Advanced Programming

Week 5

JDBC

- Introduction
- JDBC Driver
- JDBC Architecture
- JDBC Classes and Interfaces

Tilahun Melak(PhD)

October, 2025

Objectives

By the end of this lecture, students should be able to:

- Use the JDBC API to access and manipulate databases.
- Describe the different types of JDBC drivers and their purposes.
- Understand how to integrate and use JDBC technology in Java applications.
- Load and register JDBC drivers properly.

Introduction

- Programs communicate with relational databases through an interface — software that enables data exchange between applications and a database management system (DBMS).
- In Java, this interface is provided by the JDBC API (Java Database Connectivity).
- The JDBC API allows Java applications to:
  - Establish connections with databases.
  - Send SQL queries and updates.
  - Retrieve and manipulate data programmatically.

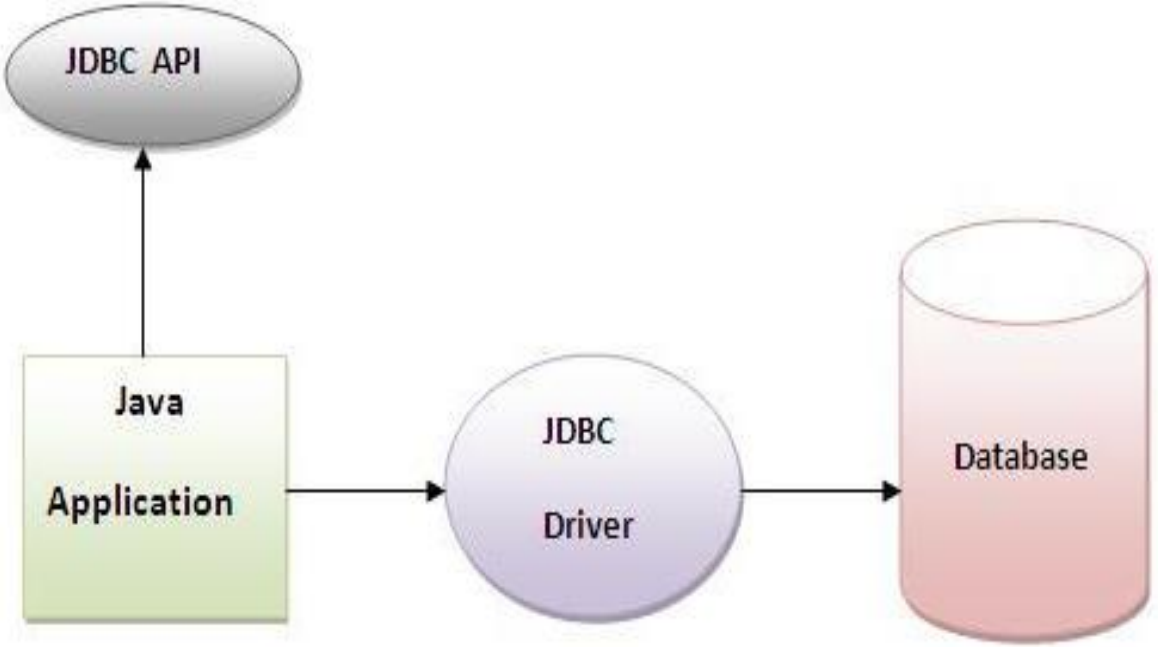
Oracle. (n.d.). JDBC overview. Oracle Help Center.

Introduction Cntd…

- Package java.sql contains classes and interfaces for accessing relational databases in Java.
- Using the JDBC API enables developers to change the underlying DBMS without modifying the Java code that accesses the database.
- The JDBC API can also be used to interact with multiple data sources in a distributed, heterogeneous environment.

Oracle. (n.d.). JDBC overview. Oracle Help Center.

Introduction Cntd…



Oracle. (n.d.). JDBC overview. Oracle Help Center.

JDBC Driver

- JDBC Driver is a software component that enables java application to interact with the database. There are 4 types of JDBC drivers:

    1. JDBC-ODBC bridge driver
    2. Native-API driver (partially java driver)
    3. Network Protocol driver (fully java driver)
    4. Thin driver (fully java driver)

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

1. JDBC-ODBC bridge driver

   - The JDBC-ODBC bridge driver uses ODBC driver to connect to the database.
   - The JDBC-ODBC bridge driver converts JDBC method calls into the ODBC function calls.
   - This is now discouraged because of thin driver.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

1. JDBC-ODBC bridge driver

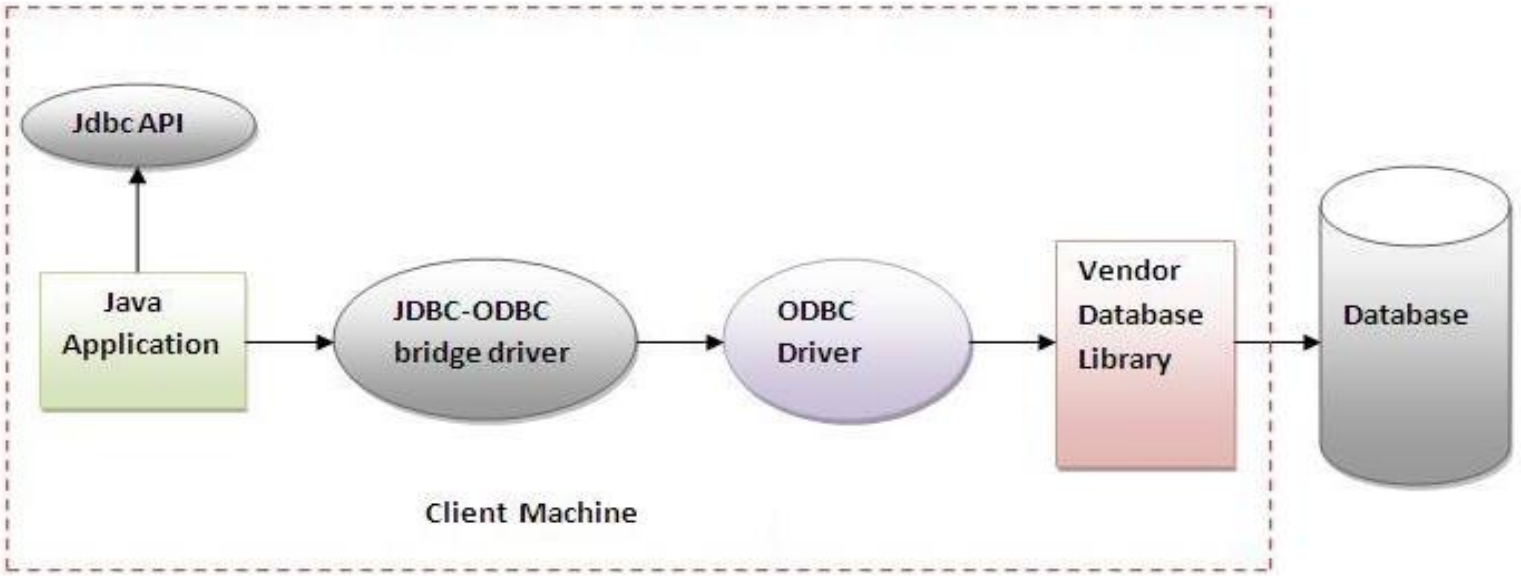


Figure- JDBC-ODBC Bridge Driver

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

1. JDBC-ODBC bridge driver

Advantages:

- easy to use.
- can be connected to any database.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

1. JDBC-ODBC bridge driver

Disadvantages:

- Performance degraded because JDBC method call is converted into the ODBC function calls.
- The ODBC driver needs to be installed on the client machine.
- This type of driver cannot talk to the database directly.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

2. Native-API driver (Type 2)

- The Native API driver uses the client-side libraries of the database.
- The driver converts JDBC method calls into native calls of the database API.
- It is not written entirely in java.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

2. Native-API driver (Type 2)

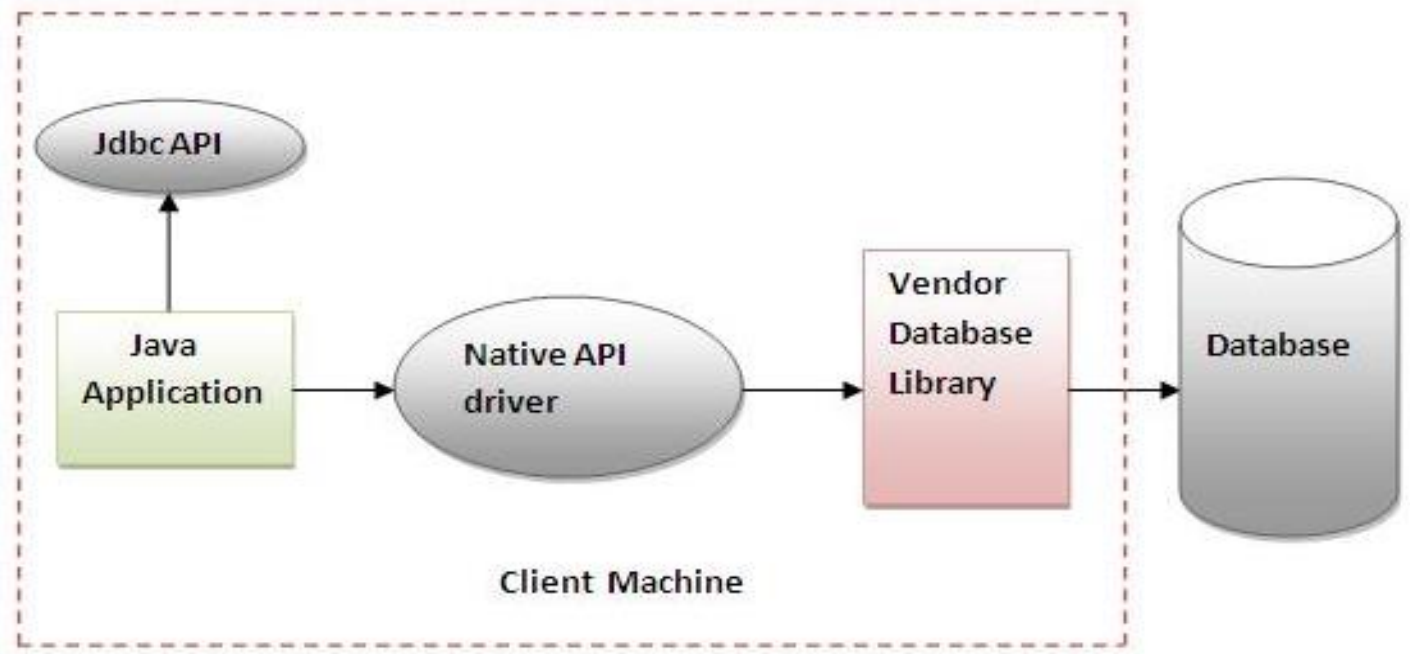


Figure- Native API Driver

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

2. Native-API driver (Type 2)

Advantages:

- performance upgraded than JDBC-ODBC bridge driver.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

2. Native-API driver (Type 2)

Disadvantage:

- The Native driver needs to be installed on the each client machine.
- The Vendor client library needs to be installed on client machine.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

3. Network Protocol driver (Type 3)

- The Network Protocol driver uses middleware (application server) that converts JDBC calls directly or indirectly into the vendor-specific database protocol.
- It is fully written in java.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

3. Network Protocol driver (Type 3)

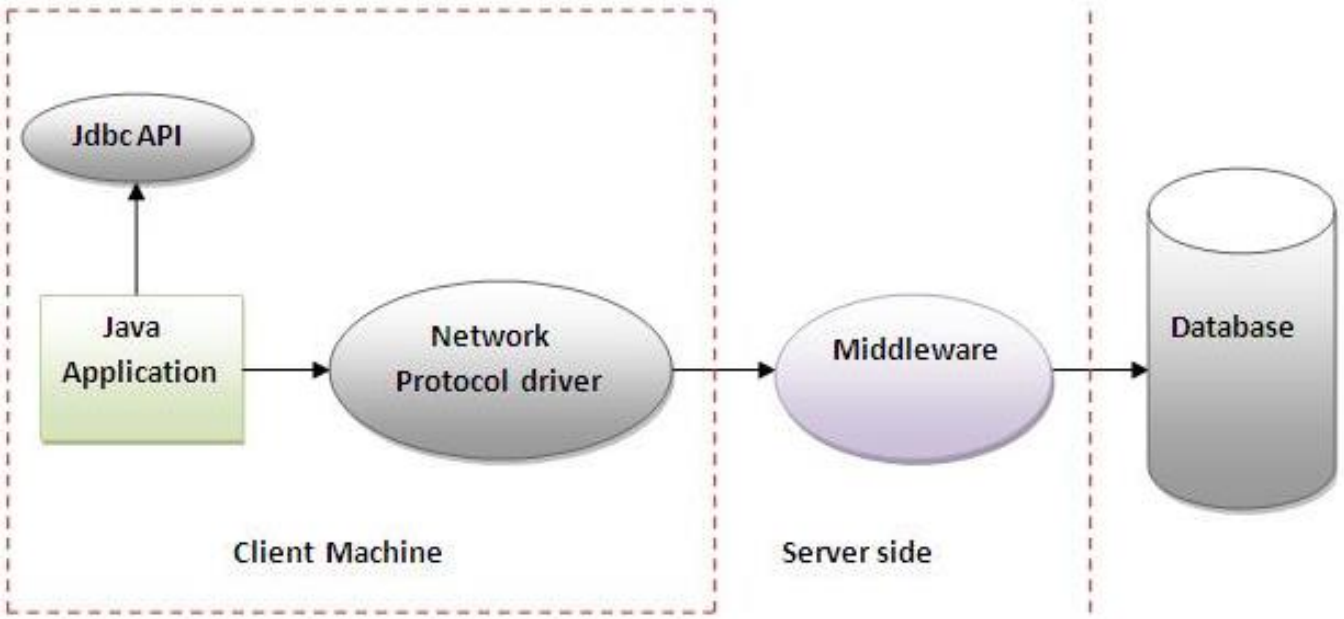


Figure- Network Protocol Driver

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

3. Network Protocol driver (Type 3)

Advantages:

- No client side library is required because of application server that can perform many tasks like auditing, load balancing, logging etc.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

3. Network Protocol driver (Type 3)

Disadvantage:

- Network support is required on client machine.
- Requires database-specific coding to be done in the middle tier.
- Maintenance of Network Protocol driver becomes costly because it requires database-specific coding to be done in the middle tier.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

4. Thin driver –Pure Java (Type 4)

- The thin driver converts JDBC calls directly into the vendor-specific database protocol.
- That is why it is known as thin driver.
- It is fully written in Java language.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

4. Thin driver –Pure Java (Type 4)

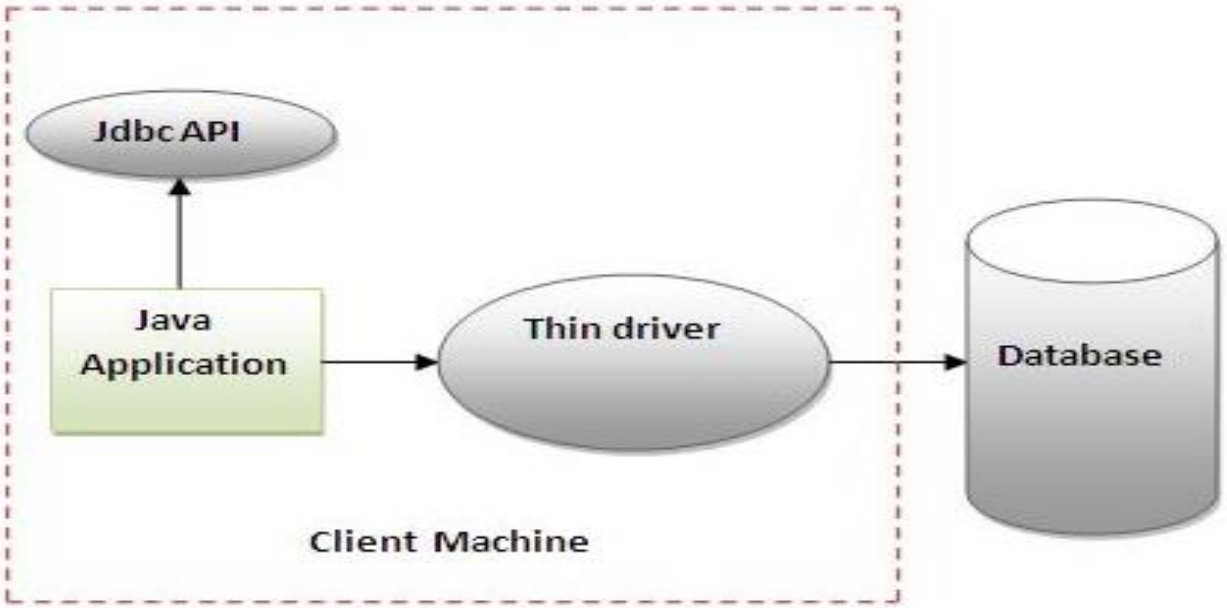


Figure- Thin Driver

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Driver Cntd…

4. Thin driver –Pure Java (Type 4)

Advantages:

- Better performance than all other drivers.
- Does not require any native library.

Oracle. (n.d.). JDBC Overview. Oracle Java Documentation.

JDBC Driver Cntd…

4. Thin driver –Pure Java (Type 4)

Disadvantage:

- Drivers depends on the Database.

Oracle. (n.d.). JDBC Overview. Oracle Java Documentation.

JDBC Architecture

- JDBC architecture can be classified into two types:
  - Two-Tier Architecture
  - Three-Tier Architecture
- This classification depends on how the application communicates with the database.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Architecture

- Driver Mapping:
  - Two-Tier Architecture:
    - Used by Type 2 and Type 4 drivers
  - Three-Tier Architecture:
    - Used by Type 1 and Type 3 drivers

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Architecture

- Figure (i): JDBC Two-Tier Architecture
  - The Java application communicates directly with the database using a JDBC driver.
  - No intermediate layer is involved.
- Figure (ii): JDBC Three-Tier Architecture
  - The application connects to a middleware or application server, which then interacts with the database.
  - Adds a layer between client and database for scalability and security.

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Architecture Cntd…

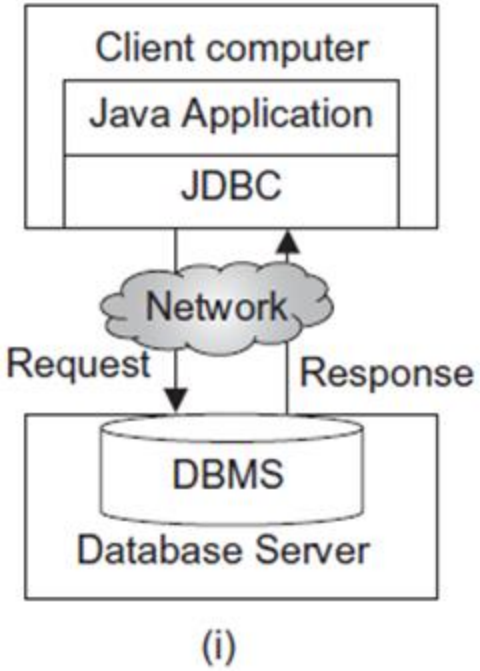


Fig. Two-tier JDBC Architecture

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Architecture Cntd…

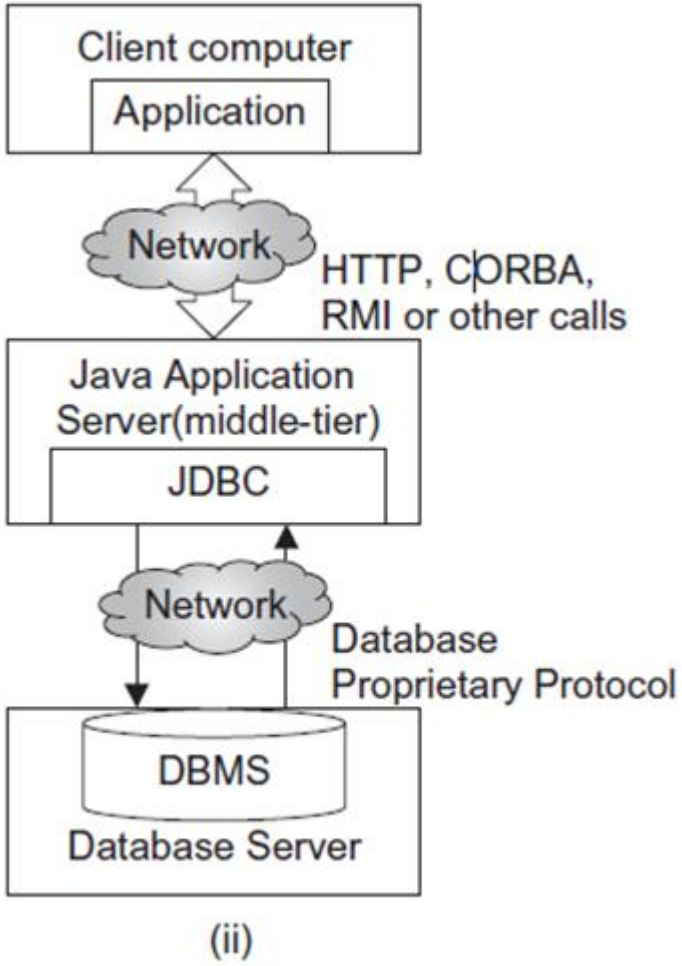


Fig. Three-tier JDBC Architecture

JDBC Classes and Interfaces

- Java provides an API for accessing and processing data stored in a data source (usually a relational database).

Class/Interface	Description
DriverManager	The basic service for managing a set of JDBC drivers
Connection	A connection (session) with a specific database
Statement	The object used for executing a static SQL statement and returning the results it produces

Oracle. (n.d.). *JDBC overview*. Oracle.

JDBC Classes and Interfaces Cntd…

ResultSet	A table of data representing a database result set, which is usually generated by executing a statement that queries the database
PreparedStatement	An object that represents a precompiled SQL statement
CallableStatement	The interface used to execute SQL stored procedures
DatabaseMetaData	Comprehensive information about the database as a whole
ResultSetMetaData	An object that can be used to get information about the types and properties of the columns in a ResultSet object.

Oracle. (n.d.). *JDBC overview*. Oracle.

Summary

- In today's lecture we have discussed about;
  - JDBC Driver
  - JDBC Architecture
  - JDBC Classes and Interfaces

References

- Oracle. (n.d.). JDBC overview. Oracle Help Center.
- Oracle. (n.d.). JDBC overview. Oracle.
- Oracle. (n.d.). *JDBC Overview*. Oracle Java Documentation.