



## Structure of JDBC

## 1.0 What Is JDBC?

Working with leaders in the database field, Sun developed a single API for database access called JDBC. As part of this process, they kept three main goals in mind:

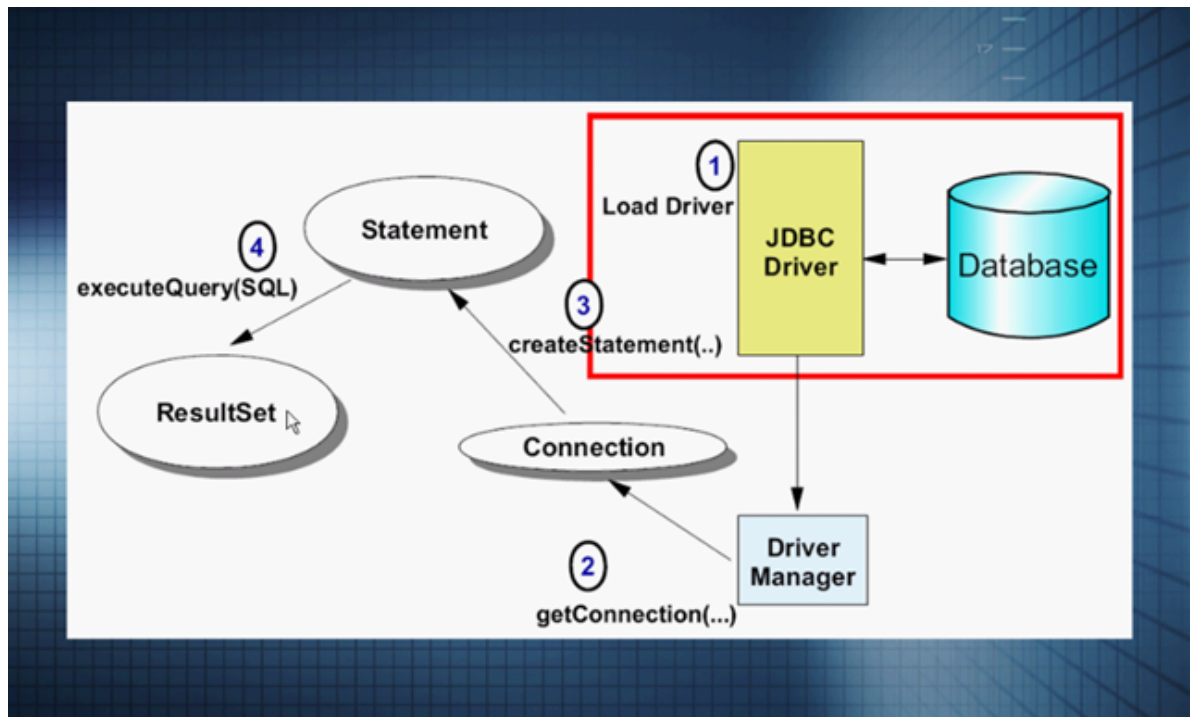
- JDBC should be a SQL-level API.
- JDBC should capitalize on the experience of existing database APIs.
- JDBC should be simple.

A SQL-level API means that JDBC allows you to construct SQL statements and embed them inside Java API calls. In short, you are basically using SQL. But JDBC lets you smoothly translate between the world of the database and the world of the Java application. Your results from the database, for instance, are returned as Java objects, and access problems get thrown as exceptions.

## 1.1 What Does the JDBC API Do?

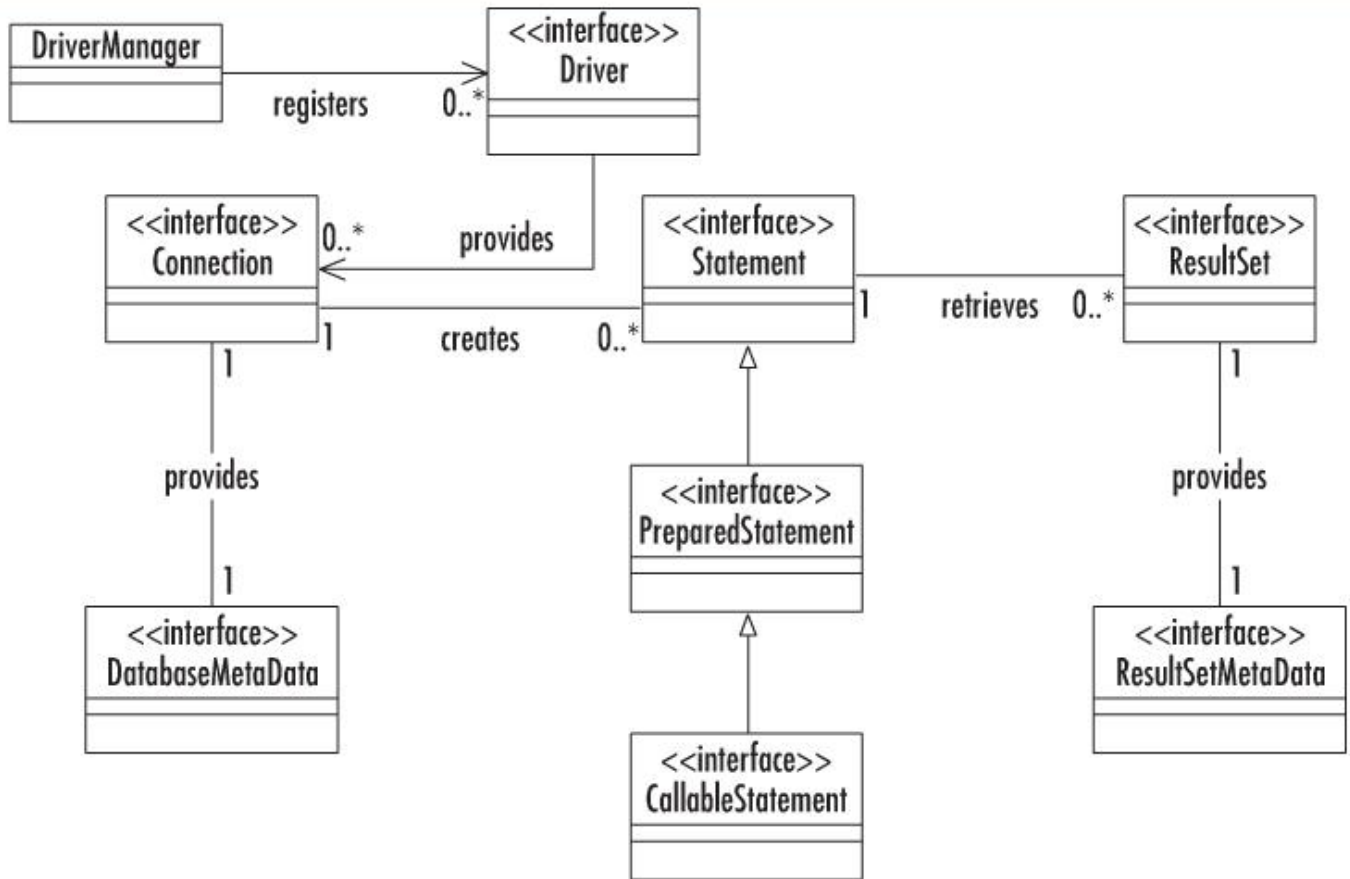
In simplest terms, a JDBC API makes it possible to do the following things:

1. Load the appropriate driver for the data source
2. Establish a connection with a data source
3. Send queries and update statements to the data source
4. Process the results



## 1.2 The Structure of JDBC

JDBC API is found in java.sql packages. The following self explanatory diagram shows the relationship between the JDBC interfaces



### 1.3 References

<http://java.sun.com/developer/onlineTraining/Database/JDBC20Intro/JDBC20.html#Introduction>

<http://www.roseindia.net/jdbc/what-is-jdbc.shtml>

[http://www.tomgerber.com/jdbc\\_desc.html](http://www.tomgerber.com/jdbc_desc.html)

[http://onjava.com/onjava/excerpt/javaentnut\\_2/index1.html](http://onjava.com/onjava/excerpt/javaentnut_2/index1.html)