Database Connectivity

By:
Gaurav Godhwani,
Sushil Sinhal

CSE Department , IES IPS Academy

Database Connectivity History

Before APIs like JDBC and ODBC, database connectivity was tedious:

- Database vendor provided function libraries for database access.
- Connectivity library was proprietary.
- Data access portions had to be rewritten with changes in the application.
- Application developers were stuck with a particular database product for a given application

Why Java ???

- Write once, run anywhere
  - Multiple client and server platforms
- Object-relational mapping
  - databases optimized for searching/indexing
  - objects optimized for engineering/flexibility
- Network independence
  - Works across Internet Protocol
- Database independence
  - Java can access any database vendor
- Ease of administration

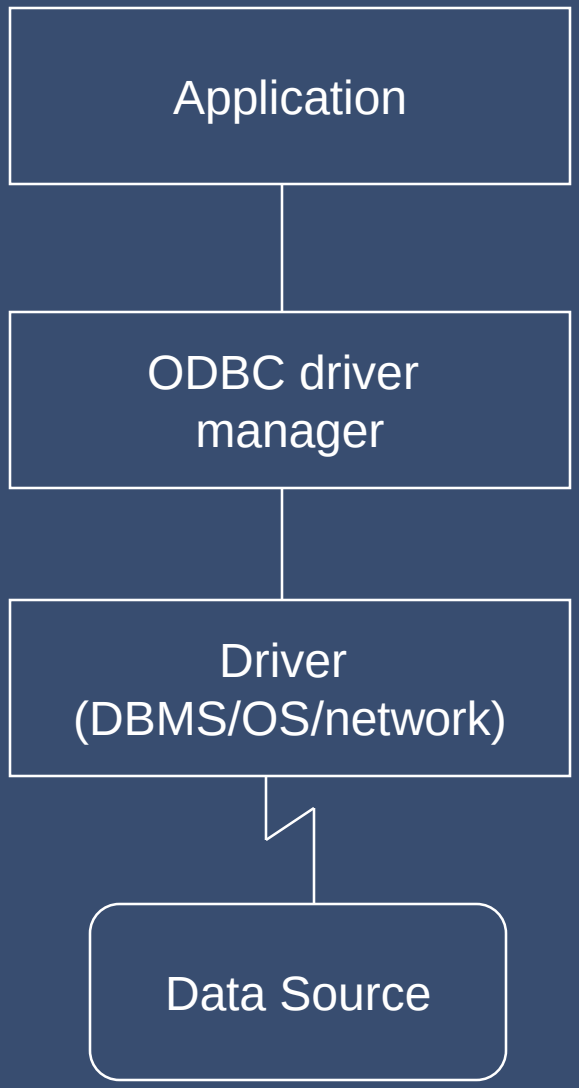
ODBC

- A standard or open application programming interface (API) for accessing a database
- Developed by SQL Access Group, chiefly Microsoft, in 1992
- Access to various kinds of Databases
- Allows programs to use SQL requests that will access databases without knowledge of the proprietary interfaces to the databases
- Handles these requests and converts it into a request understandable by an individual database system.

ODBC Requirements

- An ODBC software for a particular OS.
- A separate module or *driver* for each database to be accessed
- Driver masks the heterogeneity of DBMS, operating system and network protocol.
- Ex: Sybase Driver, Windows Driver, etc.

ODBC Architecture



JDBC

- Java API for connecting programs written in Java to the data in relational databases

- The standard defined by Sun Microsystems, allowing individual providers to implement and extend the standard with their own JDBC drivers.

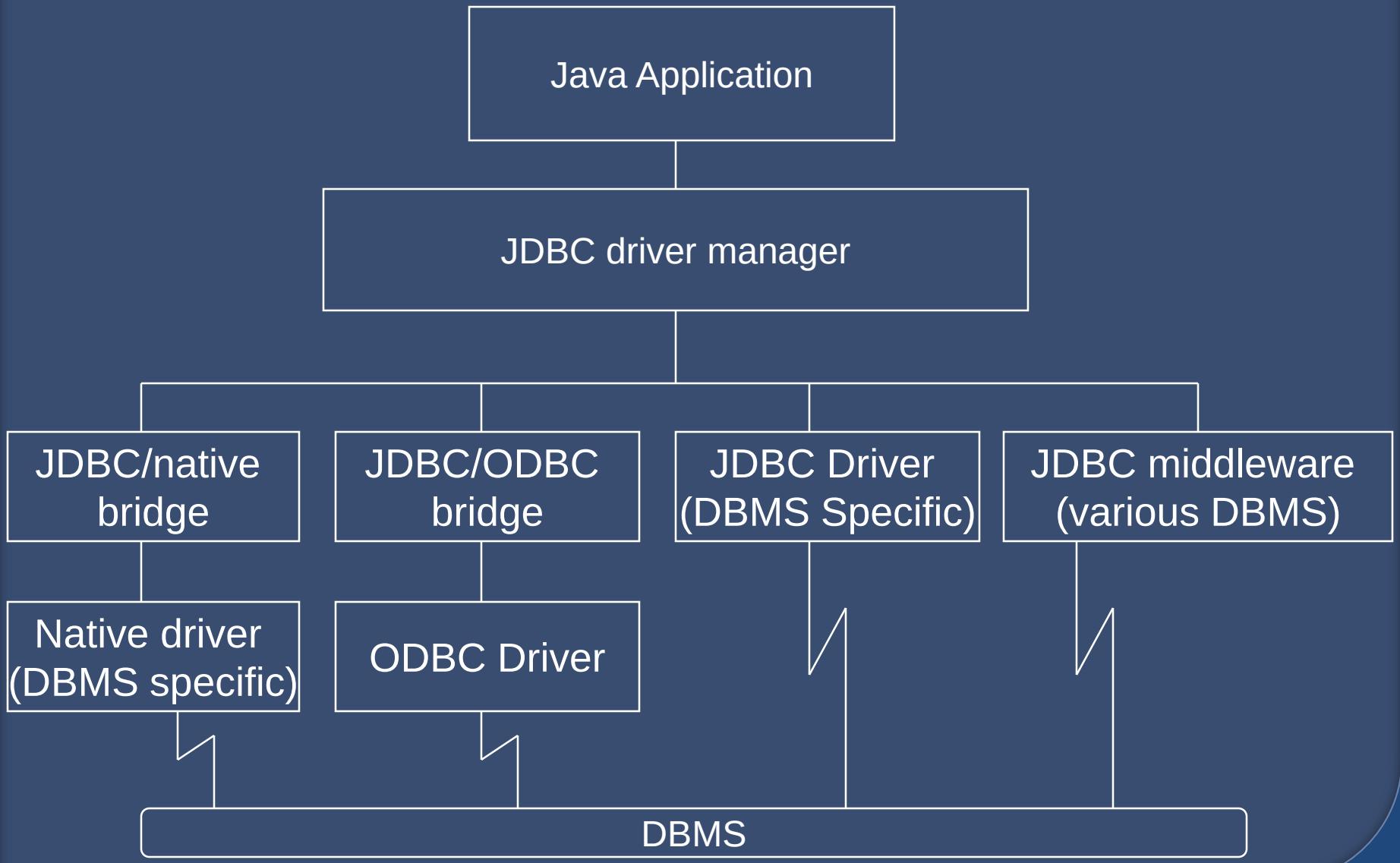
- Tasks of JDBC:

    1) establishes a connection with a database
    2) sends SQL statements
    3) processes the results

JDBC API

- The JDBC API supports both two-tier and three-tier models for database access.
- Two-tier model -- a Java applet or application interacts directly with the database.
- Three-tier model -- introduces a middle-level server for execution of business logic:
  - the middle tier to maintain control over data access.
  - the user can employ an easy-to-use higher-level API which is translated by the middle tier into the appropriate low-level calls.

JDBC Architecture



JDBC Driver Types

- Class-I:
  JDBC:ODBC (mainly for Desktop Applications)
  - > Use bridging technology
  - > Requires installation/configuration on client machines
  - > Not good for Web

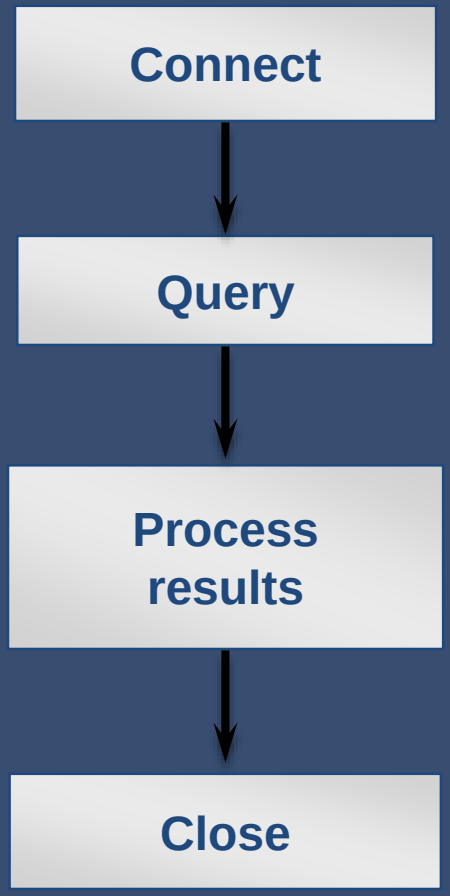
- Class-II:
  Native API Drivers (Vendor Specific drivers)
  - > Requires installation/configuration on client machines
  - > Used to leverage existing CLI libraries
  - > Usually not thread-safe
  - > Mostly obsolete now
  - > e.g. Intersolv Oracle Driver, WebLogic drivers

JDBC Driver Types

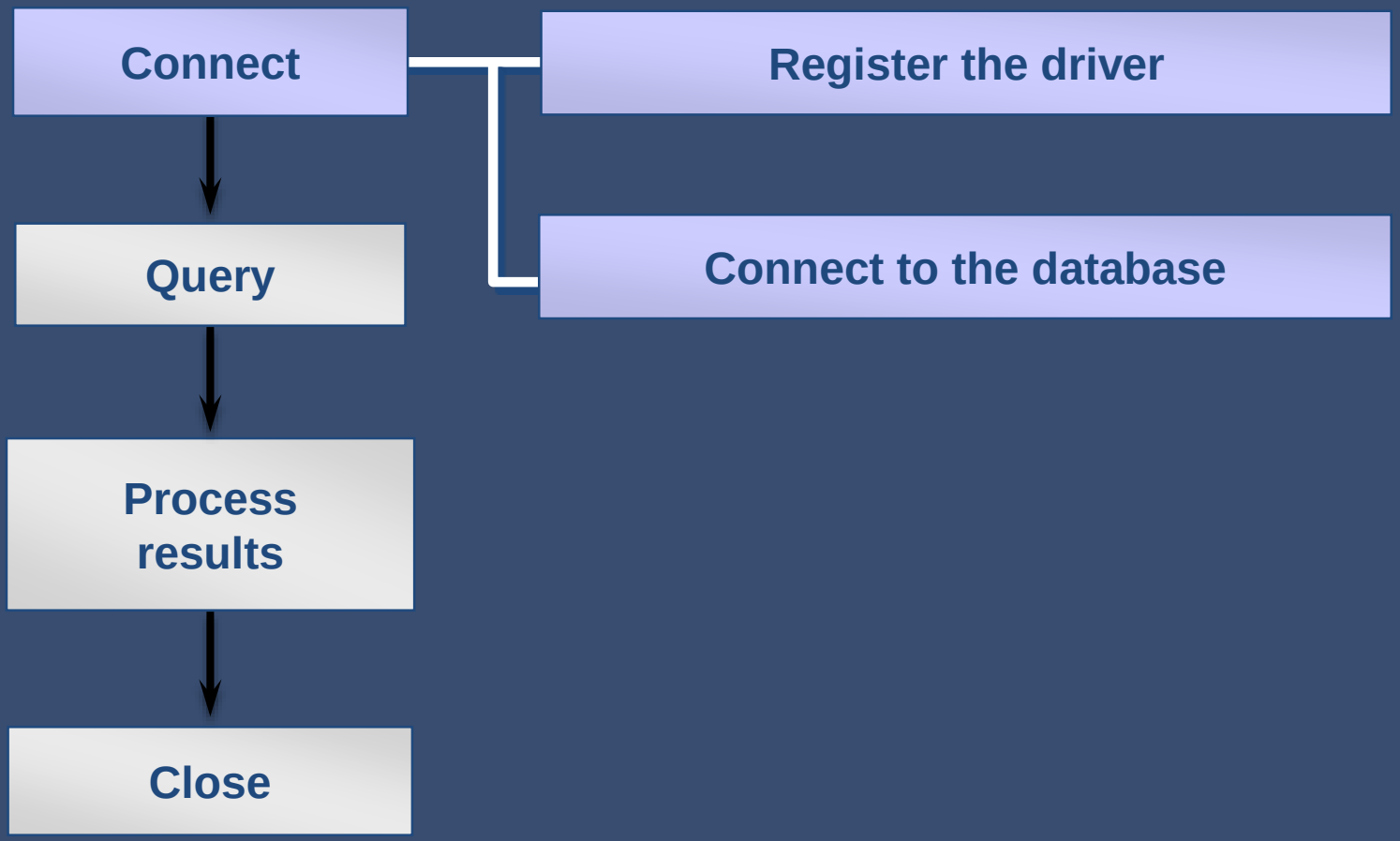
- Class-III:
  Network API
  - > Calls middleware server, usually on database host
  - > Very flexible & allows access to multiple databases using one driver
  - > Only need to download one driver
  - > But it's another server application to install and maintain
  - > e.g. Symantec DBAnywhere

- Class-IV:
  Network Protocol Driver (used for Network based Applications)
  - > Pure Java Drivers
  - > Use Java networking libraries to talk directly to database engines
  - > need to download a new driver for each database engine
  - > e.g. Oracle, MySQL

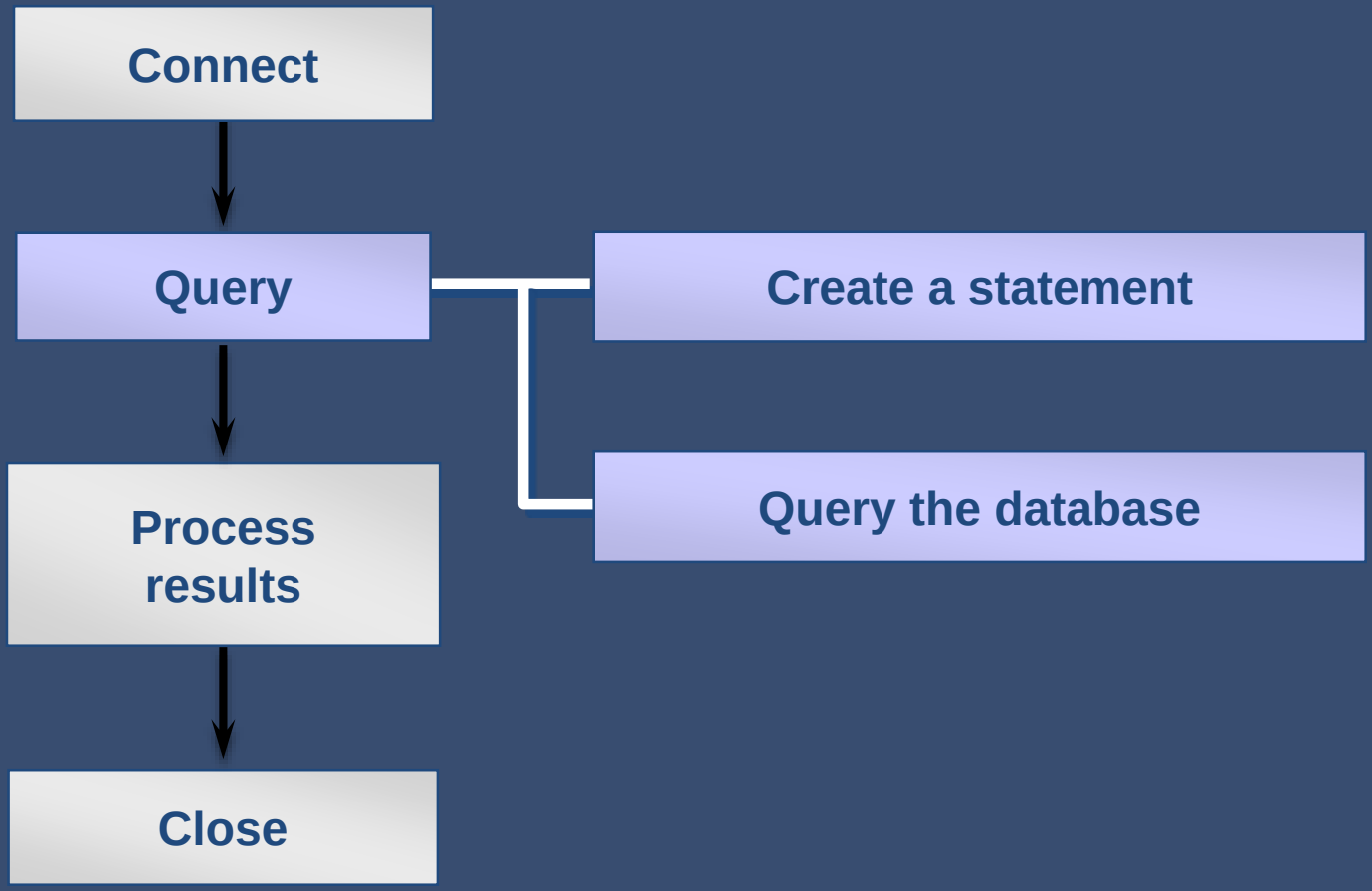
Overview of Querying a Database With JDBC



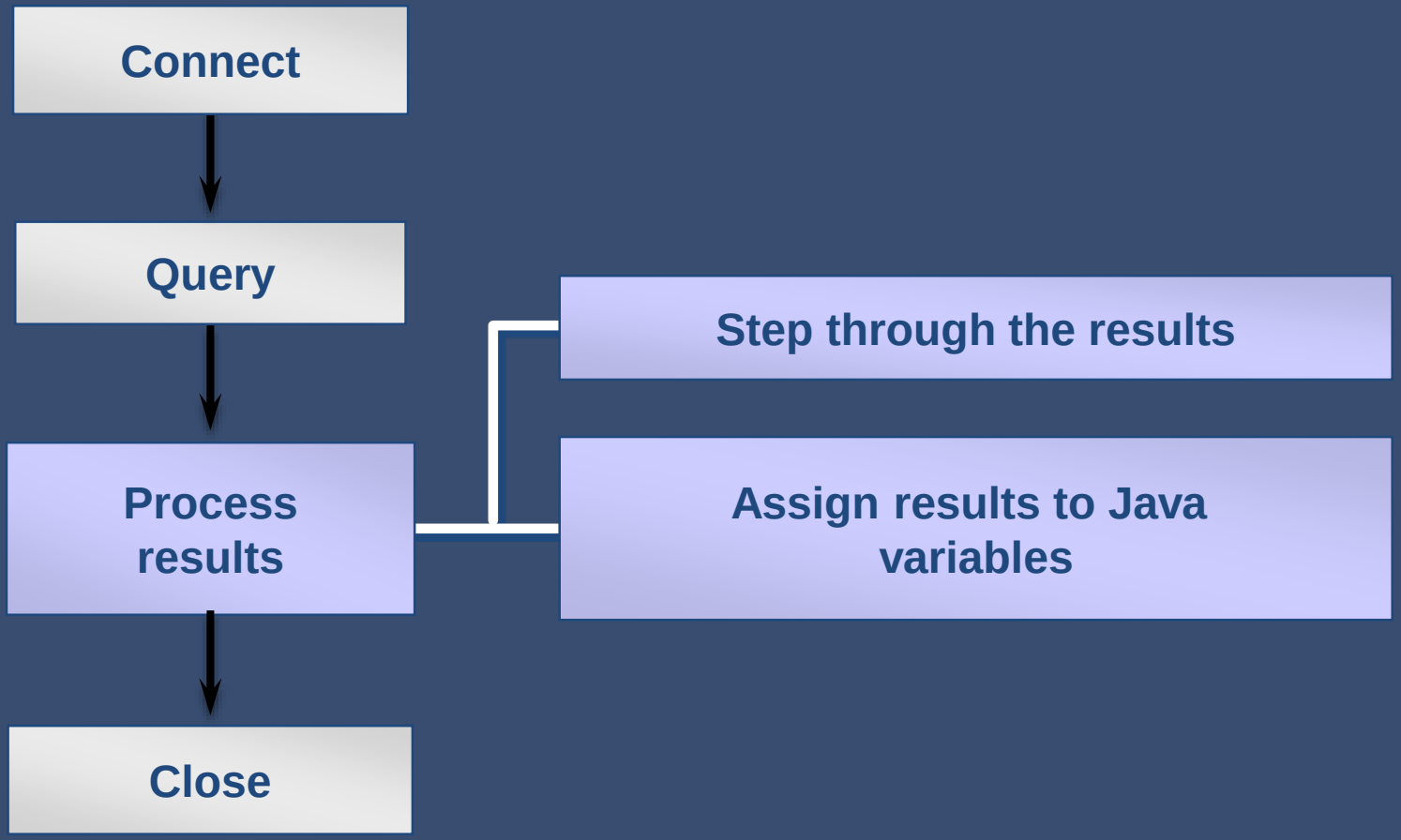
Stage 1: Connect



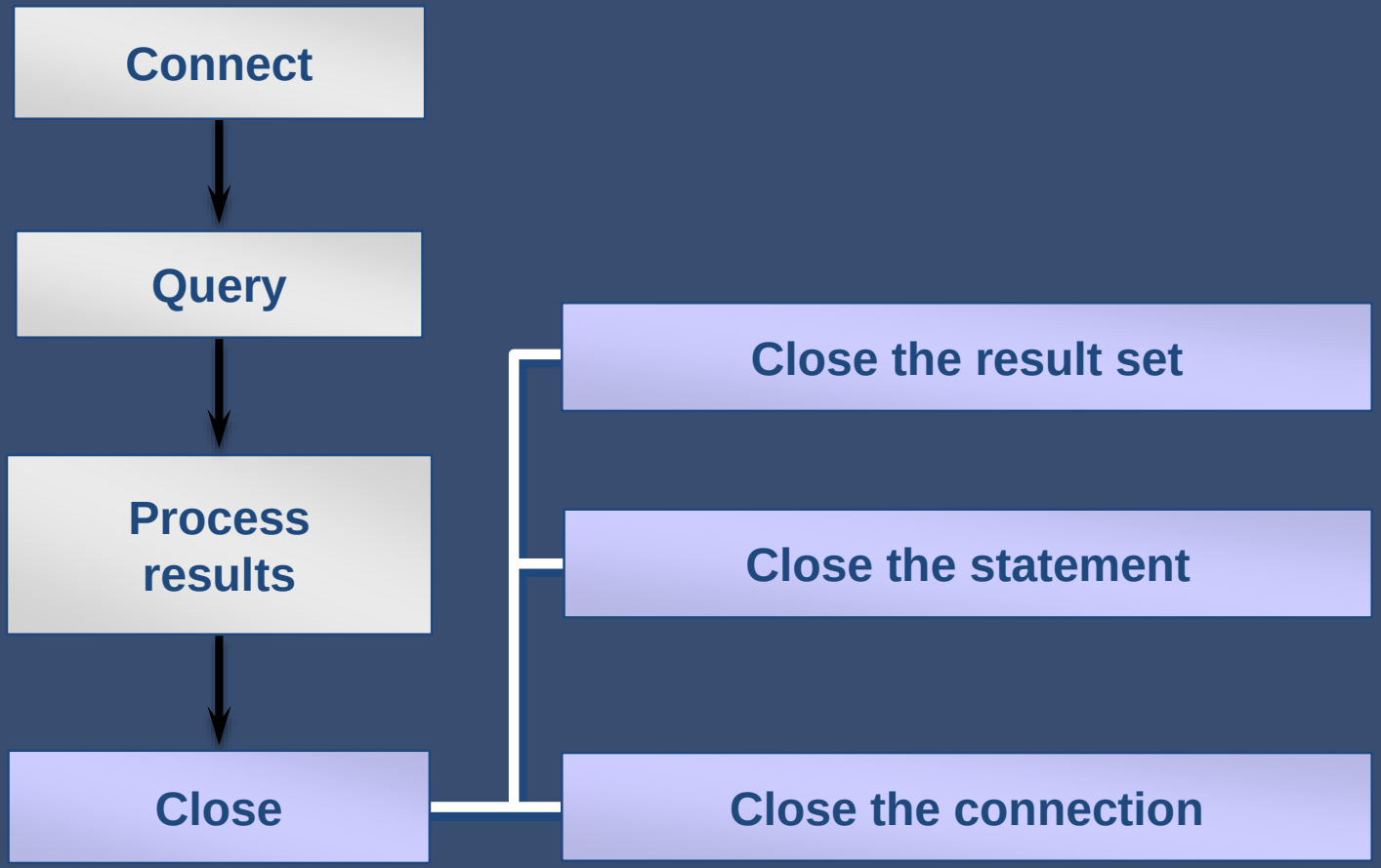
Stage 2: Query



Stage 3: Process Results



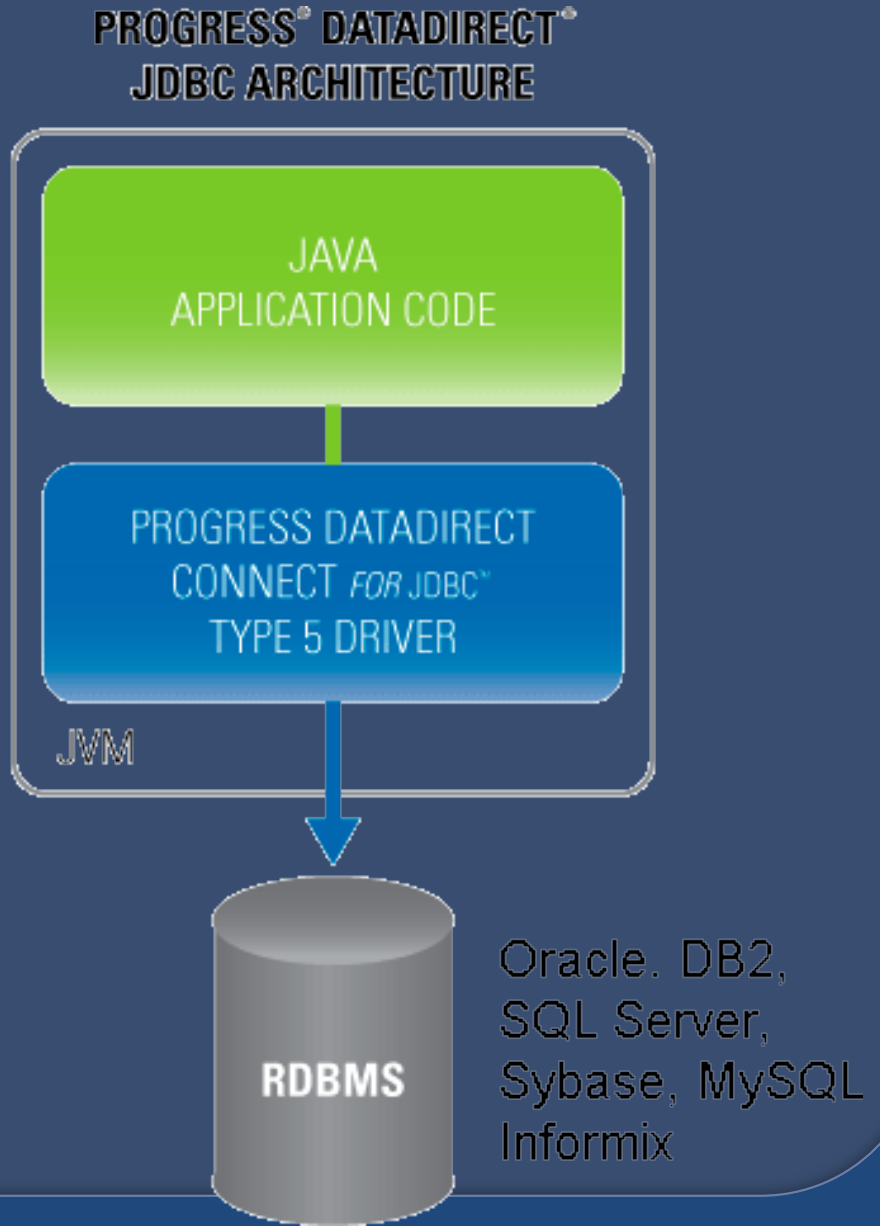
Stage 4: Close

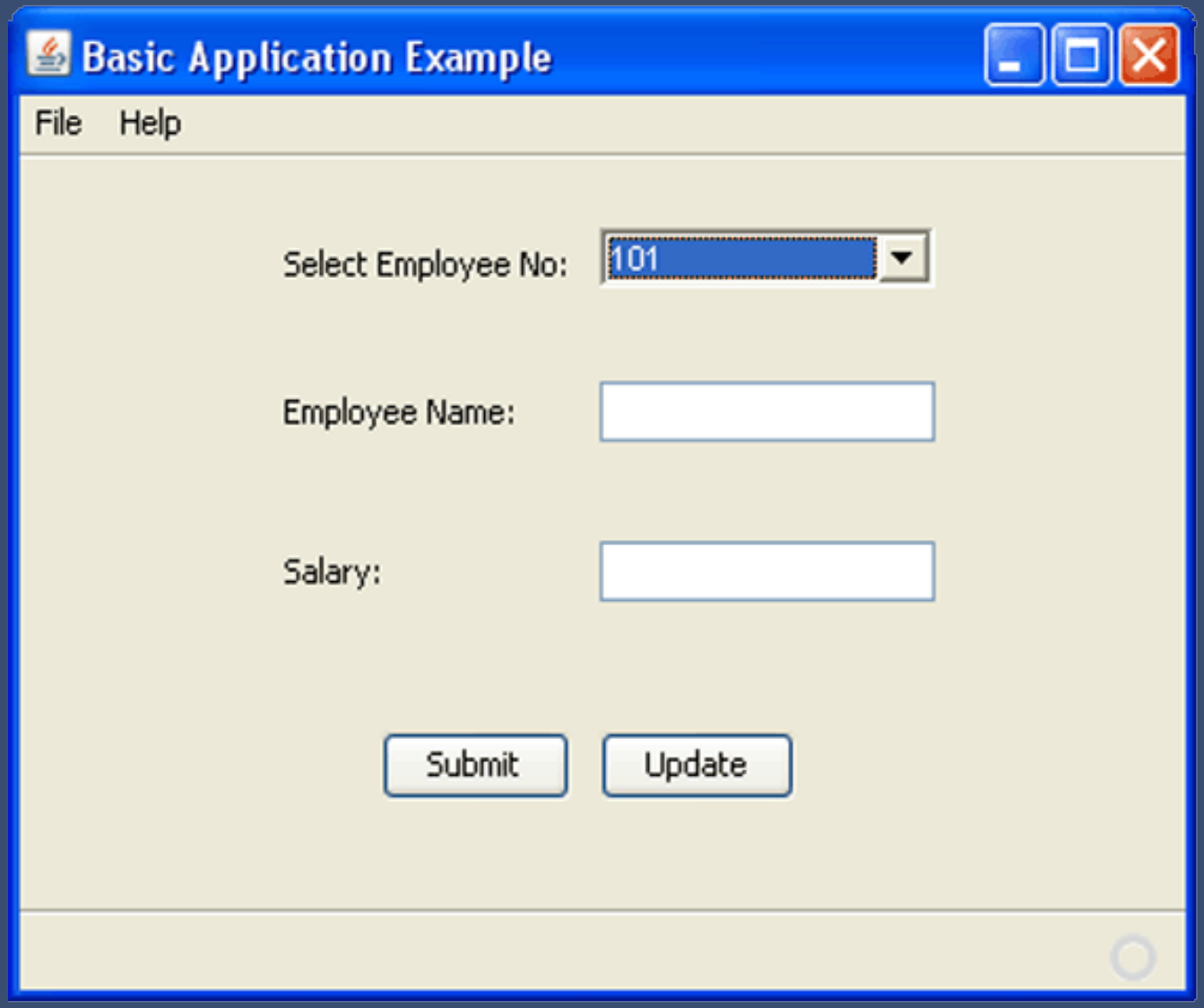


Recent Advancement

ADVANTAGES

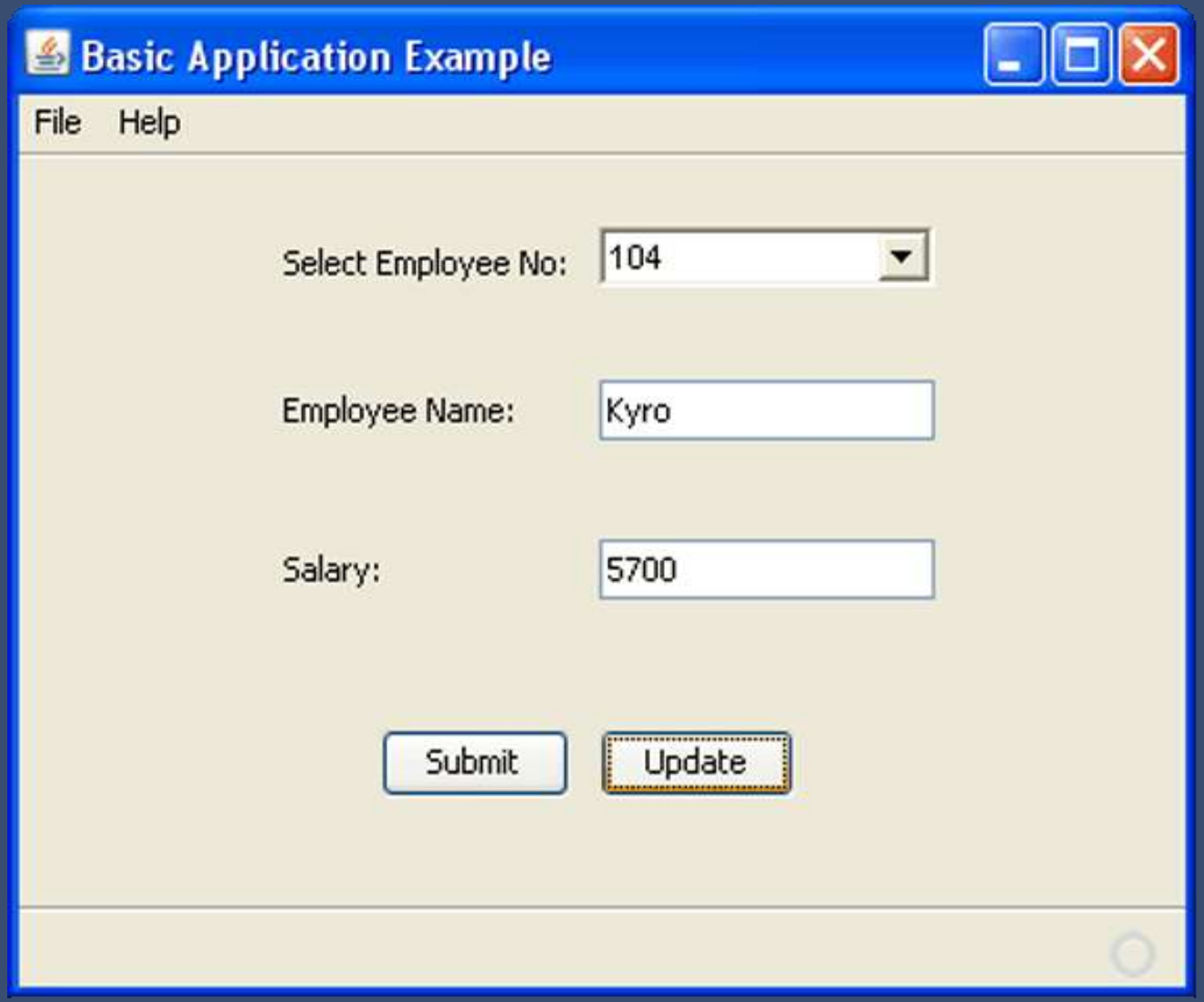
- Common Type 5 architecture and features across all major databases
- Code and feature interoperability support across databases
- One driver for each database regardless of database version or JDK version
- Robust JDBC 3.0/4.0 specification feature set improves developer productivity
- Performance and scalability leader in SPECjAppServer/ECPerf benchmarks
- Extensive test suites and customer deployments ensure unrivaled reliability

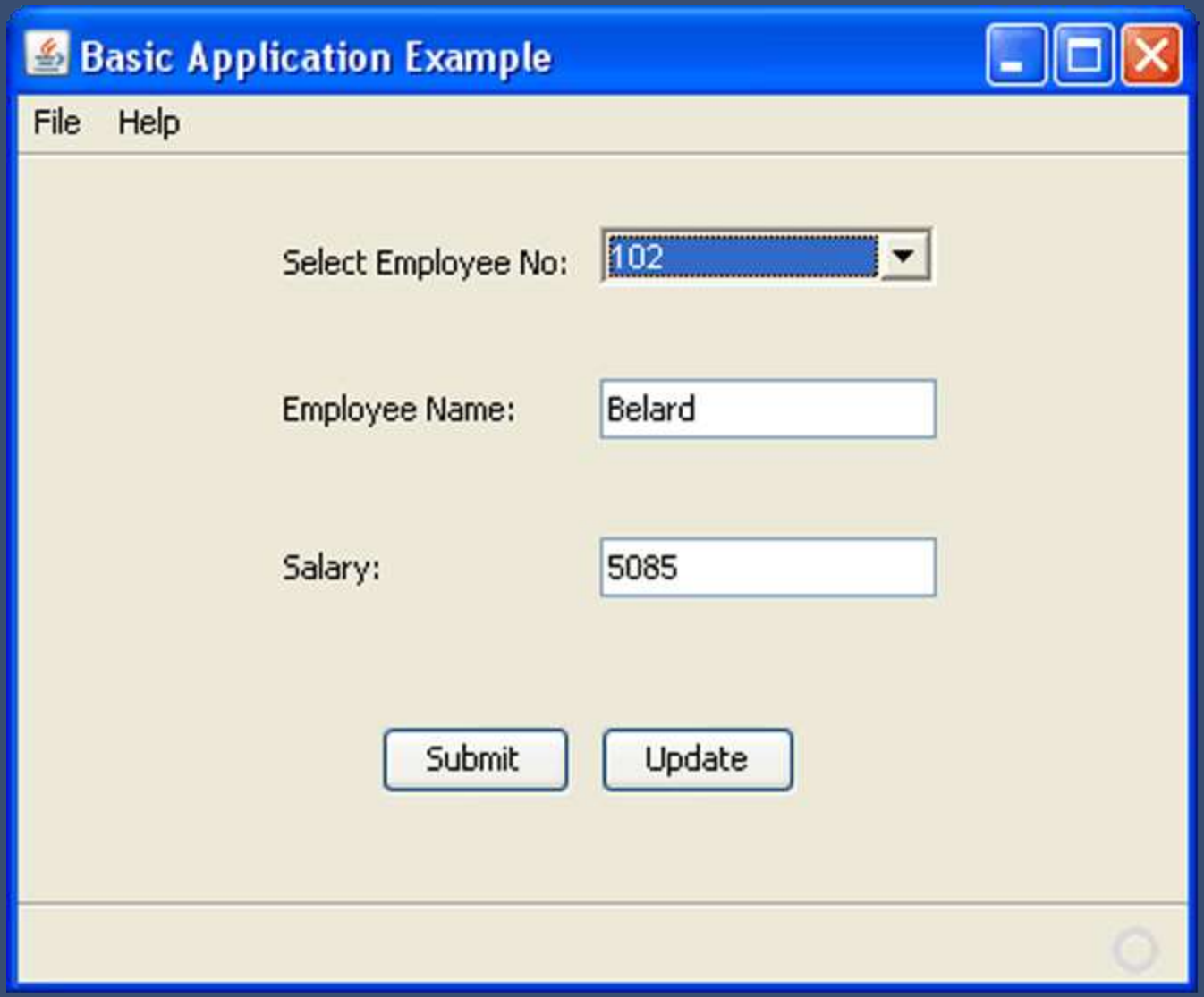


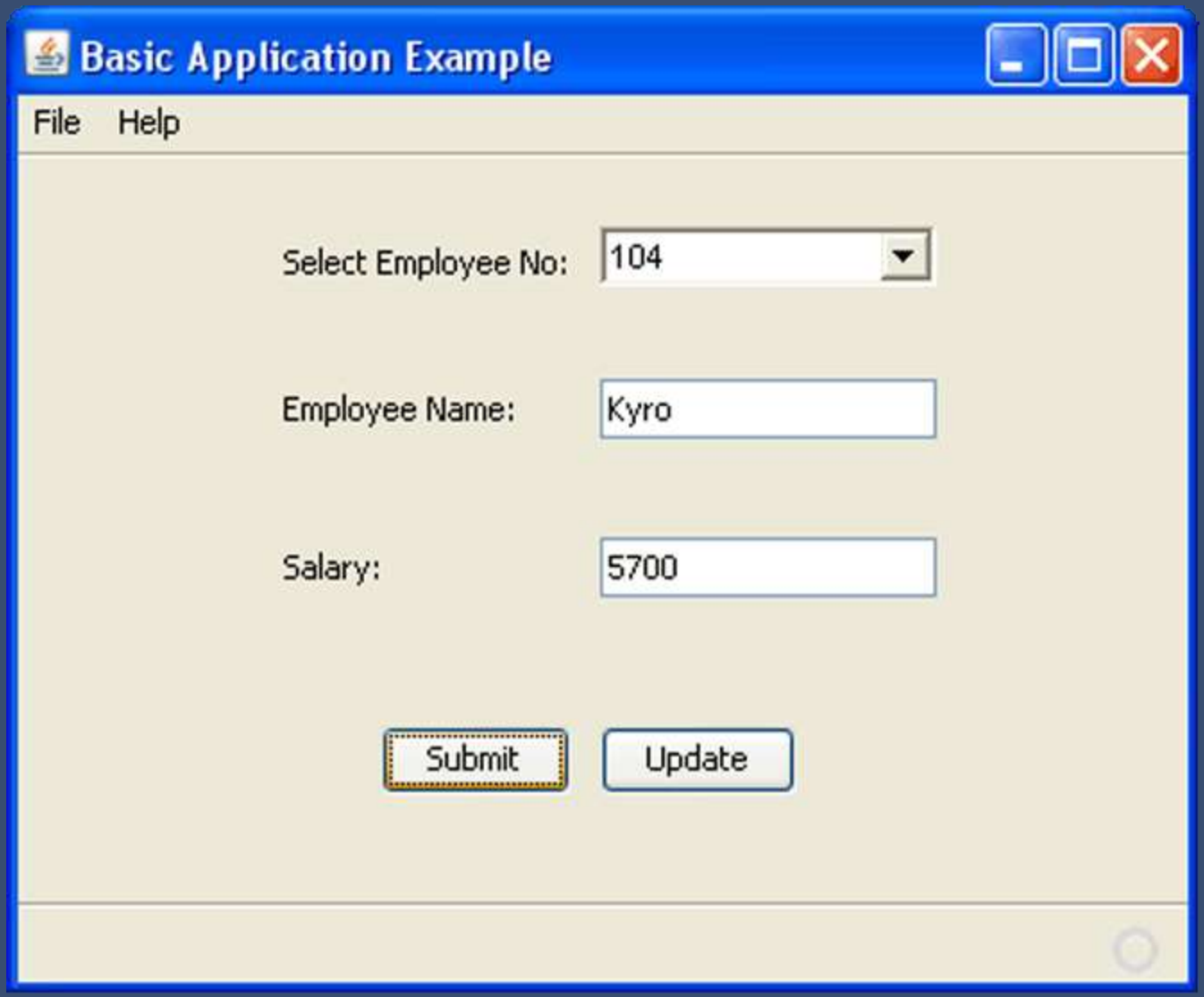

Basic Application Example
File
Help
Select Employee No:
101
Employee Name:
Salary:
Submit
Update


Basic Application Example
File
Help
Select Employee No:
101
101
102
103
104
Employee Name:
Salary:
Submit
Update


Basic Application Example
File
Help
Select Employee No:
104
Employee Name:
Kyro
Salary:
4600
Submit
Update


Basic Application Example
File
Help
Select Employee No:
104
Employee Name:
Kyro
Salary:
5700
Submit
Update


Basic Application Example
File
Help
Select Employee No:
102
Employee Name:
Belard
Salary:
5085
Submit
Update


Basic Application Example
File
Help
Select Employee No:
104
Employee Name:
Kyro
Salary:
5700
Submit
Update

Thank You !!!