

Course: Web Application Development

Lab Instructor: Vo Duy Khoi, MSc

Email: khoivd@hcmiu.edu.vn

Lab 5 - Database (MySQL), JSP

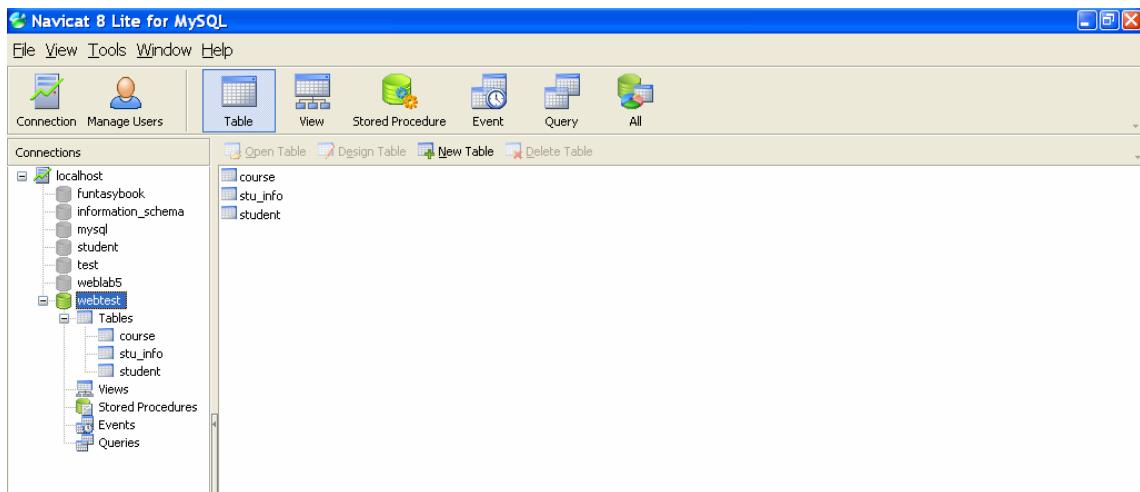
Content:

- Introduction DBMS - MySQL
- Java Server Pages.
- Connection JSP-MySQL

Duration: 3 hours

Part 1: Introduction Database Management System MySQL

Install (MySQL Version 6, admin tool - use Graphic User Interface). Refer from this site:
<http://dev.mysql.com/doc/>



You can use the command line on window to create database for your website

A screenshot of a Windows command prompt window titled 'C:\WINDOWS\system32\cmd.exe - mysql -u root -p'. The window shows the MySQL command-line interface. The user enters 'mysql -u root -p' and provides a password. The MySQL version 6.0.3 is displayed. The user then runs 'show databases;' which lists databases: information_schema, funtasybook, mysql, student, test, weblab5, and webtest. Next, 'use webtest;' is run, changing the database. 'show tables;' is then run, showing tables: course, stu_info, and student. Finally, an 'insert' query is run: 'mysql> insert into student<IDStudent,name,age>values<'IT070056','ABC',22>;'. The response shows 'Query OK, 1 row affected <0.47 sec>'. The MySQL prompt 'mysql>' is visible at the bottom.

Basic commands on mysql command line:

mysql -u root -p mysql	Logon to MySQL
mysqladmin -u root shutdown	Shutdown MySQL
show databases;	View Databases
show tables;	View Tables
select * from tables_name;	View content of tables
exit;	Exit and close MySQL

Get help from mysql command

help [command]	Help for syntax of a command on server side
?	Help for syntax of commands on client side

Example: creates a database for registry course

```
-- 
-- Table structure for table 'course'
--

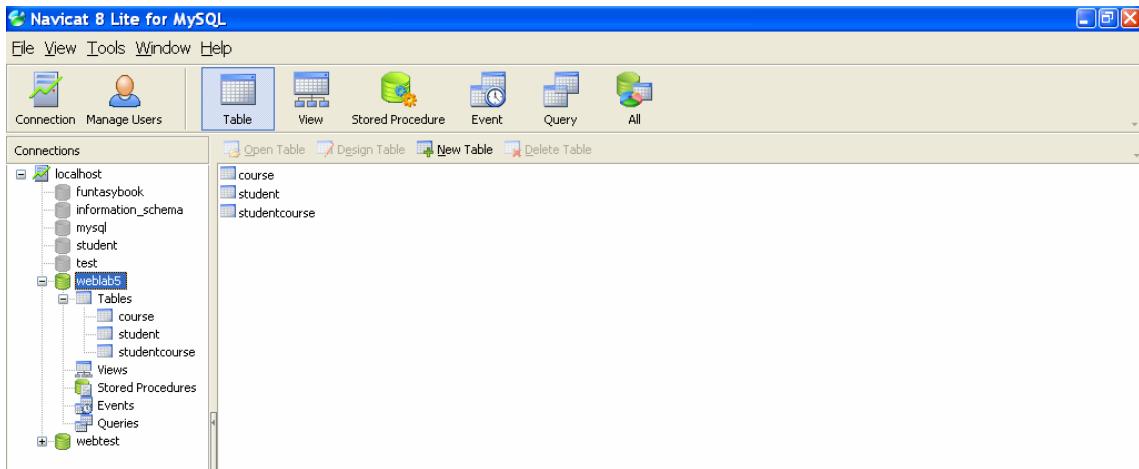
CREATE TABLE IF NOT EXISTS 'course' (
  'CourseID' bigint(20) NOT NULL auto_increment,
  'CourseName' varchar(255) NOT NULL,
  PRIMARY KEY  ('CourseID'),
  KEY 'Course_CourseID_INDEX' ('CourseID')
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

-- 
-- Table structure for table 'student'
--

CREATE TABLE IF NOT EXISTS 'student' (
  'StudentID' int(11) NOT NULL auto_increment,
  'StudentName' varchar(255) NOT NULL,
  PRIMARY KEY  ('StudentID')
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;

-- 
-- Table structure for table 'studentcourse'
--

CREATE TABLE IF NOT EXISTS 'studentcourse' (
  'StudentID' bigint(20) NOT NULL,
  'CourseID' bigint(20) NOT NULL,
  PRIMARY KEY  ('StudentID','CourseID'),
  KEY 'StudentCourse_StudentID_INDEX' ('StudentID'),
  KEY 'StudentCourse_CourseID_INDEX' ('CourseID')
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```



Part 2: Java Server Pages

In this part, you will design many jsp pages. Using java source code, embed to html page. **The first page** is to check username and password to login another page.



If user input valid username and password, go to welcome page (homepage.java), else there is an error message.

The second page is the homepage of your website, there is also a hit counter

Now, create many jsp pages, use form, and base on the following source code below:

```
<%
    //Get value from text-box
    String value = request.getParameter("name_text_box");
    out.print("Hello "+value+"!<br>");

    //Get value from radio button
    String radioButton = request.getParameter("name_radio_button");
    out.print("The gender is: "+radioButton+"<br>");

    //Get value from check-box button
    String[] courses;
    courses = request.getParameterValues("name_check_box");
    if (courses != null)
    {
        out.print("The course: <br>");
        for (int i = 0; i < courses.length; i++)
    }
}
```

```

        out.println (courses[i]+<br>);
    }
} else out.println ("<b>none</b>");
%>

```

Part 3: Connection JSP - MySQL

Seven basic steps in using JDBC:

- Load the driver
- Define the connection URL
- Establish the connection
- Create a statement object
- Execute a query
- Process a results
- Close the connection

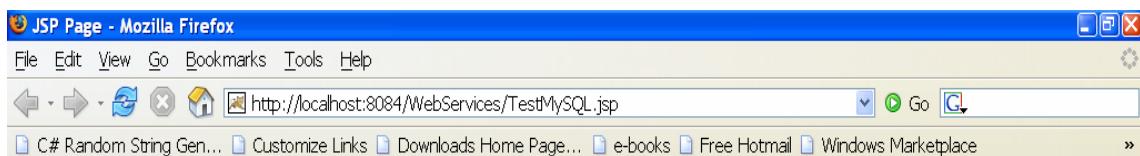
Use the following source below to test your system

```

<%
String connectionURL="jdbc:mysql://localhost:3306/weblab5?user=root;password="; //step 2
Connection connection = null; //step 3
Statement statement = null; //step 4
ResultSet rs = null;
%>
-----
<%
//Class.forName("org.gjt.mm.mysql.Driver").newInstance();
Class.forName("com.mysql.jdbc.Driver").newInstance(); //step 1
connection = DriverManager.getConnection(connectionURL, "root", "");
statement = connection.createStatement();
rs = statement.executeQuery("SELECT * FROM course"); //step 5

while (rs.next()) {
    out.println(rs.getString("CourseID"));
    out.println(rs.getString("CourseName")+"<br>");
}
//step 6
rs.close(); //step 7
%>

```

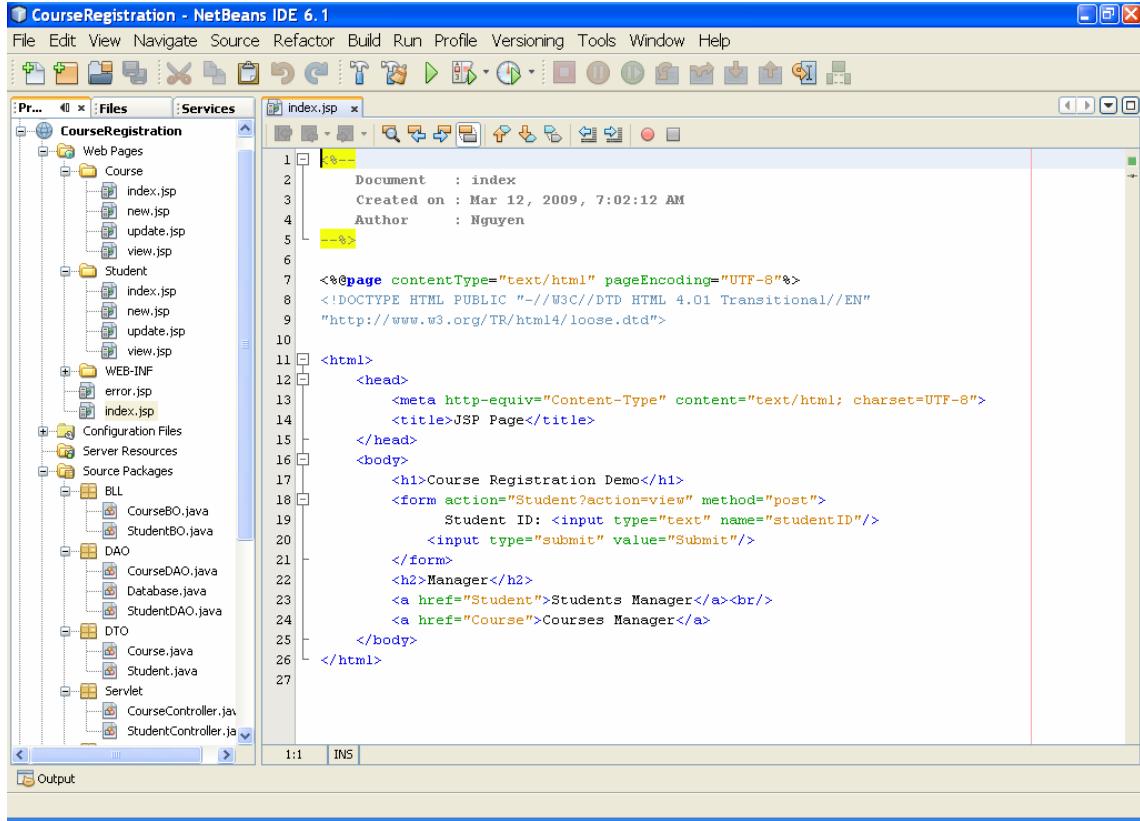


Testing Connection MySQL

1 Computer Graphics
2 Web

Now go on to develop your project base on the following interactions:

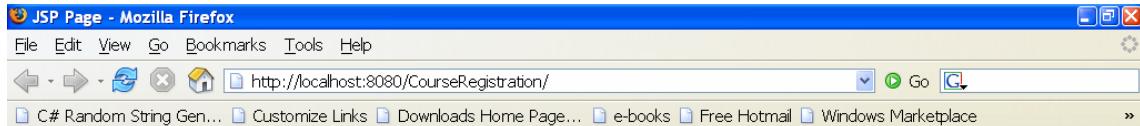
This is the structure of website:



The screenshot shows the NetBeans IDE interface with the title bar "CourseRegistration - NetBeans IDE 6.1". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Build, Run, Profile, Versioning, Tools, Window, Help. The toolbar has various icons for file operations. The left pane shows the project structure under "CourseRegistration": Web Pages (Course, Student), WEB-INF (error.jsp, index.jsp), Configuration Files, Server Resources, Source Packages (BLL, DAO, DTO, Servlet). The right pane displays the code editor for "index.jsp". The code is as follows:

```
1 Document : index
2 Created on : Mar 12, 2009, 7:02:12 AM
3 Author : Nguyen
4
5
6 <%@page contentType="text/html" pageEncoding="UTF-8"%>
7 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
8 "http://www.w3.org/TR/html4/loose.dtd">
9
10 <html>
11     <head>
12         <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
13         <title>JSP Page</title>
14     </head>
15     <body>
16         <h1>Course Registration Demo</h1>
17         <form action="Student?action=view" method="post">
18             Student ID: <input type="text" name="studentID"/>
19             <input type="submit" value="Submit"/>
20         </form>
21         <h2>Manager</h2>
22         <a href="Student">Students Manager</a><br/>
23         <a href="Course">Courses Manager</a>
24     </body>
25 </html>
```

First page: allow to input Student ID and click Submit button to search



Course Registration Demo

Student ID:

Manager

[Students Manager](#)
[Courses Manager](#)

Second page: return information of registry course

JSP Page - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

HTTP://localhost:8080/CourseRegistration/Student?action=view

C# Random String Gen... Customize Links Downloads Home Page... e-books Free Hotmail Windows Marketplace >>

Student's Details

Student ID: 2
Student name: Nguyen Van Sinh

Select Course

Course: Computer Graphics Add

Registered Courses:

Course ID	Course Name	Action
1	Computer Graphics	Remove
2	Web	Remove

[homepage](#)

Third page: when click the link “Student Manager”:

JSP Page - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

HTTP://localhost:8080/CourseRegistration/Student

C# Random String Gen... Customize Links Downloads Home Page... e-books Free Hotmail Windows Marketplace >>

Student List

Student ID	Student Name	Action
2	Nguyen Van Sinh	Edit Delete
3	Le Van Son	Edit Delete

[New Student](#)

[homepage](#)

The page 4: when click the link “Course Manager”

Course List - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

HTTP://localhost:8080/CourseRegistration/Course

C# Random String Gen... Customize Links Downloads Home Page... e-books Free Hotmail Windows Marketplace >>

Course List

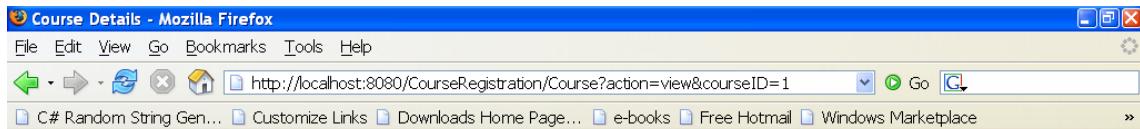
Course ID	Course Name	Action
1	Computer Graphics	Edit Delete
2	Web	Edit Delete

[New course](#)

[homepage](#)

In this page, user can click to “Edit” to modify or click to “Delete” to delete information.

The next page: from the above page, user click to the name of course to see “Course’s Detail”



Course's Details

Course ID: 1
Course name: Computer Graphics

Student list:

Student ID	Student Name	Action
2	Nguyen Van Sinh	Remove

[homepage](#)