MySQL JDBC Transaction

?

In this tutorial, you will learn how to use commit() and rollback() methods of the Connection object to control transaction.

Setting auto-commit mode

When you connect to MySQL databases, the auto-commit mode is set to `true` by default. It means that the changes will be applied to the database once the statement successfully executed. In case you want to control when to commit the transaction, you call the `setAutoCommit()` method of the`Connection` object as follows:

```
Connection conn = DriverManager.getConnection(dbURL,dbUser,dbPassword);
conn.setAutoCommit(false);
```

Once you have set auto-commit mode to `false` , you can call`commit()` or `rollback()` methods of the `Connection` object to commit or rollback the transaction.

Notice that you should always call `setAutoCommit()` method right after you open a connection to the database.

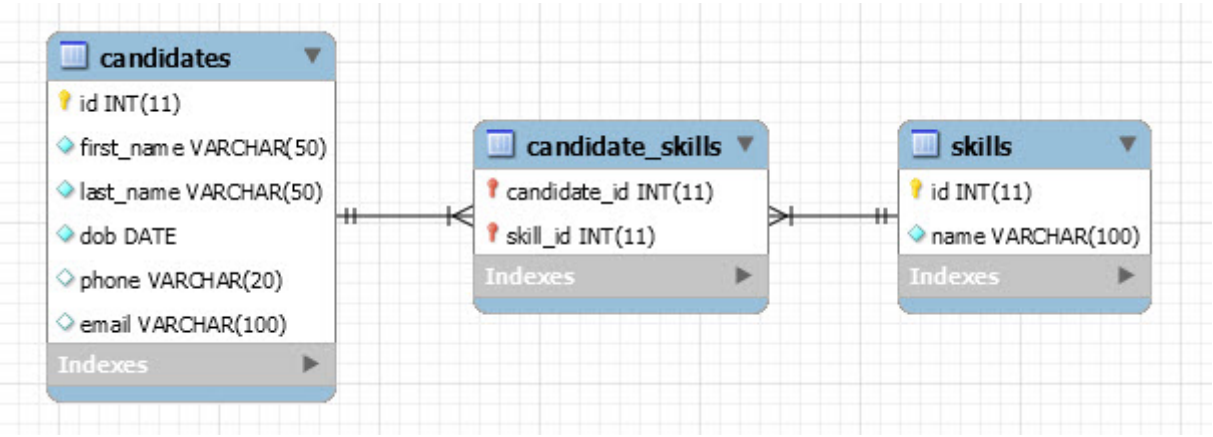
Committing and rolling back a transaction

Once the auto-commit mode is set to `false` , you can commit or rollback the transaction. The flow of using those methods is as follows:

```
try(Connection conn = DriverManager.getConnection(
URL,dbUser,dbPassword);){
   conn.setAutoCommit(false);

   // perform operations such as insert, update,
 delete here
   // ..

   // if everything is OK, commit the transaction
   conn.commit();

} catch(SQLException e) {
   // in case of exception, rollback the transact
ion
   conn.rollback();
}
```

MySQL JDBC transaction example

In this example, we will insert a new record into the`candidates` table and also assign some of skills to the newly inserted candidate.



We will perform both inserting a candidate and assigning skills within one transaction. The steps will be as follows:

1. Insert a record into the `candidates` table and get the inserted ID back.
2. Insert a set of candidate ID and Skill ID into the`candidate_skills` table.
3. If all above operations are successfully, commit the transaction, otherwise roll it back.

The following is the complete example of using JDBC transaction.

```
1  package org.mysqltutorial;
2
3  import java.sql.Connection;
4  import java.sql.PreparedStatement;
5  import java.sql.SQLException;
6  import java.sql.Date;
7  import java.sql.ResultSet;
8  import java.sql.Statement;
9
10 /**
11  *
12  * @author mysqltutorial.org
13  */
14 public class Main {
15   /**
16    * Insert and assign skills to a specific cand
17 idates
18    * @param firstName
19    * @param lastName
20    * @param dob
21    * @param email
22    * @param phone
23    * @param skills
24    */
     public static void addCandidate(String first
25 Name,String lastName,Date dob,
                                       String
26 email, String phone, int[] skills) {
27
28         Connection conn = null;
29
30         // for insert a new candidate
31         PreparedStatement pstmt = null;
32
33         // for assign skills to candidate
34
```

```
        PreparedStatement pstmtAssignment =
null;

        // for getting candidate id
        ResultSet rs = null;

        try {
            conn =
MySQLJDBCUtil.getConnection();
            // set auto commit to false
            conn.setAutoCommit(false);
            //
            // Insert candidate
            //
            String sqlInsert = "INSERT INTO cand
idates(first_name,last_name,dob,phone,email) "
                               + "VALUES
(?,?,?,?,?)";

            pstmt = conn.prepareStatement(sqlIns
ert,Statement.RETURN_GENERATED_KEYS);

            pstmt.setString(1, firstName);
            pstmt.setString(2, lastName);
            pstmt.setDate(3, dob);
            pstmt.setString(4, phone);
            pstmt.setString(5, email);

            int rowAffected = pstmt.executeUpdat
e();

             // get candidate id
            rs = pstmt.getGeneratedKeys();
            int candidateId = 0;
            if(rs.next())
                candidateId = rs.getInt(1);
            //
            // in case the insert operation succ
esses, assign skills to candidate
```

```
70              //
71              if(rowAffected == 1)
72              {
73                  // assign skills to candidates
74                  String sqlPivot = "INSERT INTO c
75  andidate_skills(candidate_id,skill_id) "
76                                      +
77  "VALUES(?,?)";
78
79                  pstmtAssignment = conn.prepareSt
80  atement(sqlPivot);
81                  for(int skillId : skills) {
82
83                      pstmtAssignment.setInt(1, ca
84  ndidateId);
85                      pstmtAssignment.setInt(2, sk
86  illId);
87
88                      pstmtAssignment.executeUpdat
89  e();
90                  }
91                  conn.commit();
92              } else {
93                  conn.rollback();
94              }
95          } catch (SQLException ex) {
96              // roll back the transaction
97              try{
98                  if(conn != null)
99                      conn.rollback();
100             }catch(SQLException e){
101                 System.out.println(e.getMessage()
102 ;
103             }
104
105
106             System.out.println(ex.getMessage());
107         } finally {
108             try {
```

```
                if(rs != null)  rs.close();
                if(pstmt != null) pstmt.close();
                if(pstmtAssignment != null) pstm
tAssignment.close();
                if(conn != null) conn.close();

            } catch (SQLException e) {
                System.out.println(e.getMessage()
;
            }
        }
    }

    public static void main(String[] args) {
       // insert and assign skills
        int[] skills = {1,2,3};
        addCandidate("John", "Doe", Date.valueOf(
990-01-04"),
                        "john.d@yahoo.com", "(40
8) 898-5641", skills);
    }
}
```

Let's check the table `candidates` table before we run the program.

```
SELECT * FROM candidates
ORDER BY id DESC;
```

id	first_name	last_name	dob	phone	email
121	Tony	Snowden	1980-05-01	(649) 555-5500	tony.s@outlook.com
120	Valarie	Franco	1980-04-30	(617) 555-2555	valarie.f@outlook.com
119	Thomas	Smith	1980-04-29	(171) 555-7555	thomas.s@outlook.com
118	Sue	Taylor	1980-04-28	(415) 555-4312	sue.t@outlook.com
117	Rosa	Salazar	1980-04-27	(215) 555-9857	rosa.s@outlook.com
116	José Pedro	Roel	1980-04-26	(955) 558-2820	josé pedro.r@outlook.com
115	Alexander	Semenov	1980-04-24	(781) 229-3052	alexander.s@outlook.com
114	Hanna	Moos	1980-04-23	(621) 085-5522	hanna.m@outlook.com

Now, we run the program.