Basic JDBC Operations

Create the following table using SQL Command Line:

Column Name	Field Type
UserID	Varchar(200)
Password	Varchar(200)
Name	Varchar(200)
IncorrectAttempts	Number(2)
LockStatus	Number(2)
UserType	Varchar(200)

Insert the following record(s) to the table – (use SQL Command Line)

1	(AB1001, AB1001, Hari, 0, 0, Admin)
2	(TA1002, TA1002, Prasath, 0, 0, Employee)
3	(RS1003, RS1003, Ganesh, 0, 0, Employee)

Establish JDBC Connection and perform the following operations –

Scenario 1:

Create a method **String getUserType(String userID)**. This method should do the following operation.

a) Obtain the Connection Object by calling the **getConnection()** method.
b) Using the Connection Object, retrieve the userType and return it.
c) Create a main method and call the **getUserType** method and print the value obtained.

Scenario 2:

Create a method **String getIncorrectAttempts(String userID)**. This method should do the following operation.

a) Obtain the Connection Object by calling the **getConnection()** method.
b) Using the Connection Object, retrieve the Incorrect Attempts. If the incorrect attempt is 0, return "No Incorrect Attempt", else if the incorrect attempt is 1, return "One Time", else return "Incorrect Attempt Exceeded".
c) Create a main method and call the **getIncorrectAttempts** method and print the value obtained.

Scenario 3:

Create a method **String changeUserType(String userID)**. This method should do the following operation.

a) Obtain the Connection Object by calling the **getConnection()** method.

b) Using the Connection Object, update the user type of the given userID to **Admin**. If more than one row updated return message as **"Update Success"**, else return **"Update Failed"**

c) Create a main method and call the **changeUserType** method and print the value obtained.

Scenario 4:

Create a method **int getLockStatus()**. This method should do the following operation.

a) Obtain the Connection Object by calling the **getConnection()** method.

b) Using the Connection Object, count the total rows with the lock status 0.

c) Create a main method and call the **getLockStatus** method and print the value obtained.

Scenario 5:

Create a method **String changeName(String id, String name)**. This method should do the following operation.

a) Obtain the Connection Object by calling the **getConnection()** method.

b) Using the Connection Object, change the name for the given ID. On successful change return "Success", on-fail return "Failed".

c) Create a main method and call the **changeName** method and print the value obtained.

d) Go to SQL plus and check the correctness of change.

Scenario 6:

Create a method **String changePassword(String password)**. This method should do the following operation.

a) Obtain the Connection Object by calling the **getConnection()** method.

b) Using the Connection Object, change the password to the parameterized password value for all the records with userType "Admin". On success, return "Changed", On Failure return 0.

c) Create a main method and call the **changePassword** method and print the value obtained.

d) Go to SQL plus and check the correctness of change.

Scenario 7:

Create a method **String addUser_1(UserBean bean)**. This method should do the following operation.

a) Create User Bean with following private variables (ID, Password, Name, Incorrect Attempts, Lock Status, User Type) and initialize getters and setters.

b) Obtain the Connection Object by calling the **getConnection()** method.

c) Using the Connection Object, insert a new record to the table. **(Use the parameter of the method to get the values to insert).** On successful insert, return "Success", on-fail return "Fail"

d) Create a main method and set values for the private variables of UserBean, call the **addUser**_1 method and print the value obtained.

e) Go to SQL plus and check the correctness of change.

Scenario 8:

Create a method **String addUser_2(UserBean bean)**. This method should do the following operation.

a) Create User Bean with following private variables (ID, Password, Name, Incorrect Attempts, Lock Status, User Type) and initialize getters and setters.
b) Obtain the Connection Object by calling the **getConnection()** method.
c) Using the Connection Object, insert a new record to the table if and only if the lockstatus is 0. **(Use the parameter of the method to get the values to insert).** On successful insert, return "Success", on-fail return "Fail"
d) Create a main method and set values for the private variables of UserBean, call the **addUser_2** method and print the value obtained.
e) Go to SQL plus and check the correctness of change.

Scenario 9:

Create a method **ArrayList<UserBean> getUsers(String userType)**. This method should do the following operation.

a) Create User Bean with following private variables (ID, Password, Name, Incorrect Attempts, Lock Status, User Type) and initialize getters and setters.
b) Obtain the Connection Object by calling the **getConnection()** method.
c) Using the Connection Object, retrieve the records of specified user-type and store it into array list and return. Note: Use the method parameter and extract the user records of the specified user type.
d) Create a main method call the **getUsers** method and print the values obtained.

Scenario 10:

Create a method **ArrayList<UserBean> storeAllRecords()**. This method should do the following operation.

a) Create User Bean with following private variables (ID, Password, Name, Incorrect Attempts, Lock Status, User Type) and initialize getters and setters.
b) Obtain the Connection Object by calling the **getConnection()** method.
c) Using the Connection Object, retrieve **all** the records and store it into array list.
d) Create a main method call the **storeAllRecords** method and print the values obtained.

Scenario 11:

Create a method **String[] getNames()**. This method should do the following operation.

a) Create User Bean with following private variables (ID, Password, Name, Incorrect Attempts, Lock Status, User Type) and initialize getters and setters.
b) Obtain the Connection Object by calling the **getConnection()** method.
c) Using the Connection Object, retrieve **all names** from the table and store it into string array.
d) Create a main method call the **getNames** method and print the values obtained. *****