Banking APP With JDBC

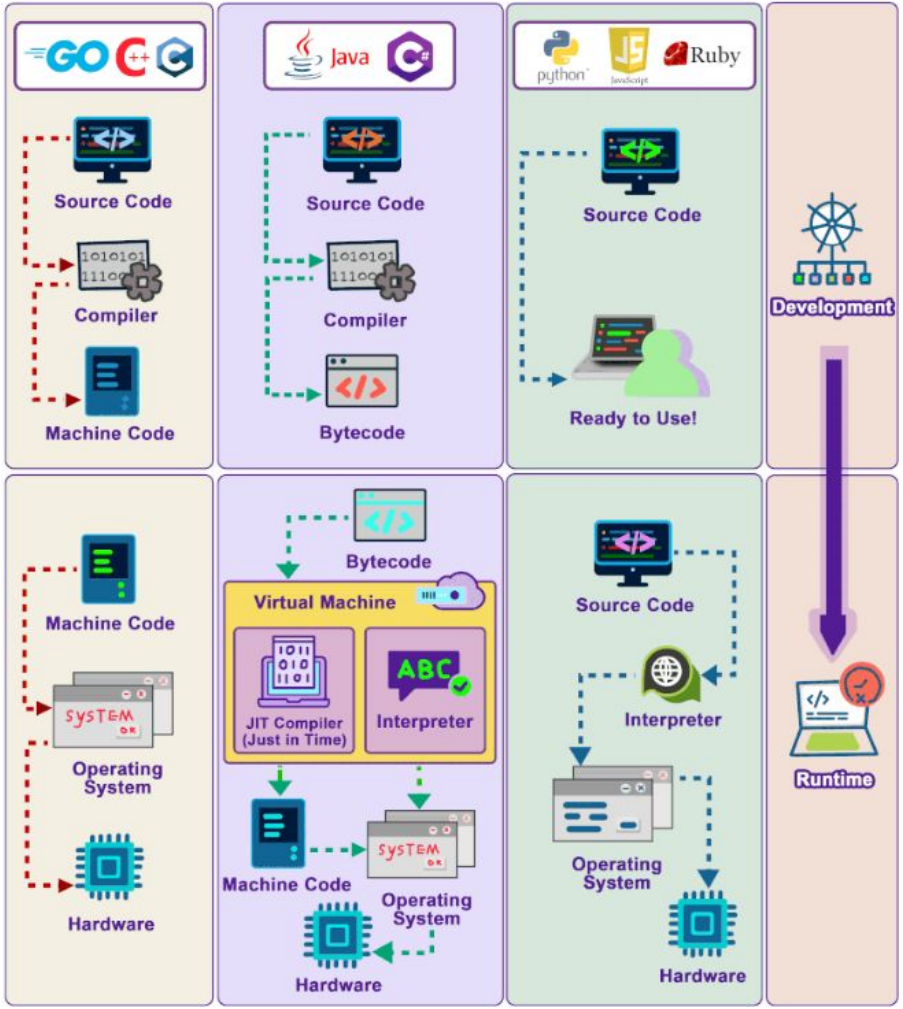
By Apurv Henkare

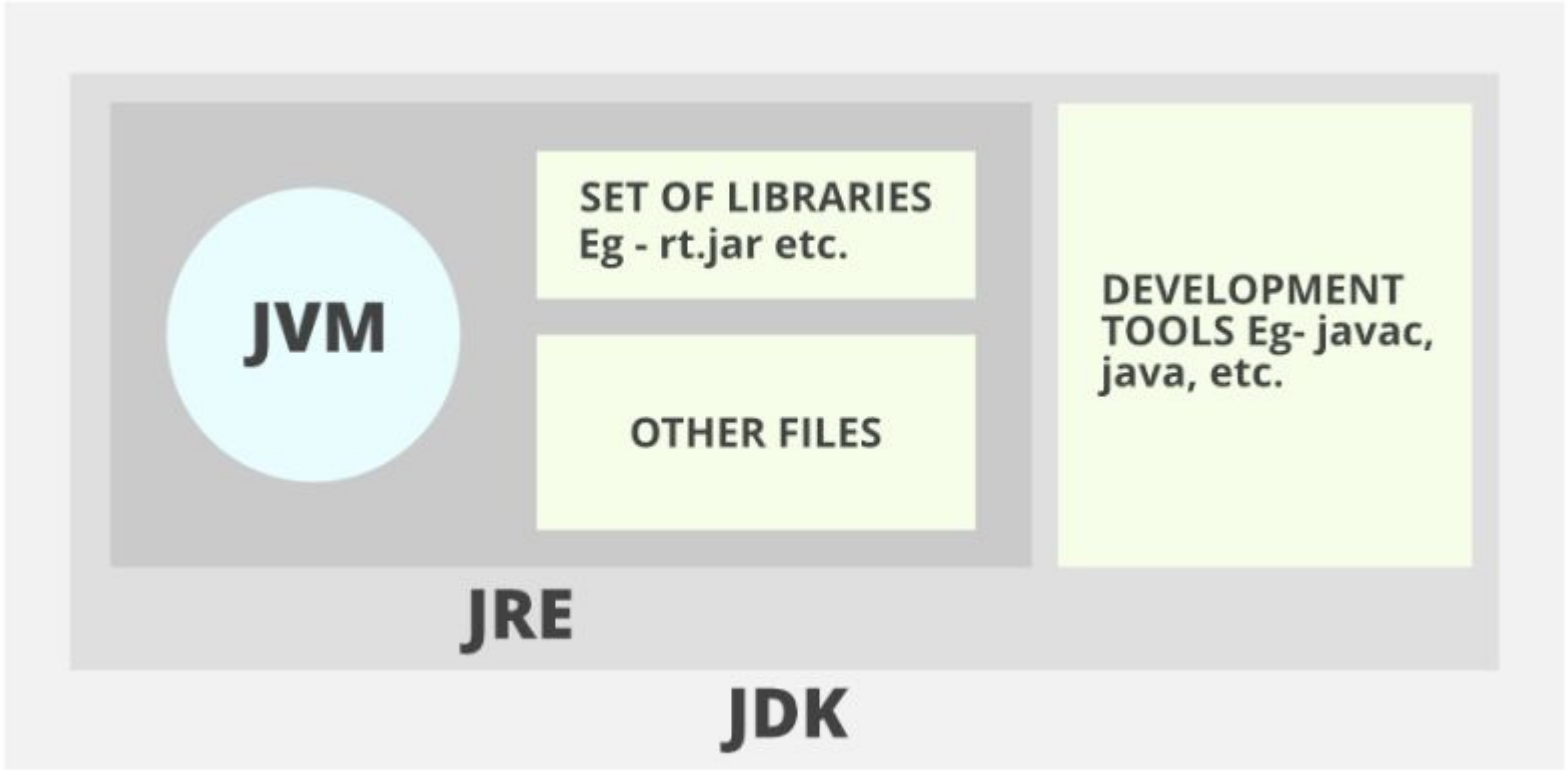
Index

What Is Java

- Java is a class-based, object-oriented, high-level programming language
- It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA),

- Java was originally developed by James Gosling at Sun Microsystems.


GO
C++
C
Source Code
Compiler
Machine Code
Java
C#
Source Code
Compiler
Bytecode
python
JavaScript
Ruby
Source Code
Ready to Use!
Development
Machine Code
SYSTEM
Operating System
Hardware
Bytecode
Virtual Machine
JIT Compiler (Just in Time)
ABC
Interpreter
Machine Code
SYSTEM
Operating System
Hardware
Source Code
Interpreter
Operating System
Hardware
Runtime


JVM
SET OF LIBRARIES
Eg - rt.jar etc.
OTHER FILES
DEVELOPMENT TOOLS Eg- javac, java, etc.
JRE
JDK

What is JDBC

- Java Database Connectivity (JDBC) is an application programming interface (API) for the Java programming language which defines how a client may access a database.

- The JDBC API consists of a set of interfaces and classes written in the Java programming language.


Data
Database
SQL


Java
Database
JDBC

SCOPE OF PROJECT

- The Bank App manages money transactions in a bank.
- It handles deposits and withdrawals, checks if transactions follow rules, and categorizes them as "Valid" or "InValid."
- It stores data in a MySQL database, separating valid and invalid transactions for auditing.
- The user interface provides a quick summary for easy monitoring.

Dependencies

- The primary dependency for this project is the MySQL JDBC driver. "com.mysql.cj.jdbc.Driver" is the class name of the MySQL JDBC driver.

- Mysql-connector-j-8.2.0.jar contains the necessary classes and code for the MySQL JDBC driver, which is crucial for establishing a connection between your Java application and a MySQL database.

Components

Test Class

It's responsible for loading the database driver, establishing a connection, and invoking methods from the JDBC class to perform various database operations.

JDBC Class

It has methods for displaying bank statements, updating bank statements, and inserting statements into either a "ValidTrans" or "InValidTrans" table

Utility Method

1. `initializeDisplayStatement()`
   - Initializes a `Statement` and `ResultSet` for displaying bank statements by executing a SELECT query on the "BankTrans" table.
2. `displayBankStatement()`
   - Displays transaction details (transaction ID, account number, old balance, transaction type, transaction amount) obtained from the "BankTrans" table.
3. `updateBankStatement()`
   - Updates the "BankTrans" table by calculating new balances and transaction statuses based on transaction type and amount.
4. `getNewBal(double oldBal, double transAmt, String transType)`
   - Calculates and returns the new balance based on the old balance, transaction amount, and transaction type (withdrawal or deposit).
5. `getStatus(double tempNewBal)`
   - Determines and returns the transaction status ("Valid" or "Invalid") based on the calculated new balance.
6. `insertStatement()`
   - Inserts statements into "ValidTrans" or "InValidTrans" tables based on transaction status.
7. `insertValidTrans(String transID, String transType, double transAmt, String transStat)`
   - Inserts valid transactions into the "ValidTrans" table.
8. `insertInValidTrans(String transID, String transType, double transAmt, String transStat)`
   - Inserts invalid transactions into the "InValidTrans" table.

Method Chaining

- Method Chaining is the practice of calling different methods in a single line.
- Instead of calling other methods with the same object reference separately.
- write the object reference once and then call the methods by separating them with a (dot.).
  obj.method1().method2().method3();

  EG : new A().setint(10).setfloat(20).display();

REQUIREMENT ID	REQUIREMENT CATEGORY	REQUIREMENT TYPE	PRIORITY	HIERARCHY	REF
R001	FUNCTIONAL	STATED	HIGH		

REQUIREMENT DESCRIPTION	ESTABLISHING DATABASE CONNECTION
SCOPE	This functionality involves establishing a connection to the MySQL database to facilitate interactions with transaction records.
REQUIREMENT METHODOLOGICAL DETAILS	➢ **Implementation Details:** Uses JDBC to load the MySQL driver and establish a connection. Ensures the connection is created only if it does not already exist. ➢ **Considerations:** Handles exceptions related to database connection, printing stack traces if necessary.

REQUIREMENT ID	REQUIREMENT CATEGORY	REQUIREMENT TYPE	PRIORITY	HIERARCHY	REF
R001	FUNCTIONAL	STATED	HIGH		

REQUIREMENT DESCRIPTION	**Display Transaction Record of BankTrans**
SCOPE	The Functionality involves displaying record of BankTrans from the transactions.
REQUIREMENT METHODOLOGICAL DETAILS	**this.initializeDisplayStatement();** **Iterated the ResultSet**

REQUIREMENT ID	REQUIREMENT CATEGORY	REQUIREMENT TYPE	PRIORITY	HIERARCHY	REF
R001	FUNCTIONAL	STATED	HIGH		

REQUIREMENT DESCRIPTION	**Updating The Transaction Record of BankTrans**
SCOPE	The Functionality involves updating details such as new Balance and transaction status in the database .
REQUIREMENT METHODOLOGICAL DETAILS	**this.initializeDisplayStatement();** **Iterate the ResultSet.** **getNewBal(double oldBal, double transAmt, String transType)** **getStatus(double tempNewBal)**

REQUIREMENT ID	REQUIREMENT CATEGORY	REQUIREMENT TYPE	PRIORITY	HIERARCHY	REF
R001	FUNCTIONAL	STATED	HIGH		

REQUIREMENT DESCRIPTION	**Inserting The Transaction Records Of Valid InValid Table**
SCOPE	The Functionality involves inserting details into appropriate table.
REQUIREMENT METHODOLOGICAL DETAILS	**this.initializeDisplayStatement();** **Iterate the ResultSet.** **this.insertValidTrans(transID, transType, transAmt, transStat);** **this.insertInValidTrans(transID, transType, transAmt, transStat);**

OPERATING ENVIRONMENT

Database and Server:

- MySQL Server: Ensure proper configuration.
- Version: Compatible with MySQL, specify the version.

Java Environment:

- JRE Version: Specify the required Java Runtime Environment version.
- Dependencies: List necessary libraries or dependencies.

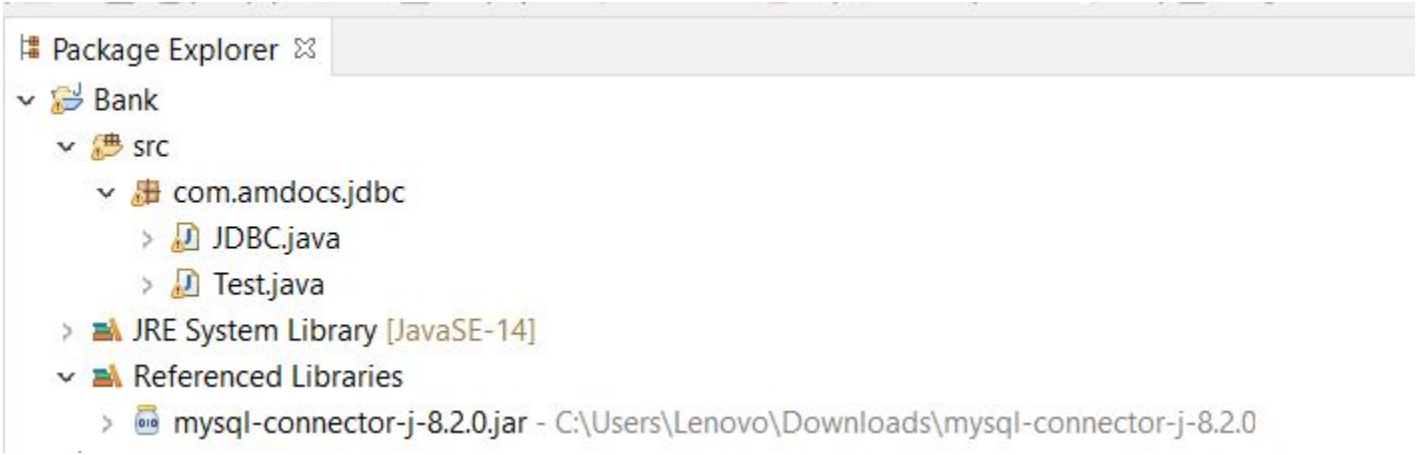
Development Environment:

- Integrated Development Environment (IDE): Mention the IDE used for development (e.g., IntelliJ IDEA, Eclipse).

Network and Configuration:

- Connection: Stable network connection to the MySQL database server.
- Ports: Specify required open network ports.
- Security: Briefly mention security measures and configuration files.

Hierarchy Of Project Artifacts



Thank You