

# Java Database Connectivity (JDBC)

Upender Yadav

Department Of Computer Science and Engineering,  
Dronacharya College of Engineering, Gurgaon

**Abstract-** Java Database Connectivity(JDBC) is an application programming interface(API) for the programming languages Java,that defines how a client may access a database. It is part of the Java Standard Edition platform,from Oracle Corporation. It provides methods to query and update data in a database.

## I. STEPS IN JDBC

- Loading Driver
- Establishing Connection
- Executing Statements
- Getting Results
- Closing Database Connection

### 1. Loading Driver

```
Connection con=null;
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

### 2. Establishing Connection

```
con=DriverManager.getConnection  
("jdbc:odbc:table1", " ", " ");
```

### 3. Executing Statements

```
Statement statement=con.createStatement();  
ResultSet rs=statement.executeQuery("select * from  
tab1");
```

### 4. Getting Results

```
while(rs.next())  
{  
out.println("\n"+rs.getString("id")+"\t");  
out.println("\nh"+rs.getString("name")+"\t");  
;  
out.println("\n"+rs.getString("roll")+"\t");  
out.println("\n"+rs.getString("price")+"\t");  
}
```

### 5. Closing Database Connection

```
rs.close();
```

## II. JDBC DRIVERS

- Type 1 (JDBC-ODBC Bridge Technology)
- Type 2 (JNI drivers for C/C++ connection libraries)

- Type 3 (Socket-level Middleware Translator)
- Type 4 (Pure Java-DBMS driver)

Type 1 (JDBC-ODBC Bridge Technology):

JDBC driver translates call into ODBC and redirects ODBC call to an ODBC driver on the DBMS.ODBC binary code must exist on every client.Translation layer compromises execution speed to small degree.

Type 2 (JNI drivers for C/C++ connection libraries):  
Java driver makes JNI calls on the client API (usually written in C or C++)

eg: Sybase dblink or ctlib

eg: Oracle Call Interface libs (OCI)

Requires client-side code to be installed,Often the fastest solution available.Native drivers are usually delivered by DBMS vendor.there are bug in driver can crash JVMs

Example: JDBC=>Sybase dblink or ctlib

Type 3 (Socket-level Middleware Translator):

JDBC driver translates JDBC calls into a DBMS-independent protocol.Then, communicates over a socket with a middleware server that translates Java code into native API DBMS calls,No client code need be installed.

Single driver provides access to multiple DBMSs, eg. WebLogic Tengah drivers.Type 3 drivers auto-download for applets.

Type 4 (Pure Java-DBMS driver):

Java drivers talk directoy to the DBMS using Java sockets.No Middleware layer needed, access is direct.There is a Simplest solution available.In this No client code need be installed.

Example: JConnect for Sybase.

### III. JDBC EXAMPLE

```
package com.java2novice.jdbc;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
public class MyExecuteMethod {
public static void main(String a[])
{
Connection con = null;
    try {
        Class.forName("oracle.jdbc.driver.OracleDriver"
);
con=DriverManager.getConnection("jdbc:oracle:thin:
@<hostname>:<port num>:<DB
name>", "user", "password");
Statement stmt = con.createStatement();
    String query = "select * from emp";
    boolean status =stmt.execute(query);
    if(status)
{
        ResultSet rs = stmt.getResultSet();
        while(rs.next()){
            System.out.println(rs.getString(1));
        }
        rs.close();
    } else {
        int count = stmt.getUpdateCount();
        System.out.println("Total records
updated:"+count);
    }
}
catch (ClassNotFoundException e) {
    e.printStackTrace();
}
catch (SQLException e) {
    e.printStackTrace();
}
finally{
try{
    if(con != null) con.close();
} catch (Exception ex){ }
}
}
```

### REFERENCES

<http://www.easysoft.com/blog/java-8.html>  
<https://en.wikipedia.org/>  
JAVA: The complete reference