|  |
| --- |
| package com.company.customersdata; |
|  |  |
|  | import java.sql.\*; |
|  | import java.util.Date; |
|  |  |
|  | public class Datasource { |
|  |  |
|  | static final String DB\_NAME = "customers"; |
|  | static final String SERVER = "localhost:3306"; |
|  | static final String CONNECTION\_STRING = "jdbc:mysql://" + SERVER + "/?useSSL=false&serverTimezone=UTC"; |
|  | static final String USER = "student"; |
|  | static final String PASSWORD = "student"; |
|  |  |
|  | static final String TABLE\_BRANCHES = "branches"; |
|  | static final String COLUMN\_BRANCH\_ID = "\_id"; |
|  | static final String COLUMN\_BRANCH\_NAME = "name"; |
|  |  |
|  | static final String TABLE\_CUSTOMERS = "customers"; |
|  | static final String COLUMN\_CUSTOMER\_ID = "\_id"; |
|  | static final String COLUMN\_CUSTOMER\_NAME = "name"; |
|  | static final String COLUMN\_CUSTOMER\_BRANCH\_ID = "branch"; |
|  |  |
|  |  |
|  | static final String TABLE\_TRANSACTIONS = "transactions"; |
|  | static final String COLUMN\_TRANSACTION\_DATE = "\_date"; |
|  | static final String COLUMN\_TRANSACTION\_VALUE = "\_value"; |
|  | static final String COLUMN\_TRANSACTION\_CUSTOMER\_ID = "customer"; |
|  |  |
|  | private Connection conn; |
|  |  |
|  | // opening connection with server, creating database if's not created and tables |
|  |  |
|  | public boolean open() { |
|  | try { |
|  | conn = DriverManager.getConnection(CONNECTION\_STRING, USER, PASSWORD); |
|  |  |
|  | String createDB = "CREATE DATABASE IF NOT EXISTS " + DB\_NAME; |
|  | String createBranches = "CREATE TABLE IF NOT EXISTS " + TABLE\_BRANCHES + " (" |
|  | + COLUMN\_BRANCH\_ID + " int NOT NULL AUTO\_INCREMENT, " |
|  | + COLUMN\_BRANCH\_NAME + " text, " |
|  | + "PRIMARY KEY(" + COLUMN\_BRANCH\_ID + ")" |
|  | + ")"; |
|  | String createCustomers = "CREATE TABLE IF NOT EXISTS " + TABLE\_CUSTOMERS + " (" |
|  | + COLUMN\_CUSTOMER\_ID + " int NOT NULL AUTO\_INCREMENT, " |
|  | + COLUMN\_CUSTOMER\_NAME + " text, " |
|  | + COLUMN\_CUSTOMER\_BRANCH\_ID + " int NOT NULL, " |
|  | + "PRIMARY KEY(" + COLUMN\_CUSTOMER\_ID + ")" |
|  | + ")"; |
|  | String createTransactions = "CREATE TABLE IF NOT EXISTS " + TABLE\_TRANSACTIONS + " (" |
|  | + COLUMN\_TRANSACTION\_DATE + " TIMESTAMP, " |
|  | + COLUMN\_TRANSACTION\_VALUE + " decimal(10,2), " |
|  | + COLUMN\_TRANSACTION\_CUSTOMER\_ID + " int NOT NULL, " |
|  | + "PRIMARY KEY(" + COLUMN\_TRANSACTION\_DATE + ")" |
|  | + ")"; |
|  | conn.setCatalog(DB\_NAME); |
|  | Statement statement = conn.createStatement(); |
|  | statement.executeUpdate(createDB); |
|  | statement.executeUpdate(createBranches); |
|  | statement.executeUpdate(createCustomers); |
|  | statement.executeUpdate(createTransactions); |
|  | statement.close(); |
|  |  |
|  |  |
|  | System.out.println("Connection successfully opened."); |
|  | return true; |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR: " + e.getMessage()); |
|  | return false; |
|  | } |
|  | } |
|  |  |
|  | public void close() { |
|  | try { |
|  | if (conn != null) { |
|  | conn.close(); |
|  | } |
|  | } catch (SQLException e) { |
|  | System.out.println("Couldn't close connection: " + e.getMessage()); |
|  | } |
|  | } |
|  |  |
|  | public void addBranch(String name) { |
|  | if (selectBranchId(name)==0){ |
|  | //if (!checkBranch(name)) { |
|  | String insertBranch = "INSERT INTO " + TABLE\_BRANCHES + " (" + COLUMN\_BRANCH\_NAME + ") " + "VALUES ('" + name + "')"; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | statement.executeUpdate(insertBranch); |
|  | statement.close(); |
|  | System.out.println("Branch " + name + " is" + " successfully added."); |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in addBranch method: " + e.getMessage()); |
|  | } |
|  | } else |
|  | System.out.println("Branch already exists in database."); |
|  | } |
|  |  |
|  | // public boolean checkBranch(String name) { |
|  | // String selectBranch = "SELECT " + COLUMN\_BRANCH\_NAME + " FROM " + TABLE\_BRANCHES + " WHERE " + COLUMN\_BRANCH\_NAME + "='" + name + "'"; |
|  | // boolean checker = true; |
|  | // try { |
|  | // Statement statement = conn.createStatement(); |
|  | // ResultSet resultSet = statement.executeQuery(selectBranch); |
|  | // if (!resultSet.next()) |
|  | // checker = false; |
|  | // else |
|  | // checker = true; |
|  | // resultSet.close(); |
|  | // statement.close(); |
|  | // } catch (SQLException e) { |
|  | // System.out.println("ERROR in checkBranch method: " + e.getMessage()); |
|  | // } |
|  | // return checker; |
|  | // } |
|  |  |
|  | public int selectBranchId(String name) { |
|  | String selectBranchId = "SELECT " + COLUMN\_BRANCH\_ID + " FROM " + TABLE\_BRANCHES + " WHERE " + COLUMN\_BRANCH\_NAME + "='" + name + "'"; |
|  | int id = 0; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | ResultSet resultSet = statement.executeQuery(selectBranchId); |
|  | while (resultSet.next()) { |
|  | id = resultSet.getInt(COLUMN\_BRANCH\_ID); |
|  | } |
|  | resultSet.close(); |
|  | statement.close(); |
|  | return id; |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in selectBranchId method: " + e.getMessage()); |
|  | return 0; |
|  | } |
|  | } |
|  |  |
|  | public void selectBranches(){ |
|  | String selectBranches = "SELECT \* FROM " + TABLE\_BRANCHES; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | ResultSet resultSet = statement.executeQuery(selectBranches); |
|  | System.out.println("ID\tBRANCH\_NAME"); |
|  | while (resultSet.next()) { |
|  | System.out.println(resultSet.getInt(COLUMN\_BRANCH\_ID)+"\t"+ |
|  | resultSet.getString(COLUMN\_BRANCH\_NAME)); |
|  | } |
|  | resultSet.close(); |
|  | statement.close(); |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in selectBranches method: " + e.getMessage()); |
|  | } |
|  | } |
|  |  |
|  | public void addCustomer(int branchId, String name, double initial) { |
|  | String insertCustomer = "INSERT INTO " + TABLE\_CUSTOMERS + " (" |
|  | + COLUMN\_CUSTOMER\_NAME + " , " |
|  | + COLUMN\_CUSTOMER\_BRANCH\_ID + ") " |
|  | + "VALUES ('" + name + "' , " + branchId + ")"; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | statement.executeUpdate(insertCustomer); |
|  | statement.close(); |
|  | System.out.println("Customer " + name + " is" + " successfully added."); |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in addCustomer method: " + e.getMessage()); |
|  | } |
|  | addTransaction(1,initial); |
|  |  |
|  | } |
|  |  |
|  | // private boolean checkCustomer(String name, int branchId) { |
|  | // String selectCustomer = "SELECT " + COLUMN\_CUSTOMER\_NAME + " FROM " + TABLE\_CUSTOMERS + " WHERE " + COLUMN\_CUSTOMER\_NAME + "='" + name + "'" |
|  | // + " and " + COLUMN\_CUSTOMER\_BRANCH\_ID + "=" + branchId; |
|  | // boolean checker = true; |
|  | // try { |
|  | // Statement statement = conn.createStatement(); |
|  | // ResultSet resultSet = statement.executeQuery(selectCustomer); |
|  | // if (!resultSet.next()) |
|  | // checker = false; |
|  | // else |
|  | // checker = true; |
|  | // resultSet.close(); |
|  | // statement.close(); |
|  | // } catch (SQLException e) { |
|  | // System.out.println("ERROR in checkCustomer method: " + e.getMessage()); |
|  | // } |
|  | // return checker; |
|  | // } |
|  |  |
|  | public int selectCustomerId(String name, int branchId) { |
|  | String selectCustomerId = "SELECT " + COLUMN\_CUSTOMER\_ID + " FROM " |
|  | + TABLE\_CUSTOMERS |
|  | + " WHERE " |
|  | + COLUMN\_CUSTOMER\_NAME + "='" + name + "'" |
|  | + " and " + COLUMN\_CUSTOMER\_BRANCH\_ID + "=" + branchId; |
|  | int id = 0; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | ResultSet resultSet = statement.executeQuery(selectCustomerId); |
|  | while (resultSet.next()) { |
|  | id = resultSet.getInt(COLUMN\_CUSTOMER\_ID); |
|  | } |
|  | resultSet.close(); |
|  | statement.close(); |
|  | return id; |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in selectCustomer method: " + e.getMessage()); |
|  | return 0; |
|  | } |
|  | } |
|  |  |
|  | public void selectCustomers(int branchId){ |
|  | String selectCustomers = "SELECT \* FROM " + TABLE\_CUSTOMERS + " WHERE " + COLUMN\_CUSTOMER\_BRANCH\_ID + "=" + branchId; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | ResultSet resultSet = statement.executeQuery(selectCustomers); |
|  | System.out.println("ID\tCUSTOMER\_NAME\tBRANCH\_ID"); |
|  | while (resultSet.next()) { |
|  | System.out.println(resultSet.getInt(COLUMN\_CUSTOMER\_ID)+"\t"+ |
|  | resultSet.getString(COLUMN\_CUSTOMER\_NAME)+"\t"+ |
|  | resultSet.getInt(COLUMN\_CUSTOMER\_BRANCH\_ID)); |
|  | } |
|  | resultSet.close(); |
|  | statement.close(); |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in selectCustomers method: " + e.getMessage()); |
|  | } |
|  | } |
|  |  |
|  | public void addTransaction(int customerId, double value) { |
|  | java.util.Date now = new Date(); |
|  | java.sql.Timestamp date = new java.sql.Timestamp(now.getTime()); |
|  | String insertTransaction = "INSERT INTO " + TABLE\_TRANSACTIONS + " (" |
|  | + COLUMN\_TRANSACTION\_DATE + " , " |
|  | + COLUMN\_TRANSACTION\_VALUE + " , " |
|  | + COLUMN\_TRANSACTION\_CUSTOMER\_ID + ") " |
|  | + "VALUES ('" |
|  | + date + "' , '" |
|  | + value + "' , " |
|  | + customerId + ")"; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | statement.executeUpdate(insertTransaction); |
|  | statement.close(); |
|  | System.out.println("Transaction successfully added."); |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in addCustomer method: " + e.getMessage()); |
|  | } |
|  | } |
|  |  |
|  | public void selectTransactions(int customerId){ |
|  | String selectTransactions = "SELECT \* FROM " + TABLE\_TRANSACTIONS + " WHERE " + COLUMN\_TRANSACTION\_CUSTOMER\_ID + "=" + customerId; |
|  | try { |
|  | Statement statement = conn.createStatement(); |
|  | ResultSet resultSet = statement.executeQuery(selectTransactions); |
|  | System.out.println("DATE\tVALUE\tCUSTOMER\_ID"); |
|  | while (resultSet.next()) { |
|  | System.out.println(resultSet.getTimestamp(COLUMN\_TRANSACTION\_DATE)+"\t"+ |
|  | resultSet.getString(COLUMN\_TRANSACTION\_VALUE)+"\t"+ |
|  | resultSet.getInt(COLUMN\_TRANSACTION\_CUSTOMER\_ID)); |
|  | } |
|  | resultSet.close(); |
|  | statement.close(); |
|  | } catch (SQLException e) { |
|  | System.out.println("ERROR in selectCustomers method: " + e.getMessage()); |
|  | } |
|  | } |
|  |  |
|  | } |