# Ziegfreid Morissey Flameño

# W2D2 Homework

# Question 1

Table emp, order by ename increasing，which one or more are correct?

A.select ename , salary from emp   order by ename asc

B.select ename , salary from emp   order by ename desc

C.select ename , salary from emp   order by salary asc

D.select ename , salary from emp   order by ename  ;

# Question 2

Table EMPLOYEES

EMPLOYEE\_ID        NUMBER          Primary Key

 FIRST\_NAME             VARCHAR2(25)

 LAST\_NAME              VARCHAR2(25)

Which one insert one record to the table correctly?

A. INSERT INTO employees(employee\_id) VALUES (1000);

B. INSERT INTO employees (first\_name,last\_name) VALUES (‘John’,’Smith’);

 C. INSERT INTO employees VALUES(1000,’John’,’NULL’);

D. INSERT INTO employees VALUES (‘1000’,’John’);

# Question 3

SELECT name, rank, serialNo FROM employee.

Which one to get the value of first column of rs?

A. rs.getString(0);

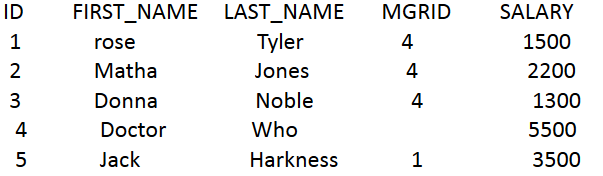
B. rs.getString("name");

C. rs.getString(1);

D. rs.getString(“ename”);

# Question4

At first in database, create a table Employee, id is primary key.



1. Create a table, and insert a record to the table, and print out all the data in the table

(6 Alex Lee 1 4500)

1. Modify the last\_name to “Tate” , and save it to database. And print out all the data in the table.

# Question 5

Why we use Class.forName?

**This is used to get the Class using the name of the class. Particularly, we use it to get the Class instance of the oracle database driver ("oracle.jdbc.driver.OracleDriver").**

# Question 6

What’s the difference between Statement and PreparedStatement?

**Statement executes SQL statements for particular connection and retrieves the results while PreparedStatement allows the programmer to create prepared SQL statements**

**Based on my knowledge, the main advantage of PreparedStatement is that it can be parameterized. Another advantage is that once it is compiled (once), it is faster to use that statement again (especially in queries where the parameters change).**

# Question 7

How to close the resultset, statement and connection after we finished the database operation? Write the codes.

ResultSet:

|  |
| --- |
| pstmt = (PreparedStatement) conn.prepareStatement(sql);  ResultSet rs = pstmt.executeQuery();  **int** col = rs.getMetaData().getColumnCount();  **while** (rs.next()) {  **for** (**int** i = 1; i <= col; i++) {  System.***out***.print(rs.getString(i) + "\t");  **if** ((i == 2) && (rs.getString(i).length() < 8)) {  System.***out***.print("\t");  }  }  System.***out***.println("");  }  **rs.close();** |

Statement:

|  |
| --- |
| Connection conn = *getConn*();  **int** i = 0;  String sql = "update employee set last\_name='" + newLastName + "'where last\_name like '%" + oldLastName + "%'";  PreparedStatement pstmt;  **try** {  pstmt = (PreparedStatement) conn.prepareStatement(sql);  i = pstmt.executeUpdate();  **pstmt.close();**  conn.close();  } **catch** (SQLException e) {  e.printStackTrace();  } |

Connection:

|  |
| --- |
| Connection conn = *getConn*();  **int** i = 0;  String sql = "update employee set last\_name='" + newLastName + "'where last\_name like '%" + oldLastName + "%'";  PreparedStatement pstmt;  **try** {  pstmt = (PreparedStatement) conn.prepareStatement(sql);  i = pstmt.executeUpdate();  **pstmt.close();**  **conn.close();**  } **catch** (SQLException e) {  e.printStackTrace();  } |