JDBC API is defined in the following packages

* java.sql and javax.sql

**Key classes**

* java.sql.DriverManager
* java.sql.Connection
* java.sql.Statement
* java.sql.ResultSet
* javax.sql.DataSource (for connection pooling)

**Development Process**

1. Get a connection to database
2. Create a Statement object
3. Execute SQL query
4. Process Result Set

**Step 1: Get a Connection to Database**

**Basic syntax**

Jdbc:<driver protocol>:<driver connection details>

**Examples**

MS SQL Server: jdbc:odbc:DemoDSN

Oracle: jdbc:oracle:thin@myserver:1521:demodb

MySQL: jdbc:mysql://localhost:3306/demodb

MySQL:

Import.sql.\*;

…

String dbUrl = “jdbc:msql://localhost:3306/demo”;

String user = “student”;

String pass = “student”;

Connection myConn = DriverManager.getConnection(dbURL, user, pass);

Failure to connect will throw an exception:

* Java.sql.SQLException: bad url or credentials
* Java.lang.ClassNotFoundException: JDBC driver not in classpath

**Step 2: Create a Statement object**

* The statement object is based on connection
  + It will be used later to execute SQL query

Import.sql.\*;

…

String dbUrl = “jdbc:msql://localhost:3306/demo”;

String user = “student”;

String pass = “student”;

Connection myConn = DriverManager.getConnection(dbURL, user, pass);

Statement myStmt = myConn.createStatement();

**Step 3: Execute SQL Query**

* Pass in your SQL query

Import.sql.\*;

…

String dbUrl = “jdbc:msql://localhost:3306/demo”;

String user = “student”;

String pass = “student”;

Connection myConn = DriverManager.getConnection(dbURL, user, pass);

Statement myStmt = myConn.createStatement();

ResultSet myRs = myStmt.executeQuery(“select \* from employees”);

**Step 4: Process the Result Set**

* Result Set is initially placed before first row
* Method: ResultSet.next()
  + Moves forward one row
  + Returns true if there are more rows to process
* Looping through a result set

…

ResultSet myRs = myStmt.executeQuery(“select \* from employees”);

while (myRs.next()) {

// read data from each row

}

* Collection of methods for reading data
  + getXXX(columnName)
  + getXXX(columnIndex) one-based

…

ResultSet myRs = myStmt.executeQuery(“select \* from employees”);

while (myRs.next()) {

System.out.println(“myRs.getString(“last\_name”));

System.out.println(“myRs.getString(“first\_name”));

}