

NGÔN NGỮ LẬP TRÌNH JAVA

Nội dung

Cơ bản về ngôn ngữ lập trình Java

Lập trình hướng
đối tượng

Biến, từ khoá,
kiểu dữ liệu

Biểu thức, các
cấu trúc điều
khiển

Dữ liệu kiểu
mảng

Các khía cạnh nâng cao của lập trình hướng đối tượng

Thiết kế lớp

Thiết kế lớp
nâng cao

Xử lý ngoại lệ

Generics

Java Collection
Framework

Multithread&
Concurrency

Java Database
Programming

Xây dựng ứng dụng đồ họa Java

LT đồ họa với
AWT

LT đồ họa với
SWING

JAVA DATABASE PROGRAMMING

Nội dung

1. Giới thiệu về CSDL và ngôn ngữ SQL
2. Giới thiệu về MySQL
3. Cài đặt và vận hành MySQL
4. Cài đặt MySQL JDBC Driver
5. Lập trình với JDBC

1. Giới thiệu về CSDL và SQL

1.1 Cơ sở dữ liệu quan hệ

- ❑ Một CSDLQH thực hiện việc tổ chức dữ liệu theo các bảng
- ❑ Một bảng có các hàng (bản ghi) và các cột (trường)
- ❑ Các bảng có thể được kết nối với nhau dựa trên các cột chung để loại trừ dữ liệu dư thừa và đảm bảo tính toàn vẹn dữ liệu.
- ❑ Một số hệ CSDLQH:
 - ❑ Oracle, IBM, DB2, MS SQL Server & Access
 - ❑ Sap Sybase, Teradata,
 - ❑ Free : MySQL, mSQL (miniSQL), SQLite, Apache OpenOffice's Base

1.2. Structure Query Language

- ❑ Một ngôn ngữ cấp cao, Structure Query Language, đã được thiết kế cho việc
 - ❑ cấu trúc hoá các csdlqh.
 - ❑ Tạo, cập nhật, đọc và xoá các bản ghi
- ❑ SQL định nghĩa một tập các lệnh
 - ❑ SELECT, INSERT, UPDATE, DELETE, CREATE TABLE, DROP TABLE, ...
- ❑ Một hệ thống CSDLQH gồm nhiều database, mỗi database lại bao gồm một hoặc nhiều bảng

Ví dụ SQL (1)

Database: **studentdb**

Table: **class101**

+-----+-----+-----+			
id (INT)	name (VARCHAR(50))	gpa (FLOAT)	
+-----+-----+-----+			
1001	Tan Ah Teck	4.5	
1002	Mohammed Ali	4.8	
1003	Kumar	4.8	
1004	Kevin Jones	4.6	
+-----+-----+-----+			

SQL cung cấp khả năng tương tác với CSDLQH một cách dễ dàng và trực quan

Ví dụ SQL (2)

SELECT command

```
-- Syntax
SELECT column1, column2,... FROM tableName WHERE criteria
SELECT * FROM tableName WHERE criteria -- Wildcard * for all Columns
```

```
-- Examples
```

```
SELECT name, gpa FROM class101
```

```
-- Select columns name and gpa from table class101.
```

name	gpa
Tan Ah Teck	4.5
Mohammed Ali	4.8
Kumar	4.8
Kevin Jones	4.6

```
SELECT * FROM class101
```

```
-- Select ALL columns from table class101.
```

```
-- The wildcard * denotes all the columns.
```

id	name	gpa
1001	Tan Ah Teck	4.5
1002	Mohammed Ali	4.8
1003	Kumar	4.8
1004	Kevin Jones	4.6

Ví dụ SQL (3)

SELECT command

```
SELECT name, gpa FROM class101 WHERE gpa >= 4.7
```

```
-- Select columns name and gpa, where the rows meet the criteria.
```

```
-- You can compare numbers using =, >, <, >=, <=, <> (!=)
```

```
+-----+-----+
| name          | gpa  |
+-----+-----+
| Mohammed Ali  | 4.8  |
| Kumar         | 4.8  |
+-----+-----+
```

```
SELECT name, gpa FROM class101 WHERE name = 'Tan Ah Teck'
```

```
-- Full-match (= or !=) on string. Strings are enclosed in single quotes.
```

```
+-----+-----+
| name          | gpa  |
+-----+-----+
| Tan Ah Teck   | 4.5  |
+-----+-----+
```

Ví dụ SQL (4)

SELECT command

```
SELECT name FROM class101 WHERE name LIKE 'k%'
```

```
-- Use "LIKE" for Pattern-matching, with
-- wildcard % matches zero or more (any) characters;
-- wildcard _ matches one (any) character.
```

```
+-----+
| name   |
+-----+
| Kumar  |
| Kevin Jones |
+-----+
```

```
SELECT * FROM class101 WHERE gpa > 4 AND name LIKE 'k%' ORDER BY gpa DESC, name ASC
```

```
-- Use AND, OR, NOT to combine simple conditions.
-- Order the results in DESC (descending) or ASC (Ascending)
```

```
+-----+-----+-----+
| id    | name       | gpa  |
+-----+-----+-----+
| 1003  | Kumar      | 4.8  |
| 1004  | Kevin Jones | 4.6  |
+-----+-----+-----+
```

Ví dụ SQL (5)

DELETE command

```
-- Syntax
```

```
DELETE FROM tableName WHERE criteria
```

```
-- Examples
```

```
DELETE FROM class101
```

```
-- Delete ALL rows from the table class101! Beware that there is NO UNDO!
```

```
DELETE FROM class101 WHERE id = 33
```

```
-- Delete rows that meet the criteria.
```

Ví dụ SQL (6)

INSERT command

```
-- Syntax
INSERT INTO tableName VALUES (firstColumnValue, ..., lastColumnValue)           -- All columns
INSERT INTO tableName (column1, column2, ...) VALUES (value1, value2, ...)    -- Selected Columns

-- Example
INSERT INTO class101 VALUES (1001, 'Tan Ah Teck', 4.5)
    -- List value of all columns.
INSERT INTO class101 (name, gpa) VALUES ('Peter Jones', 4.55)
    -- Missing fields will be set to their default values or NULL
```

Ví dụ SQL (7)

UPDATE command

-- Syntax

UPDATE *tableName* SET *column* = *value* WHERE *criteria*

-- Examples

UPDATE class101 SET gpa = 5.0 -- ALL rows

UPDATE class101 SET gpa = gpa + 1.0 WHERE name = 'Tan Ah Teck' -- Selected rows

Ví dụ SQL (8)

CREATE TABLE command

-- Syntax

```
CREATE TABLE tableName (columnName column1Type, column2Name column2Type, ...)
```

-- Examples

```
CREATE TABLE class101 (id INT, name VARCHAR(50), gpa FLOAT)
```

Ví dụ SQL (9)

DROP TABLE command

```
-- Syntax
```

```
DROP TABLE tableName  -- Delete table
```

```
-- Example
```

```
DROP TABLE class101  -- Delete the table. Beware that there is No UNDO!!!
```

2. Giới thiệu về MySQL (1)

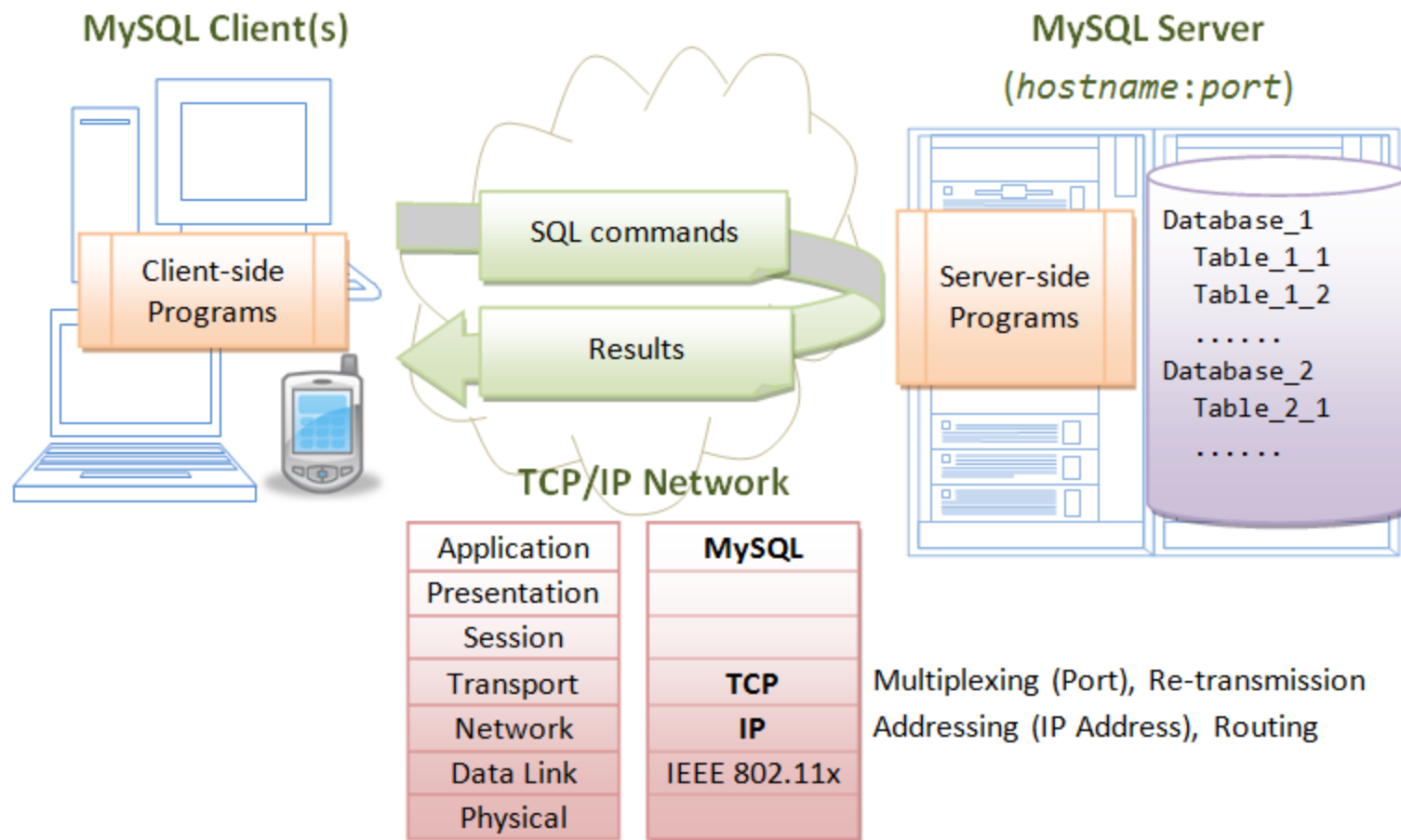
- ❑ SQL là một ngôn ngữ để tương tác với các CSDLQH
- ❑ MySQL là một hệ CSDLQH:
- ❑ Được phát triển bởi Michael “Monty” Widenius và David Axmark từ 1995
- ❑ Đầu tiên được sở hữu bởi công ty MySQL AB, Thụy Sĩ,
- ❑ Sau đó được Sun Microsystems mua lại vào 2008
- ❑ Và rồi bị thu tóm bởi Oracle vào 2010

2. Giới thiệu về MySQL (2)

❑ Đặc điểm

- ❑ Được sử dụng nhiều nhất, mã nguồn mở và miễn phí
- ❑ Nhưng thành công còn vì các đặc tính của nó: Nhanh, dễ sử dụng, tin cậy, hiệu năng cao, tính kết nối cao (connectivity), tính khả chuyển cao (portability), tính bảo mật cao, kích thước nhỏ và nhiều tính năng
- ❑ Hỗ trợ tất cả các đặc trưng trong một hệ CSDLQH hiệu năng cao:
 - ❑ transactions, foreign key, replication, subqueries, stored procedures, views and triggers.
- ❑ Thường được triển khai trong các môi trường
 - ❑ LAMP (Linux-Apache-MySQL-PHP),
 - ❑ WAMP (Windows-Apache-MySQL-PHP),
 - ❑ hoặc MAMP (Mac-Linux-Apache-MySQL-PHP)

2. Giới thiệu về MySQL



- ❑ MySQL vận hành theo mô hình client-server thông qua hệ thống TCP/IP
- ❑ MySQL server có địa chỉ IP và một cổng TCP tương ứng
- ❑ Cổng TCP mặc định là 3306, nhưng có thể thay đổi <1024-65536>

3. Cài đặt và vận hành MySQL

3.1. Download & install MySQL

☐ Download

- ☐ <http://dev.mysql.com/downloads/mysql/>

- ☐ MySQL Community Server (GPL)

- ☐ **ZIP ARCHIVE** (mysql-5.6.17-win32.zip about 212 MB).

☐ UNZIP into a directory of your choice.

- ☐ "d:\myProject" (or "c:\myproject")

- ☐ MySQL will be unzipped as "d:\myProject\mysql-5.617-win32".

- ☐ MySQL installed directory = **<MYSQL_HOME>**

☐ MySQL Distribution

- ☐ Database server (mysqld);

- ☐ A command-line client (mysql);

- ☐ Utilities: Database administration (mysqladmin), backup/restore (mysqldump), and others;

- ☐ Client libraries for you to write your own client.

3. Cài đặt và vận hành MySQL

3.2. Tạo file cấu hình

- my-default.ini

```
# Save as "my.ini" in your MySQL installed directory
[mysqld] ← Server program
# Set MySQL base (installed) directory
# @@ Change to your MySQL installed directory @@
basedir=d:/myProject/mysql

# Set MySQL data directory
# @@ Change to sub-directory "data" of your MySQL installed directory @@
datadir=d:/myProject/mysql/data

# Run the server on this TCP port number
port=8888 ← Default port=3306

[client] ← Client program
# MySQL client connects to the server running on this TCP port number
port=8888
```

3. Cài đặt và vận hành MySQL

3.3. Startup/shutdown server

```
-- Change the current directory to MySQL's "bin" directory
-- Assume that the MySQL installed directory is "d:\myProject\mysql"
Prompt> d:                                     -- Set the current drive
D:\...> cd \myProject\mysql\bin               -- Change Directory to the MySQL's bin directory

-- Start the MySQL Database Server
D:\myProject\mysql\bin> mysqld --console
.....
.....
XXXXXXXX XX:XX:XX [Note] mysqld: ready for connections.
Version: '5.6.xx-community' socket: '' port: 8888 MySQL Community Server (GPL)
```

The quickest way to shut down the database server is to press *control-c* to initiate a *normal shutdown*.

Observe these messages from the MySQL server console:

```
XXXXXXXX XX:XX:XX [Note] mysqld: Normal shutdown
.....
XXXXXXXX XX:XX:XX InnoDB: Starting shutdown...
XXXXXXXX XX:XX:XX InnoDB: Shutdown completed; log sequence number 0 44233
.....
XXXXXXXX XX:XX:XX [Note] mysqld: Shutdown complete
```

3. Cài đặt và vận hành MySQL

3.4. Khởi tạo client

```
-- Change the current directory to <MYSQL_HOME>\bin.
-- Assume that the MySQL is installed in "d:\myProject\mysql".
Prompt> d:                                     -- Change the current drive
D:\...> cd \myProject\mysql\bin                -- Change Directory to YOUR MySQL's "bin" directory

-- Start a client with superuser "root"
D:\myProject\mysql\bin> mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.39-community MySQL Community Server (GPL)
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

-- Client started. The prompt changes to "mysql>".
-- You can now issue SQL commands.
mysql>
```

```
mysql> status
-----
mysql  Ver 14.14 Distrib 5.1.51, for Win32 (ia32)
.....
Current user:          root@localhost
TCP port:              8888      -- for Windows only
.....
```

Other commands:
 >show databases;
 >use mysql;

3. Cài đặt và vận hành MySQL

3.5. Đặt password cho user root (1/2)

```
-- Query all the users and their passwords from database "mysql" table "user"
-- End the commands with a semi-colon (;)
```

```
mysql> select host, user, password from mysql.user;
```

```
+-----+-----+-----+
| host      | user  | password |
+-----+-----+-----+
| localhost | root  |          |
| 127.0.0.1 | root  |          |
| ::1       | root  |          |
| localhost |       |          |
+-----+-----+-----+
```

← Anonymous User : dangerous

```
mysql> set password for 'root'@'127.0.0.1'=password('xxxx');
Query OK, 0 rows affected (0.00 sec)
```

```
-- Set password for 'root'@'localhost'
```

```
mysql> set password for 'root'@'localhost'=password('xxxx');
Query OK, 0 rows affected (0.00 sec)
```

```
-- Set password for 'root'@'::1'
```

```
mysql> set password for 'root'@'::1'=password('xxxx');
Query OK, 0 rows affected (0.00 sec)
```

3. Cài đặt và vận hành MySQL

3.5. Đặt password cho user root (2/2)

```
mysql> select host, user, password from mysql.user;
```

host	user	password
localhost	root	*XXX
127.0.0.1	root	*XXX
:::1	root	*XXX
localhost		

```
-- logout and terminate the client program
```

```
mysql> quit
```

```
Bye
```


3. Cài đặt và vận hành MySQL

3.6. Khởi tạo client với password

```
-- Change the current working directory to <MYSQL_HOME>\bin
D:\myProject\mysql\bin> mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.
.....
-- client started, ready to issue SQL command
mysql>
```


3. Cài đặt và vận hành MySQL

3.8. Tạo User mới

```
-- Create a new user called "myuser", which can login from localhost, with password "xxxx"
mysql> create user 'myuser'@'localhost' identified by 'xxxx';
Query OK (0.01 sec)

-- Grant permission to myuser
mysql> grant all on *.* to 'myuser'@'localhost';
Query OK (0.01 sec)

-- Query all users and passwords
mysql> select host, user, password from mysql.user;
```

host	user	password
localhost	root	*XX
127.0.0.1	root	*XX
:::1	root	*XX
localhost	myuser	*XX

```
4 rows in set (0.00 sec)
```

3. Cài đặt và vận hành MySQL

3.9. Tạo CSDL mới

```
-- Start a client, if it is not started
> mysql -u myuser -p      // Windows
$ ./mysql -u myuser -p    // Macs

-- Create a new database called "studentdb"
mysql> create database if not exists studentdb;
Query OK, 1 row affected (0.08 sec)

-- list all the databases in this server
mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema      |
| studentdb               |
| test                    |
+-----+
5 rows in set (0.07 sec)
```

3. Cài đặt và vận hành MySQL

3.9. Tạo CSDL mới

```
-- Use "studentdb" database as the current (default) database
-- You can refer to tables in current database by tablename, instead of databasename.tablename
mysql> use studentdb;
Database changed

-- Remove the table "class101" in the current database if it exists
mysql> drop table if exists class101;
Query OK, 0 rows affected (0.15 sec)

-- Create a new table called "class101" in the current database
-- with 3 columns of the specified types
mysql> create table class101 (id int, name varchar(50), gpa float);
Query OK, 0 rows affected (0.15 sec)

-- List all the tables in the current database "studentdb"
mysql> show tables;
+-----+
| Tables_in_studentdb |
+-----+
| class101             |
+-----+
1 row in set (0.00 sec)
```

3. Cài đặt và vận hành MySQL

3.9. Tạo CSDL mới

```
-- Describe the "class101" table (list its columns' definitions)
mysql> describe class101;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | YES  |     | NULL    |       |
| name  | varchar(50)   | YES  |     | NULL    |       |
| gpa   | float         | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.04 sec)
```

```
-- Insert a row into "class101" table.
-- Strings are to be single-quoted. No quotes for INT and FLOAT.
mysql> insert into class101 values (11, 'Tan Ah Teck', 4.8);
Query OK, 1 row affected (0.03 sec)
```

```
-- Insert another row
mysql> insert into class101 values (22, 'Mohamed Ali', 4.9);
Query OK, 1 row affected (0.03 sec)
```

```
-- Select all columns (*) from table "class101"
mysql> select * from class101;
+-----+-----+-----+
| id | name          | gpa |
+-----+-----+-----+
| 11 | Tan Ah Teck  | 4.8 |
| 22 | Mohamed Ali  | 4.9 |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

3. Cài đặt và vận hành MySQL

3.9. Tạo CSDL mới

```
-- Select columns from table "class101" with criteria
mysql> select name, gpa from class101 where gpa > 4.85;
+-----+-----+
| name      | gpa  |
+-----+-----+
| Mohamed Ali | 4.9  |
+-----+-----+
1 rows in set (0.00 sec)

-- Update selected records
mysql> update class101 set gpa = 4.4 where name = 'Tan Ah Teck';
Query OK, 1 row affected (0.03 sec)

mysql> select * from class101;
+----+-----+-----+
| id | name      | gpa  |
+----+-----+-----+
| 11 | Tan Ah Teck | 4.4  |
| 22 | Mohamed Ali | 4.9  |
+----+-----+-----+
2 rows in set (0.00 sec)

-- delete selected records
mysql> delete from class101 where id = 22;
Query OK, 1 row affected (0.03 sec)
```

```
mysql> select * from class101;
+----+-----+-----+
| id | name      | gpa  |
+----+-----+-----+
| 11 | Tan Ah Teck | 4.4  |
+----+-----+-----+
1 rows in set (0.00 sec)
```

3. Cài đặt và vận hành MySQL

3.9. Tạo CSDL mới

```
-- Select columns from table "class101" with criteria
mysql> select name, gpa from class101 where gpa > 4.85;
+-----+-----+
| name      | gpa  |
+-----+-----+
| Mohamed Ali | 4.9  |
+-----+-----+
1 rows in set (0.00 sec)

-- Update selected records
mysql> update class101 set gpa = 4.4 where name = 'Tan Ah Teck';
Query OK, 1 row affected (0.03 sec)

mysql> select * from class101;
+----+-----+-----+
| id  | name      | gpa  |
+----+-----+-----+
| 11  | Tan Ah Teck | 4.4  |
| 22  | Mohamed Ali | 4.9  |
+----+-----+-----+
2 rows in set (0.00 sec)

-- delete selected records
mysql> delete from class101 where id = 22;
Query OK, 1 row affected (0.03 sec)
```

```
mysql> select * from class101;
+----+-----+-----+
| id  | name      | gpa  |
+----+-----+-----+
| 11  | Tan Ah Teck | 4.4  |
+----+-----+-----+
1 rows in set (0.00 sec)
```


3. Cài đặt và vận hành MySQL

3.9. Tạo CSDL mới

```
-- You can store SQL commands in a file (called SQL script) and run the script.
-- For example, use a text editor to create a SQL script called "mycommands.sql"
-- containing the following three SQL statements.
-- (For Windows) Save the file under "d:\myProject".
-- (For Macs) Save the file under "Documents".
insert into class101 values (33, 'Kumar', 4.8);
insert into class101 values (44, 'Kevin', 4.6);
Select * from class101;

-- Use the following "source" command to run the SQL script.
-- You need to provide the full path to the script.
-- (For Windows) The filename is d:/myProject/mycommands.sql.
-- (For Macs) The filename is ~/Documents/mycommands.sql

mysql> source d:/myProject/mycommands.sql // For Windows (Use forward slash here!)
mysql> source ~/Documents/mycommands.sql // For Macs
Query OK, 1 row affected (0.00 sec) -- INSERT command output
Query OK, 1 row affected (0.00 sec) -- INSERT command output
+-----+-----+-----+ -- SELECT command output
| id   | name       | gpa   |
+-----+-----+-----+
| 11   | Tan Ah Teck | 4.4   |
| 33   | Kumar      | 4.8   |
| 44   | Kevin      | 4.6   |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

4. Cài đặt MySQL JDBC Driver

- ❑ Download the latest MySQL JDBC driver from <http://dev.mysql.com/downloads>
 - ❑ "MySQL Connectors" ⇒ "Connector/J" ⇒ Connector/J 5.1.{xx} ⇒ select "Platform Independent"
 - ❑ ZIP Archive(e.g., "mysql-connector-java-5.1.{xx}.zip", where {xx} is the latest release number).
- ❑ UNZIP the download file into any temporary folder.
- ❑ Copy the JAR file "mysql-connector-java-5.1.{xx}-bin.jar" to your JDK's Extension Directory at "<JAVA_HOME>\jre\lib\ext" (where <JAVA_HOME> is the JDK installed directory, e.g., "c:\program files\java\jdk1.7.0_{xx}\jre\lib\ext").

5. Lập trình với JDBC

5.1 Tạo CSDL ebookshop (1/2)

Database: **ebookshop**

Table: **books**

id	title	author	price	qty
(INT)	(VARCHAR(50))	(VARCHAR(50))	(FLOAT)	(INT)
1001	Java for dummies	Tan Ah Teck	11.11	11
1002	More Java for dummies	Tan Ah Teck	22.22	22
1003	More Java for more dummies	Mohammad Ali	33.33	33
1004	A Cup of Java	Kumar	44.44	44
1005	A Teaspoon of Java	Kevin Jones	55.55	55

5. Lập trình với JDBC

5.1 Tạo CSDL ebookshop (2/2)

```
create database if not exists ebookshop;  
  
use ebookshop;  
  
drop table if exists books;  
create table books (  
    id int,  
    title varchar(50),  
    author varchar(50),  
    price float,  
    qty int,  
    primary key (id));  
  
insert into books values (1001, 'Java for dummies', 'Tan Ah Teck', 11.11, 11);  
insert into books values (1002, 'More Java for dummies', 'Tan Ah Teck', 22.22, 22);  
insert into books values (1003, 'More Java for more dummies', 'Mohammad Ali', 33.33, 33);  
insert into books values (1004, 'A Cup of Java', 'Kumar', 44.44, 44);  
insert into books values (1005, 'A Teaspoon of Java', 'Kevin Jones', 55.55, 55);  
  
select * from books;
```

5. Lập trình với JDBC

5.2 JDBC program template

❑ Một ứng dụng JDBC bao gồm các bước sau:

1. Cấp phát đối tượng Connection để kết nối tới CSDL
2. Cấp phát đối tượng Statement dưới đối tượng Connection vừa tạo
3. Viết câu truy vấn SQL và thực thi câu truy vấn
4. Xử lý kết quả truy vấn
5. Đóng Statement và Connection object để giải phóng tài nguyên

5. Lập trình với JDBC

5.3 Ví dụ với SQL SELECT

```
1  import java.sql.*;    // Use classes in java.sql package
2
3  // JDK 7 and above
4  public class JdbcSelectTest { // Save as "JdbcSelectTest.java"
5      public static void main(String[] args) {
6          try {
7              // Step 1: Allocate a database "Connection" object
8              Connection conn = DriverManager.getConnection(
9                  "jdbc:mysql://localhost:8888/ebookshop", "myuser", "xxxx"); // MySQL
10
11              // Step 2: Allocate a "Statement" object in the Connection
12              Statement stmt = conn.createStatement();
13          } {
14              // Step 3: Execute a SQL SELECT query, the query result
15              // is returned in a "ResultSet" object.
16              String strSelect = "select title, price, qty from books";
17              System.out.println("The SQL query is: " + strSelect); // Echo For debugging
18              System.out.println();
19
20              ResultSet rset = stmt.executeQuery(strSelect);
```

5. Lập trình với JDBC

5.3 Ví dụ với SQL SELECT

```
22         // Step 4: Process the ResultSet by scrolling the cursor forward via next().
23         // For each row, retrieve the contents of the cells with getXxx(columnName).
24         System.out.println("The records selected are:");
25         int rowCount = 0;
26         while(rset.next()) {    // Move the cursor to the next row
27             String title = rset.getString("title");
28             double price = rset.getDouble("price");
29             int qty = rset.getInt("qty");
30             System.out.println(title + ", " + price + ", " + qty);
31             ++rowCount;
32         }
33         System.out.println("Total number of records = " + rowCount);
34
35     } catch(SQLException ex) {
36         ex.printStackTrace();
37     }
38     // Step 5: Close the resources - Done automatically by try-with-resources
39 }
40 }
```

5. Lập trình với JDBC

5.4 Ví dụ với SQL UPDATE

```
1  import java.sql.*;    // Use classes in java.sql package
2
3  // JDK 7 and above
4  public class JdbcUpdateTest {    // Save as "JdbcUpdateTest.java"
5      public static void main(String[] args) {
6          try {
7              // Step 1: Allocate a database "Connection" object
8              Connection conn = DriverManager.getConnection(
9                  "jdbc:mysql://localhost:8888/ebookshop", "myuser", "xxxx"); // MySQL
10
11              // Step 2: Allocate a "Statement" object in the Connection
12              Statement stmt = conn.createStatement();
13          } {
14              // Step 3 & 4: Execute a SQL UPDATE via executeUpdate()
15              //   which returns an int indicating the number of rows affected.
16              // Increase the price by 7% and qty by 1 for id=1001
17              String strUpdate = "update books set price = price*0.7, qty = qty+1 where id = 1001";
18              System.out.println("The SQL query is: " + strUpdate); // Echo for debugging
19              int countUpdated = stmt.executeUpdate(strUpdate);
20              System.out.println(countUpdated + " records affected.");
```


5. Lập trình với JDBC

5.4 Ví dụ với SQL UPDATE

```
22      // Step 3 & 4: Issue a SELECT to check the UPDATE.
23      String strSelect = "select * from books where id = 1001";
24      System.out.println("The SQL query is: " + strSelect); // Echo for debugging
25      ResultSet rset = stmt.executeQuery(strSelect);
26      while(rset.next()) { // Move the cursor to the next row
27          System.out.println(rset.getInt("id") + ", "
28                          + rset.getString("author") + ", "
29                          + rset.getString("title") + ", "
30                          + rset.getDouble("price") + ", "
31                          + rset.getInt("qty"));
32      }
33      } catch(SQLException ex) {
34          ex.printStackTrace();
35      }
36      // Step 5: Close the resources - Done automatically by try-with-resources
37  }
38 }
```

5. Lập trình với JDBC

5.5 Ví dụ với SQL INSERT & DELETE

```
1  import java.sql.*;      // Use classes in java.sql package
2
3  // JDK 7 and above
4  public class JdbcInsertTest {    // Save as "JdbcUpdateTest.java"
5      public static void main(String[] args) {
6          try (
7              // Step 1: Allocate a database "Connection" object
8              Connection conn = DriverManager.getConnection(
9                  "jdbc:mysql://localhost:8888/ebookshop", "myuser", "xxxx"); // MySQL
10
11              // Step 2: Allocate a "Statement" object in the Connection
12              Statement stmt = conn.createStatement();
13          ) {
14              // Step 3 & 4: Execute a SQL INSERT|DELETE statement via executeUpdate(),
15              // which returns an int indicating the number of rows affected.
16
17              // DELETE records with id>=3000 and id<4000
18              String sqlDelete = "delete from books where id>=3000 and id<4000";
19              System.out.println("The SQL query is: " + sqlDelete); // Echo for debugging
20              int countDeleted = stmt.executeUpdate(sqlDelete);
21              System.out.println(countDeleted + " records deleted.\n");
22          }
```

5. Lập trình với JDBC

5.5 Ví dụ với SQL INSERT & DELETE

```
23      // INSERT a record
24      String sqlInsert = "insert into books " // need a space
25          + "values (3001, 'Gone Fishing', 'Kumar', 11.11, 11)";
26      System.out.println("The SQL query is: " + sqlInsert); // Echo for debugging
27      int countInserted = stmt.executeUpdate(sqlInsert);
28      System.out.println(countInserted + " records inserted.\n");
29
30      // INSERT multiple records
31      sqlInsert = "insert into books values "
32          + "(3002, 'Gone Fishing 2', 'Kumar', 22.22, 22),"
33          + "(3003, 'Gone Fishing 3', 'Kumar', 33.33, 33)";
34      System.out.println("The SQL query is: " + sqlInsert); // Echo for debugging
35      countInserted = stmt.executeUpdate(sqlInsert);
36      System.out.println(countInserted + " records inserted.\n");
37
38      // INSERT a partial record
39      sqlInsert = "insert into books (id, title, author) "
40          + "values (3004, 'Fishing 101', 'Kumar')";
41      System.out.println("The SQL query is: " + sqlInsert); // Echo for debugging
42      countInserted = stmt.executeUpdate(sqlInsert);
43      System.out.println(countInserted + " records inserted.\n");
```

5. Lập trình với JDBC

5.5 Ví dụ với SQL INSERT & DELETE

```
45      // Issue a SELECT to check the changes
46      String strSelect = "select * from books";
47      System.out.println("The SQL query is: " + strSelect); // For debugging
48      ResultSet rset = stmt.executeQuery(strSelect);
49      while(rset.next()) { // Move the cursor to the next row
50          System.out.println(rset.getInt("id") + ", "
51                          + rset.getString("author") + ", "
52                          + rset.getString("title") + ", "
53                          + rset.getDouble("price") + ", "
54                          + rset.getInt("qty"));
55      }
56  } catch(SQLException ex) {
57      ex.printStackTrace();
58  }
59  // Step 5: Close the resources - Done automatically by try-with-resources
60  }
61 }
```