

- 1.) Introduction to Advance Java.
- 2.) JDBC in simple way.
- 3.) Storage Areas
- 4.) Introduction to JDBC
- 5.) JDBC Architecture.
- 6.) JDBC API
- 7.) Driver Interface Vs. Driver class Vs. Driver Software
- 8.) Types of Drivers
- 9.) Standard steps to develop JDBC Application.
- 10.) Working with Type-1 Driver.
- 11.) Working with Type-2 Driver
- 12.) Working with Type-3 Driver
- 13.) Working with Type-4 Driver
- 14.) Working with Type-5 Driver.
- 15.) Summary of all 5 JDBC Drivers.

- 16.) Program to create & Drop tables.
- 17.) How to read dynamic input from the keyboard.
- 18.) How to form SQL Queries with dynamic input.
- 19.) Program for Basic ~~CR~~ CRUD Operations.
- 20.) Realtime coding standard for JDBC Application.
- 21.) How to develop & run JDBC Application with IDEs like Eclipse.
- 22.) Working with MySQL Database.
- 23.) Life cycle of SQL Query execution.
- 24.) Prepared Statement.
- 25.) Difference between Statement & PreparedStatement.
- 26.) SQL Injection Attack.
- 27.) Stored Procedures.
- 28.) Callable Statement.
- 29.) Cursors.

30.) Functions

31.) Statement Vs. Prepared Statement Vs. Callable Statement

32.) Batch Updates

33.) executeQuery() vs. executeUpdate() vs. execute()
vs. executeBatch()

34.) Insert & Retrieve Date values

35.) BLOB & CLOB Introduction

36.) Insert & Retrieve BLOB type

37.) Insert & Retrieve CLOB type

38.) Connection Pooling

39.) Working with properties.

40.) Different ways to get Connection Object.

~~41.) Different ways to get Connection Object.~~

41.) Transaction Management.

42.) Savepoint

43.) Transaction Concurrency Problems &
Isolation levels.

44.) Metadata

- Database Metadata
- Result Set Metadata
- Parameter Metadata

45.) JDBC with Non-Traditional database like Excel.

46.) Result Set Types

- Forward only & Scrollable
- Scroll sensitive & Scroll Insensitive
- Read only & Updatable

47.) ResultSet Holdability

48.) Rowsets vs. ResultSets

49.) Working with ~~all~~ all Rowsets

50.) Miscellaneous Topics like RowID etc.

⇒ when we need to show or present something to the end user, we use JSP.

⇒ when we need to do some processing on the server, we use Servlet.

⇒ when we need to connect to the Database, we use JDBC.

Editions of Java

① Java Standard Edition (J2SE / ~~JSE~~) ⇒

Core Java + JDBC

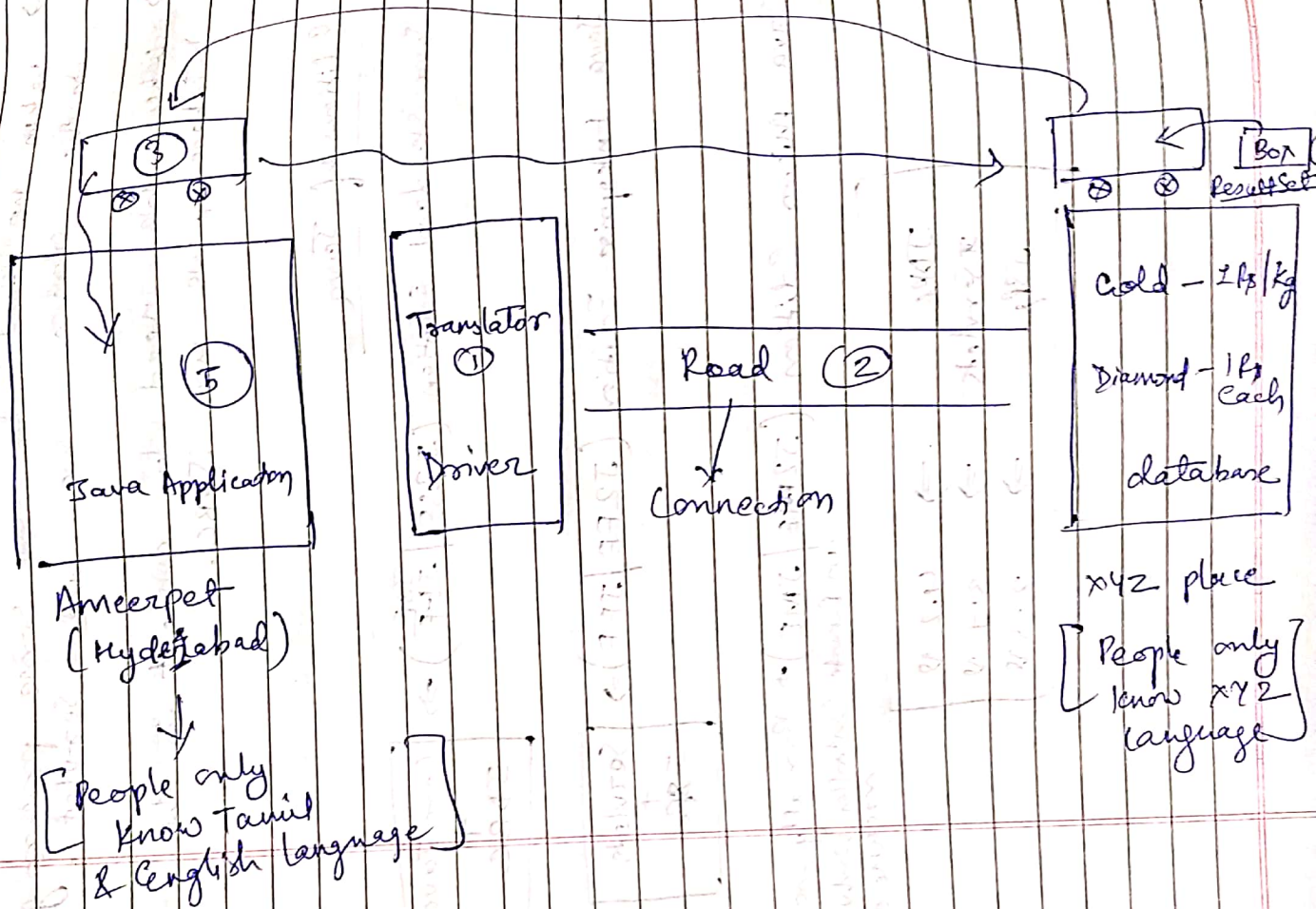
② Java Enterprise Edition (J2EE / JEE) ⇒

Servlets + JSPs

③ Java Micro Edition (J2ME / JME) ⇒ Basically used in Remote Controller, Washing machine etc.

JDBC	⇒	4.2 v
Servlets	⇒	3.1 v
JSPs	⇒	2.3 v

JDBC in Simple Way



Basic terminology of JDBC

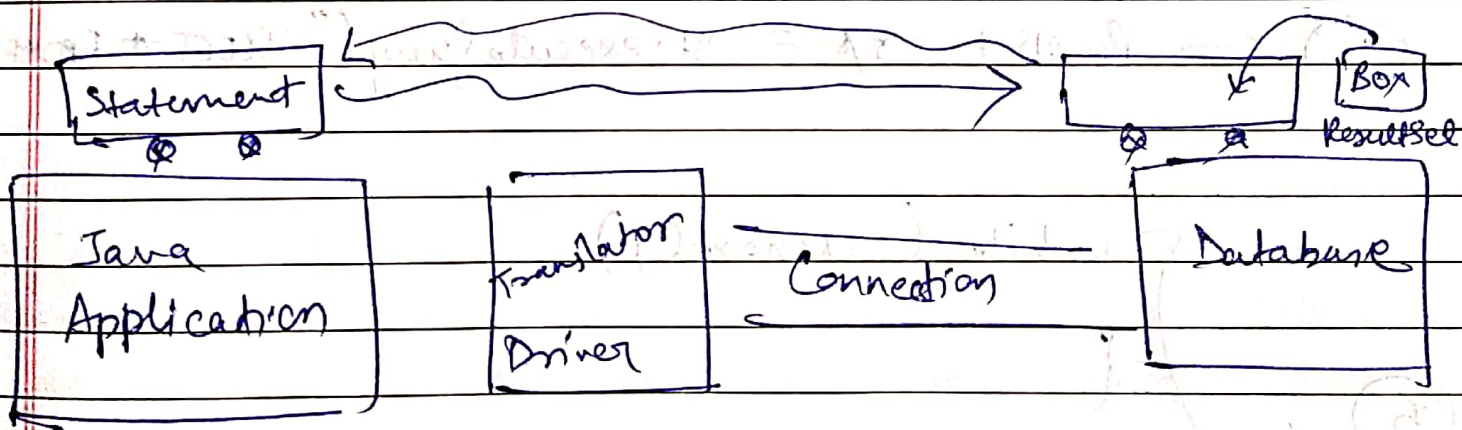
- ⇒ Here we have Java application & Database
- ⇒ Some translator must be required to convert Java calls to database calls & database calls to Java calls, that translator is nothing but driver.
- ⇒ To reach database some network socket must be required, that network socket is nothing but Connection (Road - ②)
- ⇒ Send SQL query to database and to bring results from database to Java application, some vehicle must be required, that vehicle is nothing but Statement Object.
- ⇒ After executing our query, database will store result in a box. That box is nothing but ResultSet.

Basic Components of JDBC Application

- ① Driver → Translator (Java calls to DB calls, & DB calls to Java calls)
- ② Connection → Road (From Java to DB some network socket is required, that network socket or medium is nothing but Connection object).
- ③ Statement → Vehicle (To send queries from Java to DB and then bring results from DB to Java, Statement Object is used).
- ④ ResultSet → Box (Set of Results, hold results of SQL Query. After executing SQL query, database will store result inside a box, that box is nothing but ResultSet).

Steps to develop JDBC Application

- ① Load & Register Driver.
- ② Establish Connection between Java application & Database.
- ③ Creation of Statement Object.
- ④ Send & Execute SQL Query.
- ⑤ Process Result from Result Set.
- ⑥ Close Connection.



```
import java.sql.*
```

```
class JdbcDemo
```

```
{
```

```
    public static void main(String[] args) throws Exception
```

```
{
```

① ← `Class.forName("oracle.jdbc.OracleDriver");`

② ← `Connection con = DriverManager.getConnection(url, user, pass);`

③ ← `Statement st = con.createStatement();`

④ ← `ResultSet rs = st.executeQuery("SELECT * FROM EMPLOYEES");`

```
    while (rs.next())
```

⑤ ← `{`
 `System.out.println(rs.getInt(1) + " : " + rs.getString(2) + " : " + rs.getDouble(3));`
 `}`

⑥ ← `con.close();`

```
    }
```

```
}
```