|  |
| --- |
| AbstractReturningDelegate\_Handler489  catch (SQLException e) {  throw getFactory().getSQLExceptionHelper().convert(e,"Error creating contextual LOB : " + e.getMessage());  } |
| AbstractSessionImpl\_Handler501  catch (SQLException sqle) {  throw session.getFactory().getSQLExceptionHelper().convert(sqle,"could not insert: " + MessageHelper.infoString(persister),insertSQL);  } |
| AbstractSelectingDelegate\_Handler490  catch (SQLException sqle) {  throw session.getFactory().getSQLExceptionHelper().convert(sqle,"could not insert: " + MessageHelper.infoString(persister),insertSQL);  } |
| AbstractSelectingDelegate\_Handler491  catch (SQLException sqle) {  throw session.getFactory().getSQLExceptionHelper().convert(sqle,"could not retrieve generated id after insert: " + MessageHelper.infoString(persister),insertSQL);  } |
| BasicExecutor\_Handler403  catch (SQLException sqle) {  throw factory.getSQLExceptionHelper().convert(sqle,"could not execute update query",sql);  } |
| FetchingScrollableResultsImpl\_Handler508  catch (SQLException e) {  throw getSession().getFactory().getSQLExceptionHelper().convert(e,"exception calling isAfterLast()");  } |
| FetchingScrollableResultsImpl\_Handler509  catch (SQLException e) {  throw getSession().getFactory().getSQLExceptionHelper().convert(e,"exception calling isAfterLast()");  } |
| FetchingScrollableResultsImpl\_Handler510  catch (SQLException e) {  throw getSession().getFactory().getSQLExceptionHelper().convert(e,"exception calling beforeFirst()");  } |
| FetchingScrollableResultsImpl\_Handler511  catch (SQLException e) {  throw getSession().getFactory().getSQLExceptionHelper().convert(e,"Could not determine if resultset is empty due to exception calling isBeforeFirst or isAfterLast()");  } |
| GUIDGenerator\_Handler484  catch (SQLException sqle) {  throw session.getFactory().getSQLExceptionHelper().convert(sqle,"could not retrieve GUID",sql);  } |
| Helper\_Handler464  catch (SQLException e) {  throw convert(session.getFactory(),e,"Unable to clean up id table [" + tableName + "]",sql);  } |
| IdentifierGeneratorHelper\_Handler486  catch (SQLException e) {  if (StringHelper.isQuoted(identifier,dialect)) {  return extractIdentifier(rs,StringHelper.unquote(identifier,dialect),type,clazz);  }  throw e;  } |
| IdTableHelper\_Handler459  catch ( SQLException e) {  log.debugf("Error attempting to export id-table [%s] : %s",creationStatement,e.getMessage());  }  }  statement.close();  }  catch ( SQLException e) {  log.error("Unable to use JDBC Connection to create Statement",e);  }  finally {  try {  connectionAccess.releaseConnection(connection);  }  catch ( SQLException ignore) {  }  }  }  catch (SQLException e) {  log.error("Unable obtain JDBC Connection",e);  } |
| IncrementGenerator\_Handler488  catch (SQLException sqle) {  throw session.getFactory().getSQLExceptionHelper().convert(sqle,"could not fetch initial value for increment generator",sql);  } |
| QueryTranslatorImpl\_Handler447  catch (SQLException sqle) {  throw getFactory().getSQLExceptionHelper().convert(sqle,"could not execute query using iterate",getSQLString());  } |
| ScrollableResultsImpl\_Handler520  catch (SQLException sqle) {  throw convert(sqle,"could not advance using scroll()");  } |
| TableBasedDeleteHandlerImpl\_Handler465  catch ( SQLException e) {  throw convert(e,"error performing bulk delete",delete);  }  } |
| TableGenerator\_Handler471  catch (SQLException e) {  LOG.unableToReadOrInitHiValue(e);  throw e;  } |
| BatchingBatch\_Handler257  catch (SQLException e) {  LOG.debugf("SQLException escaped proxy",e);  throw sqlExceptionHelper().convert(e,"could not perform addBatch",currentStatementSql);  } |