

Non-Inheritance Extract Class Refactoring Suggestions		
Original Class [N <sub>split</sub> ]	Extract Class	
	New Class [Superclass] [Subclass]	Methods/Fields
jdbcDatabaseMetaDa ta [3]	jdbcDatabaseMetaData _new_1 [-] [-]	getDatabaseProductName() getDatabaseProductVersion() isReadOnly() getDatabaseMajorVersion() getUserName() getDatabaseMinorVersion() jdbcDatabaseMetaData(jdbcConnection) execute(String) getConnection() connection getURL()
	jdbcDatabaseMetaData _new_2 [-] [-]	translateSchema(String) supportsSchemasInIndexDefinitions() supportsSchemasInPrivilegeDefinitions() useSchemaDefault supportsSchemasInTableDefinitions()
	jdbcDatabaseMetaData _new_3 [-] [-]	supportsDataManipulationTransactionsOnly() getMaxStatements() supportsGroupBy() supportsSavepoints() supportsOpenCursorsAcrossRollback() supportsSubqueriesInIns() supportsSchemasInDataManipulation() doesMaxRowSizeIncludeBlobs() getCatalogSeparator() storesMixedCaseQuotedIdentifiers() getMaxConnections() getMaxTableNameLength() getSearchStringEscape() supportsMultipleResultSets() getIdentifierQuoteString() supportsCatalogsInProcedureCalls() othersInsertsAreVisible(int) supportsGroupByBeyondSelect() supportsSelectForUpdate() locatorsUpdateCopy() nullsAreSortedAtEnd() supportsCatalogsInIndexDefinitions() getMaxSchemaNameLength()

		storesMixedCaseIdentifiers() othersDeletesAreVisible(int) supportsOrderByUnrelated() storesUpperCaseQuotedIdentifiers() storesLowerCaseQuotedIdentifiers() dataDefinitionCausesTransactionCommit() supportsPositionedDelete() getTimeDateFunctions() supportsUnion() supportsTransactions() getDriverName() ownInsertsAreVisible(int) supportsFullOuterJoins() supportsOpenStatementsAcrossRollback() supportsResultSetConcurrency(int, int) supportsResultSetType(int) getMaxColumnsInIndex() supportsCatalogsInPrivilegeDefinitions() supportsExtendedSQLGrammar() supportsLimitedOuterJoins() supportsAlterTableWithDropColumn() supportsMinimumSQLGrammar() getDefaultTransactionIsolation() getSchemaTerm() supportsDifferentTableCorrelationNames() getSystemFunctions() getMaxColumnsInSelect() nullsAreSortedHigh() supportsConvert(int, int) supportsMultipleTransactions() getMaxBinaryLiteralLength() supportsCorrelatedSubqueries() supportsStatementPooling() supportsOpenStatementsAcrossCommit() supportsBatchUpdates() othersUpdatesAreVisible(int) supportsANSI92FullSQL() getJDBCMajorVersion() BRI_SESSION_SCOPE_IN_LIST getMaxColumnsInOrderBy() getMaxStatementLength() getNumericFunctions() supportsExpressionsInOrderBy() getProcedureTerm()
--	--	---

		<p> supportsResultSetHoldability(int)  supportsDataDefinitionAndDataManipulationTransactions()  ownUpdatesAreVisible(int)  supportsUnionAll()  whereTrue  supportsGroupByUnrelated()  allTablesAreSelectable()  supportsSubqueriesInExists()  getMaxColumnsInTable()  allProceduresAreCallable()  supportsIntegrityEnhancementFacility()  getMaxCursorNameLength()  getDriverMinorVersion()  dataDefinitionIgnoredInTransactions()  usesLocalFiles()  getMaxTablesInSelect()  supportsCatalogsInTableDefinitions()  selstar  getExtraNameCharacters()  supportsAlterTableWithAddColumn()  ownDeletesAreVisible(int)  supportsLikeEscapeClause()  getMaxCharLiteralLength()  storesUpperCaseIdentifiers()  supportsStoredProcedures()  supportsSubqueriesInComparisons()  supportsCatalogsInDataManipulation()  supportsNamedParameters()  supportsMixedCaseIdentifiers()  getMaxColumnNameLength()  supportsConvert()  BRI_TRANSACTION_SCOPE_IN_LIST  usesLocalFilePerTable()  getMaxColumnsInGroupBy()  getMaxUserNameLength()  getJDBCMinorVersion()  supportsTableCorrelationNames()  supportsOuterJoins()  insertsAreDetected(int)  deletesAreDetected(int)  supportsNonNullableColumns()  supportsMultipleOpenResults()  nullsAreSortedAtStart() </p>
--	--	--

		getMaxRowSize() supportsPositionedUpdate() supportsMixedCaseQuotedIdentifiers() nullsAreSortedLow() getSQLStateType() supportsANSI92IntermediateSQL() nullPlusNonNullIsNull() supportsSubqueriesInQuantifieds() getCatalogTerm() supportsOpenCursorsAcrossCommit() getMaxCatalogNameLength() getDriverVersion() getResultSetHoldability() storesLowerCaseIdentifiers() supportsANSI92EntryLevelSQL() getMaxIndexLength() INT_COLUMNS_NO_NULLS static {} getBestRowIdentifier(String,String,String, int, boolean) supportsTransactionIsolationLevel(int) getSQLKeywords() updatesAreDetected(int) BRI_TEMPORARY_SCOPE_IN_LIST supportsGetGeneratedKeys() supportsCoreSQLGrammar() isCatalogAtStart() getStringFunctions() getDriverMajorVersion() supportsSchemasInProcedureCalls() supportsColumnAliasing() getMaxProcedureNameLength() getTableTypes() getTypeInfo() getSchemas() getCatalogs() getCrossReference(String,String,String,String,String, String) getPrimaryKeys(String,String,String) getExportedKeys(String,String,String) getImportedKeys(String,String,String) getIndexInfo(String,String,String, boolean, boolean) getVersionColumns(String,String,String)
--	--	---

		getColumns(String,String,String,String) getTables(String,String,String,String[]) getProcedures(String,String,String) getUDTs(String,String,String, int[]) getSuperTypes(String,String,String) getColumnPrivileges(String,String,String,String) getTablePrivileges(String,String,String) getAttributes(String,String,String,String) executeSelect(String,String) wantsIsNull(String) toQueryPrefix(String) and(String,String,Object) getProcedureColumns(String,String,String,String) getSuperTables(String,String,String)
Parser [2]	Parser_new_1 [-] [-]	processColumnList(Tokenizer, boolean) addFilterColumns(TableFilter,HsqlArrayList, int) subQueryLevel getColumnNames(Database,Table,Tokenizer, boolean) addCondition(Expression,Expression) addJoinCondition(Expression,Expression,TableFilter, boolean) reset(String) parseSubquery(int,HsqlNameManager\$HsqlName[], boolean, int) parseLikePredicate(Expression) resolveSelectTableFilter(Select,HsqlArrayList,HsqlArrayList) parseSelect(int, boolean, boolean, boolean, boolean) compileSelectStatement(int) compileCallStatement() parseTableFilter(boolean) Parser(Session,Database,Tokenizer) compileInsertStatement() compileUpdateStatement() compileDeleteStatement() readSequenceExpression() parseLimit(String,Select, boolean) getInsertColumnValueExpressions(Table,Expression[], int) session checkTableWriteAccess(Table, int) parseOrderBy(Select,HsqlArrayList)

		parseColumnList() parseExpression() tokenizer parseCloseBrackets(int) parseUnion(String) parseOpenBrackets() parseOpenBracketsSelect() parseSimpleTableFilter(int) database tokenSet simpleFunctions getViewSubquery(View) setAsView(View) subQueryList noSubqueries getSortedSubqueries() getParameters() parameters clearParameters() noParameters static {}
	Parser_new_2 [-] [-]	setCompilingView() compilingView isCompilingView() resolveOrderByExpression(Expression,Select,Hsql ArrayList) resolveOrderByColumnIndex(Expression,HsqlArra yList, int) parseCaseWhen(Expression) parseBetweenPredicate(Expression) wasQuoted sTable iType readNullIfExpression() readPositionExpression() oData readExtractExpression() sToken readTrimExpression() parseInPredicate(Expression) readColumnExpression() read() readTerm()

		readToken() readDefaultClause(int) readCastExpression() parseAllAnyPredicate() readAggregate() readCondition() readCaseExpression() readSubstringExpression() readCaseWhenExpression() readCoalesceExpression() readAnd() readOr() readThis(int) iToken readConcat() readFactor() readSum() readConcatExpression() sSchema
DatabaseManagerSw ing [2]	DatabaseManagerSwin g_new_1 [-] [-]	NULL_STR GRID_BOX_TEXT BUTTON_CHECK_PERIOD keyPressed(KeyEvent) ROWCOUNTS_BOX_TEXT windowDeactivated(WindowEvent) SHOWSCHEMAS_BOX_TEXT windowActivated(WindowEvent) quoteTableName(String) quoteObjectName(String) handlePopup(MouseEvent) mouseReleased(MouseEvent) alreadyHandled mousePressed(MouseEvent) windowClosed(WindowEvent) windowOpened(WindowEvent) mouseExited(MouseEvent) AUTOCOMMIT_BOX_TEXT keyReleased(KeyEvent) butExecute mouseEntered(MouseEvent) mouseClicked(MouseEvent) SHOWTIPS_BOX_TEXT windowDeiconified(WindowEvent) AUTOREFRESH_BOX_TEXT

		jtoolbar LOGGING_BOX_TEXT windowIconified(WindowEvent) SHOWSYS_BOX_TEXT addMenuItems(JMenu,Object[]) TT_AVAILABLE defScript addMenu(JMenuBar,String,Object[]) busyText backgroundIt(Runnable,String) refreshTree() treeRefreshRunnable jframe buttonUpdaterThread stop() start() buttonUpdater currentLAF waitCursor disableButtonRunnable dummyThread enableButtonRunnable boxAutoRefresh boxLogging boxRowCounts boxShowSchemas boxShowGrid boxShowSys schemaListListener showSchemas showSys isOracle showIndexDetails schemaFilter nonSystables usertables dMeta updateSchemaList() directRefreshTree() treeModel makeNode(Object,MutableTreeNode) rowConn getRowCounts(Vector,Vector) displayRowCounts
--	--	--



		rootNode tScrollPane oracleSysUsers mitemUpdateSchemas mnuSchemas rbAllSchemas rbNativeLF prefs localActionList mitemAbout mitemHelp jbuttonExecute main() DatabaseManagerSwing(JFrame) DatabaseManagerSwing() createToolBar() tipMap showTooltips resetTooltips() jbuttonClear boxTooltips rbJavaLF rbMotifLF setStatusLine(String, int) jStatusLine iReadyStatus bMustExit setMustExit(boolean) iMaxRecent iRecent addToRecent(String) mRecent getParameter(String) showUsage() NL ABOUT_TEXT HELP_TEXT homedir tString fString oracleSysSchemas DEFAULT_RCFILE defUser defPassword
--	--	--

		defDriver defURL init() main(String[]) static {} READY_STATUS
	DatabaseManagerSwin g_new_2 [-] [-]	keyTyped(KeyEvent) tTree txtCommandScroll doLayout() gResultTable ewSplitPane pStatus tableModel nsSplitPane fMainCursor txtResultCursor txtCommandCursor getAppletContext() sRecent defDirectory refForFontDialogSwing boxAutoCommit updateAutoCommitBox() cConn autoRefresh fMain bHelp gScrollPane lTime sqlScriptBuffer connect(Connection) txtResult txtResultScroll windowClosing(WindowEvent) setLF(String) pResult setResultsInText() initGUI() actionPerformed(ActionEvent) sStatement displayResults() gridFormat setWaiting(String)

Tokenizer [2]		showHelp(String[]) showResultInText() setResultsInGrid() updateResult() gResult formatResultSet(ResultSet) clearResultPanel() testPerformance() executeSQL() insertTestData() clear() txtCommand executeCurrentSQL() testStatement(Statement,String, int)
	Tokenizer_new_1 [-] [-]	valueTokens static {} LONG_MAX_VALUE_INCREMENT lastTokenQuotedID typeLongNameFirst tokenIndex nextTokenIndex retainFirst Tokenizer() indexLongNameFirst getLength() iLength back() getToken() reset(String) Tokenizer(String) sCommand getPart(int, int) getString(char) iIndex getPosition() setPartMarker() beginIndex getPartMarker() setPartMarker(int) getLastPart()
	Tokenizer_new_2 [-] [-]	REMARK_LINE NO_TYPE TIME BOOLEAN

		LONG_NAME LONG NAMED_PARAM NUMBER REMARK DECIMAL FLOAT QUOTED_IDENTIFIER SPECIAL DATE TIMESTAMP NULL getLnType(int) getInt() getLongNamePre() sLongNameFirst wasFirstQuotedIdentifier() wasQuotedIdentifier() bWait getLongNameFirst() wasParameter() wasValue() wasLongName() getType() iType wasSimpleToken() matchThis(String) wasSimpleName() wasName() throwUnexpected() getSimpleName() getAsValue() isGetThis(String) getString() STRING getName() NAME getThis(String) sToken getBigint() wasThis(String) getSimpleToken()
Session [3]	Session_new_1 [SessionInterface]	actionTimestamp getCurrentDate()

	[-]	currentDate getCurrentTime() currentTime getCurrentTimestamp() currentTimestamp currentDateTimeSCN currentMillis transactionTimestamp getSession() setScripting(boolean) script getScripting() loggedSchema isolationMode sqlExecuteCompiledNoPreChecks(CompiledState ment,Object[]) compiledStatementExecutor sqlExecute(CompiledStatement,Object[]) isProcessingScript() isProcessingScript sessionId compiledStatementManager sqlExecute(Result) parser tokenizer savepoints releaseSavepoint(String) nestedOldTransIndex dbCommandInterpreter sqlCompileStatement(String) sqlExecuteBatchDirect(Result) oldSchema isSchemaDefintion() startSchemaDefinition(String) getTransactionSize() rowActionList isAutoCommit getAutoCommitStatement() isNestedTransaction() isNestedTransaction Session(Database,User, boolean, boolean, int) sqlExecuteBatch(Result) getAttributes() addDeleteAction(Table,Row)
--	-----	--

		addInsertAction(Table,Row) endNestedTransaction(boolean) savepoint(String) endSchemaDefinition() getSchemaName(String) setSchema(String) getSchemaHsqlName(String) getSchemaHsqlNameForWrite(String) sqlExecuteDirectNoPreChecks(String) getDatabase() database getInternalConnectionURL() performPostExecute(Result) beginNestedTransaction() currentSchema static {} emptyUpdateCount setAttributes(Result) rollbackToSavepoint(String) getIsolation() isAutoCommit() isReadOnly() resetSession() commit() setAutoCommit(boolean) prepareCommit() isClosed() setReadOnly(boolean) rollback() execute(Result) startPhasedTransaction() close() getId() setIsolation(int)
	Session_new_2 [-] [-]	getSchemaNameForWrite(String) checkDDLWrite() check(HsqlNameManager\$HsqlName, int) getUser() setUser(User) user checkAdmin() isAccessible(HsqlNameManager\$HsqlName) check(String) isAccessible(String)

		isAdmin() getUsername() resetSchema()
	Session_new_3 [-] [-]	getConnectTime() connectTime getLastIdentity() setLastIdentity(Number) lastIdentity checkReadWrite() isReadOnly getInternalConnection() intConnection dropIndex(HsqlNameManager\$HsqlName, boolean) indexArrayMap clearIndexRoots() getIndexRoot(HsqlNameManager\$HsqlName, boolean) setIndexRoot(HsqlNameManager\$HsqlName, boolean,Node) clearIndexRootsKeep() indexArrayKeepMap isClosed isProcessingLog isProcessingLog() getAndClearWarnings() sqlWarnings addWarning(HsqlException) setSQLMaxRows(int) getMaxRows() currentMaxRows getSQLMaxRows() sessionMaxRows
Select [2]	Select_new_1 [-] [-]	UNION UNIONALL NOUNION EXCEPT INTERSECT sIntoTable asteriskPositions intoType unionType unionDepth unionMaxDepth

		<pre> prepareUnions() getResultMain(Session) resolveTablesUnion(TableFilter) unionArray havingCondition Select() isResolved isResolved() getValue(Session, int) allColumnsAreDefinedIn(Expression,HashSet) resolve(Session) iHavingLen iGroupLen resultMetaData isAggregated groupColumnNames isGrouped unionSelect tFilter setFilterConditions(Session) queryCondition getResult(Session, int) getSingleResult(Session, int) mergeResults(Session,Result,Result) inAggregateOrGroupByClause(Expression) describe(Session) buildResult(Session, int) iResultLen prepareResult(Session) getDDL() prepareSort() describeResult() resolveTables() resolveAll(Session, boolean) resolveUnionColumnTypes() checkAggregateOrGroupByOrderColumns(int, int) getColumnAliases() resolveTypes(Session) exprColumns isSimilarIn(Expression, int, int) checkAggregateOrGroupByColumns(int, int) resolveTables(TableFilter) checkResolved(boolean) </pre>
--	--	--



		isDistinctSelect
	Select_new_2 [-] [-]	sortUnion limitCondition getLimitStart(Session) getLimitCount(Session, int) getMaxRowCount(Session, int) simpleLimit iOrderLen sortResult(Session,Result) sortOrder sortDirection
Expression [3]	Expression_new_1 [-] [-]	equals(Object,Object) equals() similarTo() valueData bind(Object) canBeInAggregate() isSelfAggregate() aggregateSpec isAggregate() isDistinctAggregate describe(Session) isCompare(int) INTEGER_0 INTEGER_1 SQL_TRIM_SPECIFICATION SQL_EXTRACT_FIELD_NAMES static {} findTableFilter(TableFilter[]) schema isIdentity isWritable catalog setTableColumnAttributes(Table, int) setTableColumnAttributes() precision getColumnSize() scale getColumnScale() nullability PARAM_UNKNOWN TIMEZONE_HOUR OPEN CONCAT

		VAR_SAMP SOME FUNCTION NO_NULLS ALL NULLABLE_UNKNOWN EVERY CASE BIGGER END TRIM NOT CAST SELECT AGGREGATE_NONE DOW MAX LEADING SUM ANY SECOND FOR CONVERT AND PARAM_OUT DAY FALSE AS STDDEV_POP CASEWHEN PARAM AGGREGATE_SELF OR NEGATE THEN ALTERNATIVE ASTERISK QUERY MINUTE SUBTRACT NULLIF POSITION SUBSTRING LIMIT
--	--	--

		AGGREGATE_RIGHT AGGREGATE_FUNCTION YEAR FROM NOT_EQUAL TRAILING EXISTS T_TIMEZONE_MINUTE DISTINCT EXTRACT BIGGER_EQUAL ADD MIN BOTH STDDEV_SAMP NULLABLE PARAM_IN_OUT AVG VIEW VAR_POP BETWEEN HOUR SEQUENCE ENDWHEN DIVIDE IS_NULL WHEN COUNT COLUMN MULTIPLY AGGREGATE_LEFT EQUAL COMMA PARAM_IN PLUS IS CLOSE VALUelist ELSE COALESCE IN MONTH isQueryCorrelated hList
--	--	--

		isFixedConstantValueList isAggregate(int) Expression(int,Object, boolean) isParam isParam() paramMode AGGREGATE_BOTH LIKE getColumnNr() columnIndex outerFilter IFNULL SMALLER ROW SMALLER_EQUAL testCondition(Session) getCheckSelect(Session,Table) columnQuoted isInJoin valueList likeObject oldIType subQuery joinedTableColumnIndex function testExistsCondition(Session) isAliasQuoted() getTableName() setTableName(String) tableName getColumnDDL() getBaseColumnName() isResolved() getColumnName() setColumnName(String, boolean) columnName getAlias() collectColumnName(HashSet) getType() getContextDDL() canBeInGroupBy() exprType setTrue() TRUE
--	--	---

		Expression(String,String,String) Expression(String,String, boolean) Expression(TableFilter,Column) Expression(boolean) Expression() Expression(SubQuery) Expression(Function) Expression(NumberSequence) Expression(int,Object) Expression() testInCondition(Session,Object) getRightExpression() setRightExpression() getArg2() eArg2 setLeftExpression() getArg() eArg Expression(int) Expression() testAnyAllCondition(Session,Object) getEquiJoinColumns(TableFilter, boolean[]) getAnyAllValue(Session,Object,SubQuery) setForJoin(TableFilter, boolean) collectAllColumnNames(HashSet) setLikeOptimised() checkAggregate() isColumn() isConstant() swapCondition() isConditional() isFixedConditional() isFixedConstant() replaceAliases() checkTables(HsqlArrayList) checkResolved(boolean) Expression() resolveTables(TableFilter) getDDL() getAggregatedValue(Session,Object) describe(Session, int) getValue(Session) similarTo() test(Session)
--	--	--

		resolveTypeForIn(Session) updateAggregatingValue(Session,Object) resolveTypeForLike(Session) compareValues(Session,Object,Object) setNull() isTypeEqual() getValueClassName() valueClassName getValue(Session, int) VALUE getDataType() setDataType(int) dataType getCaseWhenType(Session) setDistinctAggregate(boolean) resolveTypes(Session) collectInGroupByExpressions(HsqlArrayList) canBeInOrderBy()
	Expression_new_2 [-] [-]	aliasQuoted getExpressionForAlias() setAlias(String, boolean) columnAlias getDefinedAlias() isDescending() setDescending() isDescending
	Expression_new_3 [-] [-]	getTableSchemaName() getFilterTableName() getTableHsqlName() tableFilter getFilter()
jdbcConnection [2]	jdbcConnection_new_1 [-] [-]	finalize() isNetConn connProperties isInternal getURL() close() jdbcConnection(Session) jdbcConnection(HsqlProperties) rootWarning_mutex getWarnings() rootWarning addWarning(SQLWarning) clearWarningsNoCheck()

		clearWarnings() rollback(Savepoint) releaseSavepoint(Savepoint) rollback() setReadOnly(boolean) getTransactionIsolation() setAutoCommit(boolean) setSavepoint(String) getAutoCommit() commit() setTransactionIsolation(int) sessionProxy isReadOnly() reset()
	jdbcConnection_new_2 [-] [-]	onStartEscapeSequence(String,StringBuffer, int) nativeSQL(String) prepareStatement(String, int[]) setTypeMap(Map) prepareStatement(String,String[]) createStatement() getCatalog() getMetaData() prepareStatement(String, int) getTypeMap() setCatalog(String) setSavepoint() prepareCall(String) prepareStatement(String) setHoldability(int) getHoldability() checkClosed() isClosed isClosed() prepareCall(String, int, int) prepareStatement(String, int, int) createStatement(int, int) prepareStatement(String, int, int, int) prepareCall(String, int, int, int) createStatement(int, int, int) xlateRSHoldability(int) xlateRSConcurrency(int) xlateRSType(int)
jdbcResultSet [3]	jdbcResultSet_new_1 [-]	updateRef(int,Ref) cancelRowUpdates()

	[-]	<p>           FETCH_UNKNOWN            CLOSE_CURSORS_AT_COMMIT            updateBoolean(String, boolean)            updateBoolean(int, boolean)            FETCH_REVERSE            updateInt(int, int)            updateInt(String, int)            CONCUR_UPDATABLE            updateRef(String,Ref)            getURL(int)            moveToCurrentRow()            updateClob(String,Clob)            insertRow()            updateRow()            refreshRow()            deleteRow()            getLong(int)            getLong(String)            getCursorName()            getURL(String)            CONCUR_READ_ONLY            getObject(int,Map)            getObject(String,Map)            updateDate(int,Date)            updateDate(String,Date)            TYPE_FORWARD_ONLY            updateArray(String,Array)            setFetchSize(int)            TYPE_SCROLL_SENSITIVE            updateNull(int)            updateNull(String)            getDouble(int)            getDouble(String)            moveToInsertRow()            getWarnings()            updateString(int,String)            updateString(String,String)            updateBigDecimal(int,BigDecimal)            updateBigDecimal(String,BigDecimal)            getBoolean(int)            getBoolean(String)            FETCH_FORWARD            updateClob(int,Clob)            getInt(int)         </p>
--	-----	---



		<code>getInt(String)</code> <code>getShort(String)</code> <code>getShort(int)</code> <code>updateFloat(int, float)</code> <code>updateFloat(String, float)</code> <code>setFetchDirection(int)</code> <code>rsType</code> <code>getType()</code> <code>updateObject(int,Object)</code> <code>updateObject(String,Object)</code> <code>updateBlob(String,Blob)</code> <code>updateBlob(int,Blob)</code> <code>updateBinaryStream(int,InputStream, int)</code> <code>updateBinaryStream(String,InputStream, int)</code> <code>updateByte(int, byte)</code> <code>updateByte(String, byte)</code> <code>getArray(String)</code> <code>getArray(int)</code> <code>clearWarnings()</code> <code>getRef(int)</code> <code>getRef(String)</code> <code>updateShort(int, short)</code> <code>updateShort(String, short)</code> <code>getBigDecimal(String)</code> <code>getBigDecimal(int)</code> <code>updateTime(int,Time)</code> <code>updateTime(String,Time)</code> <code>getBigDecimal(int, int)</code> <code>getBigDecimal(String, int)</code> <code>HOLD_CURSORS_OVER_COMMIT</code> <code>TYPE_SCROLL_INSENSITIVE</code> <code>updateArray(int,Array)</code> <code>iColumnCount</code> <code>checkColumn(int)</code> <code>iUpdateCount</code> <code>getUpdateCount()</code> <code>connProperties</code> <code>rsmd</code> <code>getMetaData()</code> <code>updateObject(int,Object, int)</code> <code>updateObject(String,Object, int)</code> <code>getBytes(String)</code> <code>getBytes(int)</code> <code>updateBytes(String, byte[])</code>
--	--	---

		updateBytes(int, byte[]) updateDouble(int, double) updateDouble(String, double) getFloat(String) getFloat(int) updateAsciiStream(String,InputStream, int) updateAsciiStream(int,InputStream, int) updateTimestamp(String,Timestamp) updateTimestamp(int,Timestamp) updateLong(int, long) updateLong(String, long) updateCharacterStream(int,Reader, int) updateCharacterStream(String,Reader, int) jdbcResultSetMetaData(jdbcResultSet,HsqlProperties) init(jdbcResultSet,HsqlProperties) rowInserted() getFetchDirection() rowDeleted() rowUpdated() getFetchSize() getConcurrency() getObject(String) isNetConn sqlStatement getStatement() autoClose jdbcResultSet(jdbcStatement,Result,HsqlProperties, boolean) close() bInit findColumn(String) checkClosed() isResult() rResult nCurrent getObject(int) getColumnInType(int, int) checkAvailable() isBeforeFirst() isAfterLast() isLast() afterLast() first()
--	--	---

		previous() last() next() absolute(int) relative(int) beforeFirst() isFirst() getRow() iCurrentRow
	jdbcResultSet_new_2 [-] [-]	getUnicodeStream(int) getUnicodeStream(String) getString(String) getString(int) getClob(String) getClob(int) getCharacterStream(String) getCharacterStream(int) getAsciiStream(int) getAsciiStream(String)
	jdbcResultSet_new_3 [-] [-]	getTime(String) getTime(int) getTime(String,Calendar) getTime(int,Calendar) getDate(int,Calendar) getDate(String,Calendar) getDate(String) getDate(int) getTimestamp(String) getTimestamp(int) getTimestamp(String,Calendar) getTimestamp(int,Calendar) getBlob(int) getBlob(String) getBytes(String) getBytes(int) checkNull(Object) wasNull() bWasNull getBinaryStream(int) getBinaryStream(String)
Index [2]	Index_new_1 [-] [-]	MEMORY_INDEX DISK_INDEX POINTER_INDEX emptyIterator()

		static {} emptyIterator getIterator(Session,Node) isEmpty(Session) lastRow(Session) replace(Session,Node,Node) next(Node) set(Node, boolean,Node) child(Node, boolean) balance(Session,Node, boolean) isConstraint() isConstraint isTemp insert(Session,Row, int) getRoot() setRoot(Session,Node) root getRoot(Session) getName() indexName setName(String, boolean) onCommitPreserve colCheck useRowId pkCols pkTypes table isForward getIndexOrderValue() isUnique isUnique() getColumnTypes() colTypes collation Index(Database,HsqlNameManager\$HsqlName,Table, int[], int[], boolean, boolean, boolean, boolean, int[], int[], boolean) compareRowForInsert(Session,Row,Row) findFirstRow(Session,Object, int) findFirstRowNotNull(Session) compareRowNonUnique(Session,Object[], int[],Object[], int) findFirstRow(Session,Object[], int) collIndex
--	--	---

		getVisibleColumns() getColumns() isNull(Object[]) getColumnNameList()
	Index_new_2 [-] [-]	hasReferencedRow(Session,Object[], int[],Index) checkReferencedRows(Session,Table, int[],Index) findRow(Session,Row) search(Session,Row) size(Session) exists(Session,Object[], int[]) isNull(Object[], int[]) compareRows(Session,Object[],Object[], int[], int[]) compareRowNonUnique(Session,Object[], int[],Object[]) findNotNull(Session,Object[], int[], boolean) findFirstRow(Session,Object[], int[]) clearAll(Session) sizeEstimate() depth firstRow(Session) delete(Session,Node) clearIterators() updatableIterators findFirstRowForDelete(Session,Object[], int[])
Column [2]	Column_new_1 [-] [-]	hexToByteArray(String) createSQLString(Object, int) createSQLString(double) createSQLString(String) isPrimaryKey isPrimaryKey() setPrimaryKey(boolean) Column() columnName identityStart identityIncrement isIdentity() isIdentity getSize() colSize isNullable() isNullable setNullable(boolean) setIdentity(boolean, long, long) duplicate(boolean)

		setType() Column(HsqlNameManager\$HsqlName, boolean, int, int, int, boolean, Expression) getType() getDType() getDTypeSub() colType getDefaultValue(Session) getDefaultExpression() setDefaultExpression(Expression) defaultExpression getDefaultDDL() colScale getScale()
	Column_new_2 [-] [-]	getCombinedNumberType(int, int, int) getNumTypeWidth(int) compare(Collation, Object, Object, int) compareTo(byte[], byte[]) compareToTypeRange(Object, int) checkChar(String, int, boolean) checkVarchar(String, int, boolean) convertString(String, int) convertToDouble(Object) enforceSize(Object, int, int, int, boolean) tenPower BIG_DECIMAL_1 MIN_LONG MAX_LONG convertToLong(Object) static {} MAX_INT convertToInt(Object) MIN_INT BIG_DECIMAL_0 convertObject(Object, int) convertObject(Session, Object, int, int, int) multiply(Object, Object, int) negate(Object, int) add(Object, Object, int) subtract(Object, Object, int) concat(Object, Object) divide(Object, Object, int)
TableSorter [2]	TableSorter_new_1 [-]	DESCENDING NOT_SORTED

	-]	<p>           ASCENDING            fireTableChanged(TableModelEvent)            getSortingStatus(int)            TableSorter(TableModel,JTableHeader)            fireTableDataChanged()            getTableHeader()            setTableHeader(JTableHeader)            tableHeader            sortingStatusChanged()            mouseListener            LEXICAL_COMPARATOR            COMPARABLE_COMPARATOR            setColumnComparator(Class,Comparator)            columnComparators            static {}            TableSorter()            isSorting()            sortingColumns            setSortingStatus(int, int)            getHeaderRendererIcon(int, int)            getDirective(int)            EMPTY_DIRECTIVE            cancelSorting()            TableSorter(TableModel)         </p>
	<p>TableSorter_new_2</p> <p>[-]</p> <p>[-]</p>	<p>           fireTableStructureChanged()            modelIndex(int)            getModelToView()            modelToView            clearSortingState()            tableModelListener            getComparator(int)            getViewToModel()            viewToModel            getValueAt(int, int)            isCellEditable(int, int)            setValueAt(Object, int, int)            getTableModel()            setTableModel(TableModel)            tableModel            getRowCount()            getColumnClass(int)            getColumnName(int)            getColumnCount()         </p>
Result	Result_new_1	rTail

[2]	[-] [-]	isEmpty() rRoot getParameterData() getSize() size addAll() clear() setRows() sortResult(Session, int[], int[]) trimResult(int, int) add(Object[]) setParameterData(Object[]) removeDuplicates(Session, int) removeSecond(Session) removeDifferent(Session) append() newExecuteDirectRequest(String) write() iterator() getException() exception subSubString sessionID databaseID removeDuplicates(Session) getColumnCount() setColumnCount(int) significantColumns compareRecord(Session,Object[],Object[], int) compareRecord(Session,Object[],Object[], int[], int[]) newSessionAttributesResult() newParameterDescriptionResult(int) getUpdateCounts() newSingleColumnResult(String, int) metaData writeMulti(RowOutputBinary) readMultiResult(RowInputBinary) isUpdateCount() getType() setResultType(int) isError() mode Result(int)
-----	------------	---



		Result(int, int) Result() getSubString() subString getMainString() setMainString(String) mainString Result(RowInputBinary) Result(Throwable,String) Result(String,String, int) getStatementID() setStatementID(int) statementID newPrepareResponse(int) newFreeStmtRequest(int) Result(int, int[], int) write(RowOutputBinary) isData() read(RowInputBinary,DataInput)
	Result_new_2 [-] [-]	newReleaseSavepointRequest(String) newRollbackToSavepointRequest(String) newSetSavepointRequest(String) setMaxRows(int) setEndTranType(int) setConnectionAttrType(int) setStatementType(int) getUpdateCount() updateCount getEndTranType() getConnectionAttrType() getStatementType()
DINamespace [2]	DINamespace_new_1 [-] [-]	classForMethodFQN(String) iterateAllAccessibleMethods(Session, boolean) iterateRoutineMethods(String, boolean) getMethodFQN(Method) iterateAccessibleTriggerClassNames(User) listVisibleSessions(Session) classForName(String) getSchemaName(Object) getCatalogName(Object) catalogName iterateCatalogNames() DINamespace(Database) isReportCatalogs()

		database getInverseAliasMap() getSignature(Method) getMethodSpecificName(Method)
	DNamespace_new_2 [-] [-]	addBuiltinToSet(HashSet) isBuiltin(String) static {} isBuiltin(Class) builtin
SqlFile [2]	SqlFile_new_1 [-] [-]	DEFAULT_ELEMENT RAW_FALSE RAW_EMPTY DEFAULT_COL_DELIM DEFAULT_NULL_REP DEFAULT_HISTORY_SIZE JDBC3_DATA LINK nonVarChars pastName(String, int) JDBC3_BOOLEAN COL_EVEN COL_HEAD DEFAULT_SKIP_PREFIX COL_ODD SPACES RAW_DATA ORACLE_ELEMENT DIVIDER HSQLDB_ELEMENT SqlFile(boolean,Map) spaces(int) PRE_TD appendLine(StringBuffer,String) LS listMDSchemaCols listMDIndexCols listMDTableCols oracleSysSchemas DEFAULT_ROW_DELIM static {} historyPattern slashHistoryPattern substitutionPattern ifwhilePattern varsetPattern

		<div>foreachPattern</div> <div>plPattern</div> <div>DEFAULT_CHARSET</div> <div>specialPattern</div> <div>revnum</div> <div>genWinArgs(String)</div> <div>wincmdPattern</div> <div>loadBinary(File)</div> <div>excludeSysSchemas</div> <div>dsvRejectFile</div> <div>dsvConstCols</div> <div>dsvTargetTable</div> <div>dsvRejectReport</div> <div>nullRepToken</div> <div>dsvRowDelim</div> <div>maxHistoryLength</div> <div>history</div> <div>dsvColDelim</div> <div>dsvSkipCols</div> <div>dsvTargetFile</div> <div>dsvSkipPrefix</div> <div>pwDsv</div> <div>D_OPTIONS_TEXT</div> <div>fetchBinary</div> <div>DSV_M_SYNTAX_MSG</div> <div>DSV_X_SYNTAX_MSG</div> <div>DSV_OPTIONS_TEXT</div> <div>curLinenum</div> <div>br</div> <div>rawPrompt</div> <div>file</div> <div>primaryPrompt</div> <div>contPrompt</div> <div>nobufferYetString</div> <div>silentFetch</div> <div>lastSqlStatement</div> <div>immCmdSB</div> <div>magicPrefix</div> <div>rawMode</div> <div>RAW_LEADIN_MSG</div> <div>prepareVar</div> <div>binBuffer</div> <div>fetchingVar</div> <div>interactive</div>
--	--	--

		doPrepare curConn continueOnError recursed plMode possiblyUncommitteds stdprintln(String) execute(Connection, boolean) stdprint(String) updateUserSettings() charset importDsv(String,String) listTables(char,String) displayResultSet(Statement,ResultSet, int[],String) dsvSafe(String) SqlFile(File, boolean,Map) execute(Connection,PrintStream,PrintStream,Boolean) processSpecial(String) historize() setBuf(String) processFromBuffer() buffer processBuffHist(String) processSQL() showHistory() commandFromHistory(int) historySearch(String) streamToString(InputStream,String) dump(File) genRejectReportRecord(PrintWriter, int, int,String,String,Throwable) enforce1charSpecial(String, char) enforce1charBH(String, char) rb dereference(String, boolean) plBlockFile(String) formatNicely(Map, boolean) processPL(String) dump(String,File) load(String,File,String) userVars eval(String[]) deTerminated(String)
--	--	--

		oldestHist sqlTypeToString(int) canDisplayType(int) streamToBytes(InputStream) pad(String, int, boolean, boolean) divider(int) describe(String,String) psErr psStd errprintln(String) errprint(String) pwQuery closeQueryOutputStream() stdprint(String, boolean) stdprintln(boolean) stdprintln(String, boolean) condlPrint(String, boolean) condlPrintln(String, boolean) htmlMode PRE_TR htmlRow(int)
	SqlFile_new_2 [-] [-]	convertNumericEscapes(String) convertNumericEscape(String, int) convertEscapes(String) convertEscapes(String,String,String)
Grid [2]	Grid_new_1 [-] [-]	update() update(Graphics) setLayout(LayoutManager) add(Component) repaint() setCursor(Cursor) iGridHeight iXDrag iColDrag iGridWidth createImage(int, int) getMaxHeight(FontMetrics) gImage iFirstRow fMetrics iSbHeight iImage iSbWidth fFont

		iHeight iY sbHoriz sbVert iWidth setBounds(int, int, int, int) iX mouseExit(Event, int, int) bDrag iRowHeight handleEvent(Event) mouseDrag(Event, int, int) iColWidth mouseMove(Event, int, int) iColCount iRowCount adjustScroll() paint(Graphics) getHead() sColHead Grid() calcAutoWidth(int) setHead(String[]) addRow(String[]) getDisplay(int, int) getData() vData get(int, int)
	Grid_new_2 [-] [-]	preferredSize() getMinimumSize() dMinimum getPreferredSize() minimumSize() setMinimumSize(Dimension)
Tree [2]	Tree_new_1 [-] [-]	iFirstRow update(Graphics) addRow(String,String) update() setLayout(LayoutManager) add(Component) getMaxHeight(FontMetrics) iTreeWidth iTreeHeight repaint()

		createImage(int, int) gImage iImage iSbWidth iSbHeight fFont iIndentWidth fMetrics static {} iX iRowHeight iWidth iY sbVert sbHoriz setBounds(int, int, int, int) handleEvent(Event) paint(Graphics) adjustScroll() mouseDown(Event, int, int) vData iMaxTextLength addRow(String,String,String, int) removeAll() iRowCount Tree() iHeight
	Tree_new_2 [-] [-]	getPreferredSize() dMinimum setMinimumSize(Dimension) getMinimumSize() minimumSize() preferredSize()
HsqlTimer [2]	HsqlTimer_new_1 [ThreadFactory] [-]	setPeriod(Object, long) isFixedRate(Object) getNextScheduled(Object) isPeriodic(Object) getLastScheduled(Object) isFixedDelay(Object) scheduleAt(Date,Runnable) schedulePeriodicallyAt(Date, long,Runnable, boolean) HsqlTimer() shutDown()

		threadFactory taskRunner HsqlTimer(ThreadFactory) nextTask() shutdown() addTask(long,Runnable, long, boolean) isShutdown taskQueue restart() newThread(Runnable) compare(Object,Object) clearThread() getThread() taskRunnerThread shutdownImmediately()
	HsqlTimer_new_2 [-] [-]	cancel(Object) isCancelled(Object) scheduleAfter(long,Runnable) now() nowCount static {} schedulePeriodicallyAfter(long, long,Runnable, boolean)
ServerConfiguration [2]	ServerConfiguration_new_1 [-] [-]	getPropertiesFromFile(String) translateDefaultNoSystemExitProperty(HsqlProperties) translateDefaultDatabaseProperty(HsqlProperties) translateAddressProperty(HsqlProperties) isEmpty(String)
	ServerConfiguration_new_2 [-] [-]	newDefaultProperties(int) getDefaultPort(int, boolean) listLocalInetAddressNames() ServerConfiguration()
ArrayUtil [2]	ArrayUtil_new_1 [-] [-]	CLASS_CODE_OBJECT CLASS_CODE_BYTE haveEqualArrays(Object[],Object[], int) CLASS_CODE_DOUBLE ArrayUtil() intIndexesToBooleanArray(int[], boolean[]) CLASS_CODE_LONG CLASS_CODE_FLOAT resizeArrayIfDifferent(Object, int) findNot(int[], int) clearArray(int,Object, int, int)



		adjustArray(int,Object, int, int, int) fillArray(Object[],Object) copyColumnValues(Object[], int[],Object[]) find(byte[], int, int, int, int) static {} getClassCode(Class) classCodeMap fillSequence(int[]) copyColumnValues(int[], int[], int[]) findIn(byte[], int, int, byte[]) CLASS_CODE_CHAR CLASS_CODE_BOOLEAN haveCommonElement(int[], int[], int) findNotIn(byte[], int, int, byte[]) copyAdjustArray(Object,Object,Object, int, int) toAdjustedArray(Object,Object, int, int) arraySlice(int[], int, int) fillArray(int[], int) CLASS_CODE_SHORT duplicateArray(Object) containsAllTrueElements(boolean[], boolean[]) commonElements(int[], int[]) countCommonElements(int[], int[]) countStartIntIndexesInBooleanArray(int[], boolean[]) CLASS_CODE_INT find(Object[],Object) toAdjustedColumnArray(int[], int, int) copyArray(Object,Object, int) find(int[], int) resizeArray(Object, int) haveEqualArrays(int[], int[], int) sortArray(int[]) haveEqualSets(int[], int[], int) areEqualSets(int[], int[]) areEqual(int[], int[], int, boolean)
	ArrayUtil_new_2 [-] [-]	find(byte[], int, int, byte[]) countNonStartElementsAt(byte[], int, byte[]) countStartElementsAt(byte[], int, byte[]) containsAt(byte[], int, byte[]) countSameElements(byte[], int, byte[])