Java JDBC Tutorial

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database. There are four types of JDBC drivers:

* JDBC-ODBC Bridge Driver,
* Native Driver,
* Network Protocol Driver, and
* Thin Driver

We have discussed the above four drivers in the next chapter.

We can use JDBC API to access tabular data stored in any relational database. By the help of JDBC API, we can save, update, delete and fetch data from the database. It is like Open Database Connectivity (ODBC) provided by Microsoft.



The current version of JDBC is 4.3. It is the stable release since 21st September, 2017. It is based on the X/Open SQL Call Level Interface. The **java.sql** package contains classes and interfaces for JDBC API. A list of popular *interfaces* of JDBC API are given below:

* Driver interface
* Connection interface
* Statement interface
* PreparedStatement interface
* CallableStatement interface
* ResultSet interface
* ResultSetMetaData interface
* DatabaseMetaData interface
* RowSet interface

A list of popular *classes* of JDBC API are given below:

* DriverManager class
* Blob class
* Clob class
* Types class

Why Should We Use JDBC

Before JDBC, ODBC API was the database API to connect and execute the query with the database. But, ODBC API uses ODBC driver which is written in C language (i.e. platform dependent and unsecured). That is why Java has defined its own API (JDBC API) that uses JDBC drivers (written in Java language).

We can use JDBC API to handle database using Java program and can perform the following activities:

1. Connect to the database
2. Execute queries and update statements to the database
3. Retrieve the result received from the database.

Do You Know

* How to connect Java application with Oracle and Mysql database using JDBC?
* What is the difference between Statement and PreparedStatement interface?
* How to print total numbers of tables and views of a database using JDBC?
* How to store and retrieve images from Oracle database using JDBC?
* How to store and retrieve files from Oracle database using JDBC?

What is API

API (Application programming interface) is a document that contains a description of all the features of a product or software. It represents classes and interfaces that software programs can follow to communicate with each other. An API can be created for applications, libraries, operating systems, etc.

Topics in Java JDBC Tutorial

[2) JDBC Drivers](https://www.javatpoint.com/jdbc-driver)

In this JDBC tutorial, we will learn four types of JDBC drivers, their advantages and disadvantages.

[3) 5 Steps to connect to the Database](https://www.javatpoint.com/steps-to-connect-to-the-database-in-java)

In this JDBC tutorial, we will see the five steps to connect to the database in Java using JDBC.

[4) Connectivity with Oracle using JDBC](https://www.javatpoint.com/example-to-connect-to-the-oracle-database)

In this JDBC tutorial, we will connect a simple Java program with the Oracle database.

[5) Connectivity with MySQL using JDBC](https://www.javatpoint.com/example-to-connect-to-the-mysql-database)

In this JDBC tutorial, we will connect a simple Java program with the MySQL database.

[6) Connectivity with Access without DSN](https://www.javatpoint.com/connectivity-with-access-without-dsn)

Let's connect java application with access database with and without DSN.

[7) DriverManager class](https://www.javatpoint.com/DriverManager-class)

In this JDBC tutorial, we will learn what does the DriverManager class and what are its methods.

[8) Connection interface](https://www.javatpoint.com/Connection-interface)

In this JDBC tutorial, we will learn what is Connection interface and what are its methods.

[9) Statement interface](https://www.javatpoint.com/Statement-interface)

In this JDBC tutorial, we will learn what is Statement interface and what are its methods.

[10) ResultSet interface](https://www.javatpoint.com/ResultSet-interface)

In this JDBC tutorial, we will learn what is ResultSet interface and what are its methods. Moreover, we will learn how we can make the ResultSet scrollable.

[11) PreparedStatement Interface](https://www.javatpoint.com/PreparedStatement-interface)

In this JDBC tutorial, we will learn what is benefit of PreparedStatement over Statement interface. We will see examples to insert, update or delete records using the PreparedStatement interface.

[12) ResultSetMetaData interface](https://www.javatpoint.com/ResultSetMetaData-interface)

In this JDBC tutorial, we will learn how we can get the metadata of a table.

[13) DatabaseMetaData interface](https://www.javatpoint.com/DatabaseMetaData-interface)

In this JDBC tutorial, we will learn how we can get the metadata of a database.

[14) Storing image in Oracle](https://www.javatpoint.com/storing-image-in-oracle-database)

Let's learn how to store image in the Oracle database using JDBC.

[15) Retrieving image from Oracle](https://www.javatpoint.com/retrieving-image-from-oracle-database)

Let's see the simple example to retrieve image from the Oracle database using JDBC.

[16) Storing file in Oracle](https://www.javatpoint.com/storing-file-in-oracle-database)

Let's see the simple example to store file in the Oracle database using JDBC.

[17) Retrieving file from Oracle](https://www.javatpoint.com/retrieving-file-from-oracle-database)

Let's see the simple example to retrieve file from the Oracle database using JDBC.

[18) CallableStatement](https://www.javatpoint.com/CallableStatement-interface)

Let's see the code to call stored procedures and functions using CallableStatement.

[19) Transaction Management using JDBC](https://www.javatpoint.com/transaction-management-in-jdbc)

Let's see the simple example to use transaction management using JDBC.

[20) Batch Statement using JDBC](https://www.javatpoint.com/batch-processing-in-jdbc)

Let's see the code to execute batch of queries.

[21) JDBC RowSet](https://www.javatpoint.com/jdbc-rowset)

Let's see the working of new JDBC RowSet interface.