Lab 10: JDBC

Objectives:

At the end of this lab, you should be able to:

1) Use prepared statement to interact with database.
2) Call stored procedures through JDBC.
3) Using JDBC Transactions
4) Display data collection in tabular form using JTable.

Prepared Statement

It is a precompiled SQL statement used to execute parameterized query. For example instead of using Instead of hardcoding the values in the SQL statement:

```
select * from emp where sal>2000 and job='MANAGER'
```

You can use placeholders where you can use question mark as placeholder symbol

```
select * from emp where sal>?  and job=?
```

Create the prepared statement

```
PreparedStatement stmt=conn.prepareStatement("select empno, ename,sal,hiredate from
emp where sal>? and job=? ");

stmt.setDouble(1,2000);
stmt.setString(2,"MANAGER");
```

Exercise 1:

Using the prepared statement above, write a program to display empno, ename, sal, hiredate of employees whose salaries are greater than 2000 and their jobs are 'MANAGER'.

Prepared Statement with DML:

The prepared statement also can be used with DML statements, for example

```
PreparedStatement stmt=conn.prepareStatement("insert into emp( empno, ename) values
(?,?)");
stmt.setDouble(1,222);
stmt.setString(2,"Ahmed");
stmt.executeUpdate();
```

Exercise 2:

Using prepared statement, write a program to insert three new employees to EMP table, insert only values for empno,ename, sal.

Calling Stored Procedure through JDBC

1- In SQL Developer, create a procedure that raiseSal that receives two paramters: deptno and number represents the percentage to raise the salary.

```
Create or replace procedure raiseSal(dno number, per number)
is
begin
update emp set sal=sal+sal*per/100
where deptno=dno;
end;
```

Exercise 3:

Write a complete java program to call the procedure above and raise the salaries of employees in dept. 20 by 5%.

To call this procedure from Java program, you need to create *CallableStatement*

```
CallableStatement stCall=conn.prepareCall("{call raiseSal(?,?)}");
        stCall.setInt(1, 20);
        stCall.setInt(2,5);
    stCall.execute();
```

Transaction Control in JDBC

Transaction: is a set of statements that either all of them are executed (commit) or none of them is executed (rollback).

In JDBC**, by default the** database connection is auto-commit, so if you want to change that, you should explicitly turn auto-commit to off

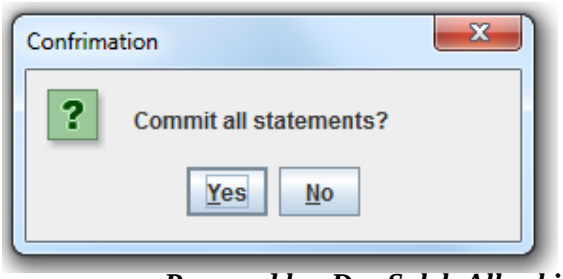
```
conn.setAutoCommit(false);
…….
…….
conn. commit( );
or
conn.rollback( );
```

Exercise 4:

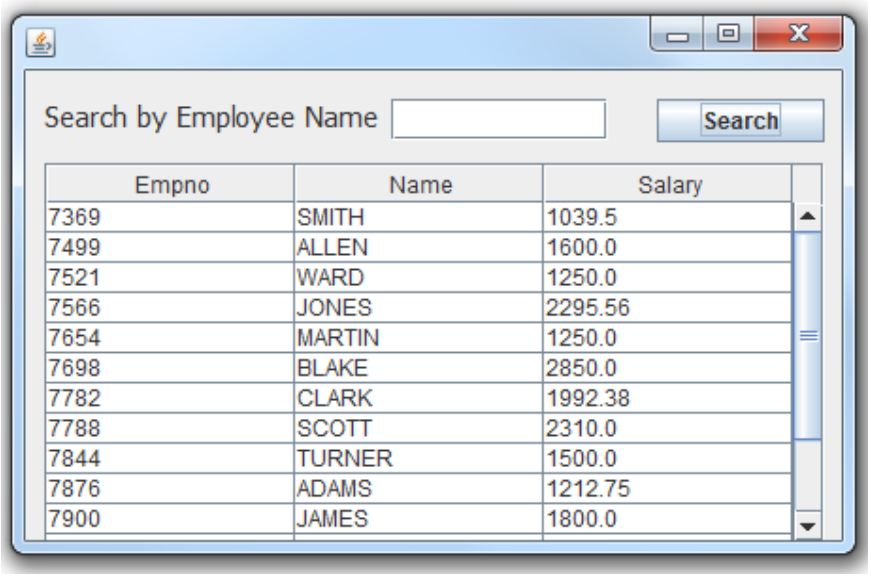
Using commit, and roll back Write Java program to insert

a) a new dept. to table DEPT as follows
   Deptno: 70 dname: 'HR' Loc: 'Doha'
b) a new employee to table EMP as follows:
   empn: 777 ename: ' Naser' sal:3000 deptno: 70.

Using *JOptionPane.showConfirmDialog*, let the user confirm these changes or ignore them.



Display Data in a GUI Table



To display the data in a tabular form, you need to use JTable.
Download the code. You have the following classes

- Class Employee : represents an employee (empno, name, sal)
- Class DAO: connect to the databse, it contains two methods :
  - search which returns array list of all employees who has the input name.
  - getAll which returns all employees in the table.
- Class EmployeeTableModel which specifies how the data will be displayed on JTabel.
- Class searchGUI which is GUI class.