

```
package fi.paatti.paattidatabasutils.dbsevice.querydelegates;

import com.vaadin.data.util.sqlcontainer.RowItem;
import com.vaadin.data.util.sqlcontainer.query.generator.StatementHelper;
import fi.paatti.paattidatabasutils.names.PaattiColumnNames;
import fi.paatti.paattidatabasutils.names.PaattiTableNames;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;

/**
 * Get the users groups and roles and researches that belong to the groups.
 */
*
* @author Lauri Satokangas, lauri.n.satokangas@student.jyu.fi
* @since
*/
public class UserGroupRolesAndResearchDelegate extends PaattiQueryDelegate {

    private static final long serialVersionUID = 1L;
    private final int userID;

    /**
     * Constructor.
     */
    * @param userID The ID of the user.
    */
    public UserGroupRolesAndResearchDelegate(int userID) {
        super(PaattiTableNames.USERGROUP, PaattiColumnNames.USERGROUP_usergroupID,
            PaattiColumnNames.getColumnNamesInArray(PaattiTableNames.USERGROUP));
        this.userID = userID;
    }

    /**
     * {@inheritDoc}
     */
    @Override
    public int storeRow(Connection conn, RowItem row) throws SQLException {
        throw new UnsupportedOperationException("Not supported yet.");
    }

    /**
     * {@inheritDoc}
     */
    @Override
```

```
protected void setRowValues(PreparedStatement statement, RowItem row) throws SQLException {
    throw new UnsupportedOperationException("Not supported yet.");
}

/**
 * Returns a new StatementHelper containing the SELECT query statement.
 */
* @param offset the starting row.
* @param limit the number of rows to get.
* @return The StatementHelper containing the SELECT query.
* @throws UnsupportedOperationException
 */
public StatementHelper getQueryStatement(int offset, int limit) throws UnsupportedOperationException {
    StatementHelper queryHelper = new StatementHelper();
    StringBuilder querySb = new StringBuilder("SELECT ");

    querySb.append(
        getArrayAsStringWithAddedString(
            new String[] {
                PaattiColumnNames.USERGROUP_usergroupId, PaattiColumnNames.USERGROUP_name,
                PaattiColumnNames.RESEARCH_researchchID, PaattiColumnNames.RESEARCH_name, PaattiColumnNames.ROLE_roleID,
                PaattiColumnNames.ROLE_name}, null, ", ", true));
    querySb.append(" FROM ").append(PaattiTableNames.USERGROUP).append(
        " LEFT OUTER JOIN " + PaattiTableNames.ROLE + " ON " + PaattiColumnNames.USERGROUP_ROLE_roleID + " = "
        + PaattiColumnNames.ROLE_roleID);
    querySb.append(
        " LEFT OUTER JOIN " + PaattiTableNames.RESEARCH + " ON " + PaattiColumnNames.USERGROUP_RESEARCH_researchchID
        + " = " + PaattiColumnNames.RESEARCH_researchchID);
    querySb.append(" WHERE " + PaattiColumnNames.USERGROUP_usergroupId + " IN (SELECT ");
    querySb.append(PaattiColumnNames.BELONGS_USERGROUP_usergroupId + " FROM " + PaattiTableNames.BELONGS + " WHERE " +
        PaattiColumnNames.BELONGS_USER_userID + " = ").append(userID).append(
        ")");
    queryHelper.setQueryString(querySb.toString());
    return queryHelper;
}
}
```