcDriver);

1. **Establish connection between Java application and database**

Connection con=

DriverManager.*getConnection*("jdbc:mysql://localhost:3306/infosys","root","");

//(URL, username, password)

//here "infosys" is database name, "root" is username, and password is none

| **Database** | **Driver Class** | **URL** |
| --- | --- | --- |
| Oracle | oracle.jdbc.driver.OracleDriver | jdbc:oracle:thin:@hostname:portnumber:databasename |
| Derby | org.apache.derby.jdbc.ClientDriver | jdbc:derby://hostname:port//databasename |
| MySQL | com.mysql.jdbc.Driver | jdbc:mysql://hostname:port//databasename |

URL formats of some Databases:

1. **Create a statement object.**

Statement stmt=con.createStatement();

| **Methods** | **Description** |
| --- | --- |
| createStatement() | Creates a Statement Object for an SQL query. |
| prepareStatement(String sql) | Creates a PreparedStatement object with an SQL query; It is a precompiled SQL query with or without parameters. |
| prepareCall(String sql) | Creates a CallableStatement object with an SQL call to a stored procedure. |
| close() | Closes the database connection. |

SQL statements can be created using the connection object. Some methods provided by the Connection interface are:

1. **Send and execute the SQL query.**

ResultSet rs=stmt.executeQuery("select \* from employee");

Here, employee is table-name.

Once we have a statement object, we can use it to send and execute SQL queries over the connection object.

| **Methods** | **Description** |
| --- | --- |
| executeQuery(String sql) | Executes an SQL statement (SELECT) that queries a database and returns a ResultSet object. |
| executeUpdate(String sql) | Executes an SQL statement (INSERT, UPDATE or DELETE) that updates the database and returns an integer, the row count associated with the SQL statement. |
| execute(String sql) | Can execute all kinds of SQL statements (SELECT, INSERT, UPDATE or DELETE). |
| getResultSet() | Retrieve the ResultSet object. |

Some methods provided by the Statement interface are:

1. **Process the result from ResultSet.**

**while**(rs.next())

System.***out***.println(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3));

The java.sql.ResultSet interface denotes a database query result.

ResultSet objects always keep a cursor pointing to the current row of the result set. It has methods for navigating, viewing, and updating the data.

Commonly used navigational methods of ResultSet are:

| **Methods** | **Description** |
| --- | --- |
| boolean next() | Moves the cursor from the current position to the next row. |
| boolean previous() | Moves the cursor from the current position to the previous row. |
| boolean first() | In the result set object, the cursor is placed in the first row. |
| boolean last() | In the result set object, the cursor is placed in the last row. |
| boolean absolute(int row) | In the result set object, the cursor is moved to the specified row number. |
| boolean relative(int row) | In the result set object, the cursor is moved to the relative row number, it can be positive or negative. |

1. **Close the statement and connection**

stmt.close();

con.close();

**Steps to Download MySQL:**

If not installed then follow the below steps for installation of the MySQL,

Steps-

1. Go to the software center and search for MySQL.

2. Select ‘MySQL Server 5.6.20’.

3. Click install to get installed on your machine.

Once the files get installed then follow the below steps-

1. Go to the program files folder in the c drive.
2. Open the MySQL folder and copy the ‘MySQL server 5.6.20’ folder.
3. Paste the above-selected folder into the D drive, then open it.
4. Inside that go to the bin folder and open the command prompt(cmd) in that location.
5. In CMD type mysql -u root -p and press enter key on the keyboard.
6. As we have not set any password so do not enter any password, just press enter key on the keyboard once again.
7. Then finally you are into the MySQL database and ready to create the databases using the appropriate MySQL commands.

Note- In all the lex demos projects which are related to the MySQL database has the default password set as ‘root’. So please remove that password before you run the application.

You will find this configuration data i.e. username and password in application.properties file under the resources folder.

Next step🡪 Once the database is running you can copy the tablescript data from the tablescript file under the resources folder and paste it into the database.

Once the tablescript is running do not close the database. Then run the application for successful execution.

**Run MySQL:**

First run ‘mysqld’ application file in ‘MySQL Server 5.6/bin’ folder .

Then open command prompt for ‘MySQL Server 5.6/bin’ folder.

mysql -u root -p

Password: (No password)

**Few MySQL Queries:**

**1. Create Database:**

create database infosys;

**2. Use/select database:**

use infosys;

**3. Create table:**

CREATE TABLE employee (id int(10), name varchar(50), city varchar(50), PRIMARY KEY (id ));

**4. Insert into a table:**

insert into employee values(1238022, 'Chandrakant','Pune');

**5. Select (print) all the records from a table:**

SELECT \* from employee;

**6. Update:**

update employee set name='XYZ', city='PQR' where id=1238022;

**7. Delete a row:**

delete from employee where id=1238022;

**8. Truncate (delete) table:**

truncate table employee;

It removes all the records of a table, but it doesn't remove structure.

**9. Drop table:**

drop table customers;

It removes all the records as well as structure of a table