Item 182

git_comments:

1. * * A variant of {@link #executeInsert(String, java.util.List)} /** * A variant of {@link #executeInsert(String, List, List)} * useful when providing the named parameters as named arguments. * This variant allows you to receive the values of any auto-generated columns, * such as an autoincrement ID field (or fields) when you know the column name(s) of the ID field(s). * * @param params a map containing the named parameters * @param sql The SQL statement to execute * @param keyColumnNames a list of column names indicating the columns that should be returned from the * inserted row or rows (some drivers may be case sensitive, e.g. may require uppercase names) * @return A list of the auto-generated row results for each inserted row (typically auto-generated keys) * @throws SQLException if a database access error occurs * @see Connection#prepareStatement(String, String[]) * @since 2.3.2
2. * * Executes the given SQL statement (typically an INSERT statement). * Use this variant when you want to receive the values of any auto-generated columns, * such as an autoincrement ID field (or fields) and you know the column name(s) of the ID field(s). * <p> * Resource handling is performed automatically where appropriate. * * @param gstring a GString containing the SQL query with embedded params * @param keyColumnNames a list of column names indicating the columns that should be returned from the * inserted row or rows (some drivers may be case sensitive, e.g. may require uppercase names) * @return A list of the auto-generated row results for each inserted row (typically auto-generated keys) * @throws SQLException if a database access error occurs * @see Connection#prepareStatement(String, String[]) * @see #expand(Object) * @since 2.3.2

git_commits:

1. **summary:** GROOVY-5996: Sql#executeInsert does not return list of auto-generated keys
   message: GROOVY-5996: Sql#executeInsert does not return list of auto-generated keys

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH

2. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
3. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
   label: documentation
4. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
5. **summary:** Sql#executeInsert does not return list of auto-generated keys

description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: " ,first_name: " ,middle_initial: " ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: " ,sex: " ,ssn: " ,claim: " ,created_by: " ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: " ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH

6. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: " ,first_name: " ,middle_initial: " ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: " ,sex: " ,ssn: " ,claim: " ,created_by: " ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: " ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
7. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: " ,first_name: " ,middle_initial: " ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: " ,sex: " ,ssn: " ,claim: " ,created_by: " ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: " ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
   label: documentation
8. **summary:** Sql#executeInsert does not return list of auto-generated keys

description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH

9. **summary:** Sql#executeInsert does not return list of auto-generated keys
   description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
10. **summary:** Sql#executeInsert does not return list of auto-generated keys
    description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
11. **summary:** Sql#executeInsert does not return list of auto-generated keys
    description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated

ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH

12. **summary:** Sql#executeInsert does not return list of auto-generated keys
    description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
13. **summary:** Sql#executeInsert does not return list of auto-generated keys
    description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH
14. **summary:** Sql#executeInsert does not return list of auto-generated keys
    description: When using {{Sql#executeInsert}}, the documentation states the return value should be a list of auto-generated column values. In my script, the return value does not contain the correct generated ID. This might be an Oracle problem. http://groovy.codehaus.org/gapi/groovy/sql/Sql.html#executeInsert%28java.lang.String%29 The

suggested approach to getting the generated ID is something like as follows. {code:javascript} def result = database.executeInsert(...) def generated_id = result[0][0] {code} My insert code as follows. {code:javascript} def result = database.executeInsert("""\ insert into buy_in (rid ,last_name ,first_name ,mi ,dob ,prog_co ,sex ,ssn ,ss_claim_no ,created_by ,created_on ,modified_by ,modified_on) values (:id ,:last_name ,:first_name ,:middle_initial ,to_date(:birthday, 'yyyy-mm-dd hh24:mi:ss') ,:program_county ,:sex ,:ssn ,:claim ,:created_by ,to_date(:created_on, 'yyyy-mm-dd hh24:mi:ss') ,:modified_by ,to_date(:modified_on, 'yyyy-mm-dd hh24:mi:ss')) """ ,[id: 'foo' ,last_name: '' ,first_name: '' ,middle_initial: '' ,birthday: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,program_county: '' ,sex: '' ,ssn: '' ,claim: '' ,created_by: '' ,created_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date()) ,modified_by: '' ,modified_on: new SimpleDateFormat('yyyy-MM-dd HH:mm:ss').format(new Date())]) {code} My return value is as follows. {code:javascript} result.dump() => <java.util.ArrayList@3811f796 elementData=[[oracle.sql.ROWID@3811f758], null, null, null, null, null, null, null, null, null] size=1 modCount=1> to_json(result) => [[{ "data": [65, 65, 67, 77, 65, 78, 65, 65, 76, 65, 65, 76, 101, 115, 57, 65, 65, 70] }]] {code} Any ideas? Thanks. -AH

jira_issues_comments:

1. DDL as follows. The {{oid}} column is the primary key, and is auto-generated by the trigger & sequence. Oracle doesn't have a feature for auto-increment columns. {code:sql} create table buy_in (oid number(12,0) ,rid varchar2(10) ,last_name varchar2(24) ,first_name varchar2(15) ,mi varchar2(1) ,dob date ,prog_co varchar2(3) ,sex varchar2(1) ,ssn varchar2(9) ,ss_claim_no varchar2(12) ,created_by varchar2(30) ,created_on date ,modified_by varchar2(30) ,modified_on date ,constraint buy_in_pk primary key(oid)); create sequence buy_in_seq increment by 1 start with 1; create or replace trigger buy_in_insert before insert on buy_in for each row declare v_nextval number(12,0); begin if :new.oid is null then select buy_in_seq.nextval into v_nextval from dual; :new.oid := v_nextval; end if; end; {code}
2. The [ROWID|http://docs.oracle.com/cd/E11882_01/appdev.112/e13995/oracle/sql/ROWID.html] instance is returned by statement.getGeneratedKeys(). The Sql class has no mechanism to convert the proprietary "id" data representations to their corresponding JDBC types. In the case of Oracle's ROWID this can be done manually via {code} def id = result[0][0].toJdbc() {code}
3. **body:** The representation is not the issue. The issue is that primary key, which is generated by a sequence, is not returned to this collection. Only the Oracle rowid is returned. In order to retrieve the row created, one cannot use the primary key, because it was not returned. The rowid must be used. {code} database.firstRow("select * from buy_in where (rowid = :row_id)", [row_id: row_id]) {code} This is a workaround, but it is still desired to actually return a primary key, like the documentation states. bq. Returns: A list of the auto-generated column values for each inserted row (typically auto-generated keys) Maybe just alter the documentation to include this kludge.
   label: documentation
4. I see. According to the documentation, ROWID should not be used as key to refer to db rows. Can you provide the DDL for the buy_in table, so i can create a close test-case?
5. Oracle JDBC documentation says: "If key columns are not explicitly indicated, then Oracle JDBC drivers cannot identify which columns need to be retrieved. When a column name or column index array is used, Oracle JDBC drivers can identify which columns contain auto-generated keys that you want to retrieve. However, when the Statement.RETURN_GENERATED_KEYS integer flag is used, Oracle JDBC drivers cannot identify these columns. When the integer flag is used to indicate that auto-generated keys are to be returned, the ROWID pseudo column is returned as key. The ROWID can be then fetched from the ResultSet object and can be used to retrieve other columns." In other words you should use prepareStatement(String sql, String[] columnNames) and explicitly set column names which contain auto-generated keys. Otherwise ROWID is returned. It would be helpful to add optional parameter to executeInsert* functions to indicate such columns. But since Sql is implemented in Java it can be implemented with varargs to avoid adding another set of executeInsert*. Not ideal but does it's job.
6. Always I get error with various tests of "executeInsert" under Oracle 12.1. In Oracle Support I found a clue. After the patch installation everything works. *NullPointerException With getGeneratedKeys().getMetaData() in JDBC 12.1.0.1.0 (Doc ID 1619185.1)* Using the 12.1.0.1.0 JDBC driver connected to an Oracle 11.2 database, the following null pointer exception is received when getting metadata from a generated key: {code} java.lang.NullPointerException at oracle.jdbc.driver.AutoKeyInfo.initMetaDataColumnIndexes(AutoKeyInfo.java:423) at oracle.jdbc.driver.AutoKeyInfo.initMetaData(AutoKeyInfo.java:396) at oracle.jdbc.driver.OracleReturnResultSet.getMetaData(OracleReturnResultSet.java:77) {code} The same code works with JDBC 11.2.0.3.0. *Solution: Apply patch 17082699 on top of JDBC 12.1.0.1.*

7. **body:** My comment is only for documentation purposes in case others also get this error. Thank you for your inquiry.
   label: documentation
8. See https://github.com/groovy/groovy-core/pull/425
9. PR#425 applied. This added numerous {{executeInsert}} variants taking List<String> keyColumnNames (or String[] keyColumnNames for the existing Object[] vararg variant). Let me know if there are any problems.
10. This change is breaking for the method {{executeInsert(String, List)}} when supplied with params of {{(GString, List sqlParams)}}. Previously, the GString would have been converted to a String but with the new {{(GString, List columnNames)}} variant, that will match now and depending on the format of the GString, various SqlExceptions are likely to occur. If using a GString without desiring the column names functionality for auto-generated keys, you can instead: * avoid using a GString parameter (convert your GString to a String by calling .toString() or concatenate a String together manually) * use the GString only variant of the method in which case you might need to use GString.EMPTY or Sql.expand to build up your GString
11. Is this just for documentation purposes in case others also get this error? Or are you suggesting we try to guard against such NPE scenarios?
12. OK, thanks for posting the information!