DataBean

1. Tạo Class DataBean.class mục đích là xây dựng một "bean" tối đa khả năng sử dụng lại trên toàn bộ các project khác nhau:

```java
package bean;
/**
 *
 * @author tuann
 */
import java.util.*;
import java.sql.*;

public class DataBean {

    private Connection connection = null;
    private ResultSet resultset = null;

    // Khai bao ve cac kieu cau truy van co the su dung
    private Statement statement = null;
    private PreparedStatement prepareStatement = null;
    private CallableStatement storeProc = null;

    // Cac thong so ve ket qua tra ve
    private int resultsettype = 0;
    private int resultsetconcurrency = 0;

    // Cac thong so can de ket noi co so du lieu de truy van
    private String driver = null;
    private String url = null;
    private String username = null;
    private String password = null;
    private Properties properties = null;

    private String database;       // Ten co so du lieu lam viec
    private String table;          // Ten bang can thuc hien truy van
    private Vector<String> fields; // Ten cac truong trong bang
```

```java
        // So hieu loi va cau thong bao loi
        private int error = 0;
        private String errmsg = null;
        private boolean preparestatement = false;

        /**
         * Phuong thuc xem loi khi thuc hien truy van
         */
        public int getError() {
            return error;
        }

        /**
         * Phuong thuc thiet lap ten nguoi dung
         *
         * @param usr Ten nguoi dung moi can thay doi
         */
        public void setUsername(String usr) {
            username = usr;
        }

        /**
         * Phuong thuc lay ten cua nguoi dung
         */
        public String getUsername() {
            return username;
        }

        /**
         * Phuong thuc thiet lap mat khau truy cap CSDL
         *
         * @param psw Mat khau truy cap
         */
        public void setPassword(String psw) {
            password = psw;
        }
```

```java
/**
 * Phuong thuc lay mat khau truy cap
 *
 * @return
 */
public String getPassword() {
    return password;
}

/**
 * Phuong thuc thiet lap driver de truy cap CSDL
 *
 * @param drv Ten driver truy cap CSDL
 */
public void setDriver(String drv) {
    driver = drv;
}

/**
 * Phuong thuc tra ve ten cua driver truy cap CSDL
 */
public String getDriver() {
    return driver;
}

/**
 * Phuong thuc thiet lap duong dan toi CSDL
 *
 * @param Url Duong dan toi bang CSDL can truy van
 */
public void setUrl(String Url) {
    url = Url;
}
```

```java
        /**
         * Phuong thuc tra ve duong dan toi bang CSDL can truy van
         */
        public String getUrl() {
            return url;
        }


        /**
         * Phuong thuc thiet lap cac thuoc tinh trong tim kiem
         */
        public void setProperties(Properties pr) {
            properties = pr;
        }

        public Properties getProperties() {
            return properties;
        }


        /**
         * Phuong thuc thiet lap ket noi toi CSDL
         *
         * @param con Doi tuong ket noi toi CSDl
         */
        public void setConnection(Connection con) {
            // Neu chua co driver thi gan loi so 1
            if (driver == null) {
                error = 1;
                return;
            }

            // Neu chua co duong dan toi CSDL thi gan loi so 2
            if (url == null) {
                error = 2;
                return;
            }
```

```java
        try {
            Class.forName(driver);   // Nap driver
            // Neu bien thuoc tinh chua co thi truy cap theo ten va mat khau
            if (properties == null) {
                if ((username == null) && (password == null)) {
                    con = DriverManager.getConnection(url);
                } else {
                    con = DriverManager.getConnection(url, username, password);
                }
            } else {
                con = DriverManager.getConnection(url, properties);
            }
            connection = con;

        } // Khi gap loi thi nhan thong diep loi tu he thong
        catch (Exception e) {
            errmsg = e.getMessage();
            error = -1;
            System.out.println("Co loi khi tao ket noi");
            return;
        }
        error = 0;
    }

    /**
     * Phuong thuc lay ket noi den CSDL
     */
    public Connection getConnection() {
        if (connection == null) {
            error = 3;
        }
        return connection;
    }
```

```java
        /**
         * Thiet lap kieu cua ket qua tra ve
         *
         * @param type Kieu cua ket qua tra ve
         */
        public void setResultsettype(int type) {
            resultsettype = type;
        }

        /**
         * Phuong thuc lay ket qua truy van CSDL
         */
        public int getResultsettype() {
            return resultsettype;
        }

        /**
         * Phuong thuc thiet lap bo ket qua dong bo
         *
         * @param type Kieu ket qua dong bo
         */
        public void setResultsetconcurrency(int type) {
            resultsetconcurrency = type;
        }

        public int getResultsetconcurrency() {
            return resultsetconcurrency;
        }

        /**
         * Phuong thuc lay ve doi tuong quan ly cac cau truy van thong thuong
         */
        public Statement getStatement() {
            try {
                if ((resultsettype == 0) && (resultsetconcurrency == 0)) {
                    statement = connection.createStatement();   // Tao doi tuong truy van thong thuong
                } else {
                    statement = connection.createStatement(resultsettype, resultsetconcurrency);
                }
            } catch (SQLException e) {
                errmsg = "Error : " + e.getSQLState() + "--" + e.getMessage();
                error = -1;
            }
            return statement;
        }

        /**
         * Phuong thuc tra ve doi tuong quan ly cau truy van co bien
         */
        public PreparedStatement getPrepareStatement(String sql) {
            try {
                if ((resultsettype == 0) && (resultsetconcurrency == 0)) {
                    prepareStatement = connection.prepareStatement(sql);
                } else {
                    prepareStatement = connection.prepareStatement(sql, resultsettype, resultsetconcurrency);
                }
            } catch (SQLException e) {
                errmsg = "Error : " + e.getSQLState() + "--" + e.getMessage();
                error = -1;
            }
            return prepareStatement;
        }

        /**
         * Phuong thuc tra ve cau truy van tao cac ham luu
         */
```

```java
    public CallableStatement getStoreProc(String sql) {
        try {
            if ((resultsettype == 0) && (resultsetconcurrency == 0)) {
                storeProc = connection.prepareCall(sql);
            } else {
                storeProc = connection.prepareCall(sql, resultsettype, resultsetconcurrency);
            }
        } catch (SQLException e) {
            errmsg = "Error : " + e.getSQLState() + "--" + e.getMessage();
            error = -1;
        }
        return storeProc;
    }

    public void setResultset(ResultSet rs) {
        resultset = rs;
    }

    // Phuong thuc tra ve doi tuong ket qua truy van
    public ResultSet getResultset() {
        return resultset;
    }

    // Phuong thuc tra ve cac thong bao loi
    public String getErrmsg() {
        switch (error) {
            case 0:
                errmsg = "No Error !!!";
                break;
            case 1:
                errmsg = "Driver invalid !!!";
                break;
            case 2:
                errmsg = "URL invalid !!!";
                break;
            case 3:
```

```java
                    errmsg = "Connection is not currently established !!!";
                    break;
            }
            return errmsg;
        }

        /**
         * Phuong thuc huy ket noi va cac ket qua truy van
         */
        public void destroy() {
            // Thu dong cac ket noi, bang truy van hay cac cau truy van neu no dang mo
            try {
                if (resultset != null) {
                    resultset.close();
                }
                if (statement != null) {
                    statement.close();
                }
                if (prepareStatement != null) {
                    statement.close();
                }
                if (connection != null) {
                    connection.close();
                }
            } catch (SQLException e) {
                errmsg = "Error : " + e.getSQLState() + "--" + e.getMessage();
                error = -1;
            }
        }

        /**
         * Phuong thuc thiet lap co so du lieu de lam viec
         *
         * @param dtb Ten co so du lieu hoat dong
         */
```

```java
    public void setDatabase(String dtb) {
        database = dtb;
    }

    /**
     * Phuong thuc tra ve ten co so du lieu
     *
     * @return
     */
    public String getDatabase() {
        return database;
    }

    /**
     * Phuong thuc thiet lap bang du lieu lam viec
     *
     * @param tb Ten bang lam viec
     */
    public void setTable(String tb) {
        table = tb;
    }

    /**
     * Tra ve ten cua bang dang lam viec
     */
    public String getTable() {
        return table;
    }

    /**
     * Thiet lap cac truong trong bang can xu ly
     *
     * @param fs Danh sach ten cac truong trong bang. Mac dinh ten truong dau
     * tien se la khoa chinh
```

```java
         */
        public void setFields(Vector<String> fs) {
            fields = fs;
        }
        /**
         * Phuong thuc chen them 1 khoi du lieu moi vao bang
         *
         * @param pField
         */
        public void insert(Properties pField) {
            String queryString = "INSERT INTO " + table + "(";
            int t = 0;
            // Liet ke ten cac cot theo tu tu gia tri them vao
            for (int i = 0; i < fields.size(); i++) {
                String s = pField.getProperty(fields.get(i));
                if (s != null) {
                    if (t != 0) {
                        queryString += ",";
                    }
                    queryString += fields.get(i);
                    t++;
                }
            }
            queryString += ") VALUES (";
            t = 0;
            // Dien cac gia tri can them vao dung vi tri tuong ung
            for (int i = 0; i < fields.size(); i++) {
                String s = pField.getProperty(fields.get(i));
                if (s != null) {
                    if (t != 0) {
                        queryString += ",";
                    }
                    queryString += "'" + pField.getProperty(fields.get(i)) + "'";
                    t++;
                }
            }
```

```java
            queryString += ");";

            try {
                System.out.println(queryString);
                if (statement == null) {
                    System.out.println("Statement == NULL");
                }
                statement.executeUpdate(queryString);
            } catch (SQLException sqlException) {
                System.out.println(sqlException);
            }
        }

        /**
         * Phuong thuc thay doi 1 khoi du lieu moi trong bang
         *
         * @param pField
         */
        public void update(Properties pField) {
            String queryString = "UPDATE " + table + " SET ";
            int t = 0;
            // Liet ke ten cac cot theo tu tu gia tri can cap nhat
            for (int i = 0; i < fields.size(); i++) {
                String s = pField.getProperty(fields.get(i));

                if (s != null) {
                    if (t != 0) {
                        queryString += ",";
                    }
                    queryString += fields.get(i) + "='" + pField.getProperty(fields.get(i)) + "'";
                    t++;
                }
            }
            // Cap nhat dua vao truong khoa chinh
            queryString += " WHERE " + fields.get(0) + "='" + pField.getProperty(fields.get(0)) + "'";

            try {
                statement.executeUpdate(queryString);
            } catch (SQLException sqlException) {
                System.out.println(sqlException);
            }
        }

        /**
         * Phuong thuc xoa 1 khoi du lieu ra khoi bang
         */
        public void delete(Properties pField) {
            String queryString = "DELETE FROM " + table + " WHERE ";
            int t = 0;
            // Liet ke ten cac cot theo tu tu gia tri can cap nhat
            for (int i = 0; i < fields.size(); i++) {
                String s = pField.getProperty(fields.get(i));
                System.out.println(s);

                if (s != null) {
                    if (t != 0) {
                        queryString += " AND ";
                    }
                    queryString += fields.get(i) + "='" + pField.getProperty(fields.get(i)) + "'";
                    t++;
                }
            }

            try {
                System.out.println(queryString);
                statement.executeUpdate(queryString);
            } catch (SQLException sqlException) {
                System.out.println(sqlException);
            }
        }
    }
```