API compatibility report for the sqlite-jdbc library between 3.8.11.2 and 3.42.0.0 versions

<table>
<thead>
<tr>
<th>Library Name</th>
<th>sqlite-jdbc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version #1</td>
<td>3.8.11.2</td>
</tr>
<tr>
<td>Version #2</td>
<td>3.42.0.0</td>
</tr>
<tr>
<td>Subject</td>
<td>Binary Compatibility</td>
</tr>
</tbody>
</table>

Test Info

<table>
<thead>
<tr>
<th>Library Name</th>
<th>sqlite-jdbc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version #1</td>
<td>3.8.11.2</td>
</tr>
<tr>
<td>Version #2</td>
<td>3.42.0.0</td>
</tr>
<tr>
<td>Subject</td>
<td>Binary Compatibility</td>
</tr>
</tbody>
</table>

Test Results

| Total Java Modules | 1 |
| Total Methods / Classes | 1246 / 81 |
| Compatibility      | 62.5% |

Problem Summary

<table>
<thead>
<tr>
<th>Added Methods</th>
<th>Severity</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-</td>
<td>203</td>
</tr>
<tr>
<td>Removed Methods</td>
<td>Severity</td>
<td>Count</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
<td>-------</td>
</tr>
<tr>
<td>Problems with Data Types</td>
<td>High</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Low</td>
<td>27</td>
</tr>
<tr>
<td>Problems with Methods</td>
<td>High</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Medium</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Low</td>
<td>46</td>
</tr>
<tr>
<td>Other Changes in Data Types</td>
<td>-</td>
<td>118</td>
</tr>
</tbody>
</table>

### Added Methods 203

**sqlite-jdbc-3.42.0.0.jar, BusyHandler.class**

```java
package org.sqlite
BusyHandler.BusyHandler ()
org/sqlite/BusyHandler."<init>"():V
BusyHandler.callback ( int p1 ) [abstract] : int
org/sqlite/BusyHandler.callback:(I)I
BusyHandler.clearHandler ( Connection conn ) [static] : void
org/sqlite/BusyHandler.clearHandler:(Ljava/sql/Connection;)V
BusyHandler.setHandler ( Connection conn, BusyHandler busyHandler ) [static] : void
org/sqlite/BusyHandler.setHandler:(Ljava/sql/Connection;Lorg/sqlite/BusyHandler;)V
```

**sqlite-jdbc-3.42.0.0.jar, Collation.class**

```java
package org.sqlite
Collation.Collation ()
org/sqlite/Collation."<init>"():V
Collation.create ( Connection conn, String name, Collation f ) [static] : void
org/sqlite/Collation.create:(Ljava/sql/Connection;Ljava/lang/String;Lorg/sqlite/Collation;)V
Collation.destroy ( Connection conn, String name ) [static] : void
org/sqlite/Collation.destroy:(Ljava/sql/Connection;Ljava/lang/String;)V
Collation.xCompare ( String p1, String p2 ) [abstract] : int
```
package org.sqlite.core

CorePreparedStatement.clearBatch(): void

CorePreparedStatement.executeLargeBatch(): [long]

CorePreparedStatement.setDateByMilliseconds(int pos, Long value, Calendar calendar): void

package org.sqlite.core

CoreResultSet.getConnectionConfig(): SQLiteConnectionConfig

CoreResultSet.getDatabase(): DB

package org.sqlite.core

CoreStatement.checkIndex(int index): void

CoreStatement.getConnectionConfig(): SQLiteConnectionConfig

CoreStatement.getDatabase(): DB

CoreStatement.notifyFirstStatementExecuted(): void

package org.sqlite.core

DB.addCommitListener(SQLiteCommitListener listener): void

DB.addUpdateListener(SQLiteUpdateListener listener): void
DB.backup (String \( p_1 \), String \( p_2 \), DB.ProgressObserver \( p_3 \), int \( p_4 \), int \( p_5 \), int \( p_6 \)) [abstract]

org/sqlite/core/DB.backup:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III)I

DB.busy_handler (BusyHandler \( p_1 \)) [abstract]: void

org/sqlite/core/DB.busy_handler:(Lorg/sqlite/BusyHandler;)V

DB.changes () [abstract]: long

org/sqlite/core/DB.changes:()J

DB.clear_progress_handler () [abstract]: void

org/sqlite/core/DB.clear_progress_handler:()V

DB.create_collation (String \( p_1 \), Collation \( p_2 \)) [abstract]: int

org/sqlite/core/DB.create_collation:(Ljava/lang/String;Lorg/sqlite/Collation;)I

DB.create_function (String \( p_1 \), Function \( p_2 \), int \( p_3 \), int \( p_4 \)) [abstract]: int

org/sqlite/core/DB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;II)I

DB.DB (String url, String fileName, SQLiteConfig config)

org/sqlite/core/DB."<init>":(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/SQLiteConfig;)V

DB.destroy_collation (String \( p_1 \)) [abstract]: int

org/sqlite/core/DB.destroy_collation:(Ljava/lang/String;)I

DB.exec (String sql, boolean autoCommit) : void

org/sqlite/core/DB.exec:(Ljava/lang/String;Z)V

DB.executeUpdate (CoreStatement stmt, Object[] vals) : long

org/sqlite/core/DB.executeUpdate:(Lorg/sqlite/core/CoreStatement;[Ljava/lang/Object;)J

DB.finalize (SafeStmtPtr safePtr, long ptr) : int

org/sqlite/core/DB.finalize:(Lorg/sqlite/core/SafeStmtPtr;J)I

DB.getConfig () : SQLiteConfig

org/sqlite/core/DB.getConfig:()Lorg/sqlite/SQLiteConfig;

DB.getUrl () : String

org/sqlite/core/DB.getUrl:(Ljava/lang/String;

DB.isClosed () : boolean

org/sqlite/core/DB.isClosed:()Z

DB.limit (int \( p_1 \), int \( p_2 \)) [abstract] : int

org/sqlite/core/DB.limit:(II)I

DB.newSQLException (int errorCode, String errorMessage) [static]: SQLiteException

org/sqlite/core/DB.newSQLException:(ILjava/lang/String;)Lorg/sqlite/S

QLiteException;

DB.open (String file, int openFlags) : void

org/sqlite/core/DB.open:(Ljava/lang/String;I)V

DB.prepare (String \( p_1 \)) [abstract]: SafeStmtPtr
org/sqlite/core/DB.prepare:(Ljava/lang/String;)Lorg/sqlite/core/SafeStmtPtr;
DB.register_progress_handler ( int p1, ProgressHandler p2 ) [abstract] : void
org/sqlite/core/DB.register_progress_handler:(ILorg/sqlite/ProgressHandler;)V
DB.removeCommitListener ( SQLiteCommitListener listener ) : void
org/sqlite/core/DB.removeCommitListener:(Lorg/sqlite/SQLiteCommitListener;)V
DB.removeUpdateListener ( SQLiteUpdateListener listener ) : void
org/sqlite/core/DB.removeUpdateListener:(Lorg/sqlite/SQLiteUpdateListener;)V
DB.restore ( String p1, String p2, DB.ProgressObserver p3, int p4, int p5, int p6 ) [abstract]
: int
org/sqlite/core/DB.restore:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III)I
DB.total_changes ( ) [abstract] : long
org/sqlite/core/DB.total_changes:()J

sqlite-jdbc-3.42.0.0.jar, Function.class
package org.sqlite
Function.create ( Connection conn, String name, Function f, int flags ) [static] : void
org/sqlite/Function.create:(Ljava/sql/Connection;Ljava/lang/String;Lorg/sqlite/Function;I)V
Function.create ( Connection conn, String name, Function f, int nArgs, int flags ) [static] : void
org/sqlite/Function.create:(Ljava/sql/Connection;Ljava/lang/String;Lorg/sqlite/Function;II)V
Function.destroy ( Connection conn, String name, int nArgs ) [static] : void
org/sqlite/Function.destroy:(Ljava/sql/Connection;Ljava/lang/String;I)V

sqlite-jdbc-3.42.0.0.jar, HexKeyMode.class
package org.sqlite
SQLiteConfig.HexKeyMode.getValue ( ) : String
org/sqlite/SQLiteConfig.HexKeyMode.getValue:(Ljava/lang/String;)
SQLiteConfig.HexKeyMode.valueOf ( String name ) [static] : SQLiteConfig.HexKeyMode
org/sqlite/SQLiteConfig.HexKeyMode.valueOf:(Ljava/lang/String;)Lorg/sqlite/SQLiteConfig$HexKeyMode;
SQLiteConfig.HexKeyMode.values ( ) [static] : SQLiteConfig.HexKeyMode[]
org/sqlite/SQLiteConfig.HexKeyMode.values:(Ljava/lang/String;)[Lorg/sqlite/SQLiteConfig$HexKeyMode;
package org.sqlite

JDBC.createConnection ( String url, Properties prop ) [static] : SQLiteConnection

package org.sqlite.jdbc3

JDBC3Connection.tryEnforceTransactionMode () : void

package org.sqlite.jdbc3

JDBC3PreparedStatement.executeUpdate ( String sql, int autoGeneratedKeys ) : int
JDBC3PreparedStatement.executeUpdate ( String sql, int[] colinds ) : int
JDBC3PreparedStatement.executeUpdate ( String sql, String[] colnames ) : int
JDBC3PreparedStatement.executeUpdate ( String sql, int[] colinds, String[] cols ) : int
String;

JDBC3PreparedStatement.executeUpdate ( String sql, String[] cols ) : int


JDBC3PreparedStatement.invalid () : SQLException

org/sqlite/jdbc3/JDBC3PreparedStatement.invalid:()Ljava/sql/SQLException;

JDBC3PreparedStatement.unsupported () : SQLException

org/sqlite/jdbc3/JDBC3PreparedStatement.unsupported:()Ljava/sql/SQLException;

sqlite-jdbc-3.42.0.0.jar, JDBC3ResultSet.class

package org.sqlite.jdbc3

JDBC3ResultSet.safeGetColumnType ( int col ) : int

org/sqlite/jdbc3/JDBC3ResultSet.safeGetColumnType:(I)I

sqlite-jdbc-3.42.0.0.jar, JDBC3Statement.class

package org.sqlite.jdbc3

JDBC3Statement.executeLargeBatch () : long[

org/sqlite/jdbc3/JDBC3Statement.executeLargeBatch:()[J

JDBC3Statement.executeLargeUpdate ( String sql ) : long

org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;);J

JDBC3Statement.executeLargeUpdate ( String sql, int autoGeneratedKeys ) : long

org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;I)J

JDBC3Statement.executeLargeUpdate ( String sql, int[] colinds ) : long

org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;[I)J

JDBC3Statement.executeLargeUpdate ( String sql, String[] cols ) : long

org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;[Ljava/lang/String;);J

JDBC3Statement.getLargeMaxRows () : long

org/sqlite/jdbc3/JDBC3Statement.getLargeMaxRows:()J

JDBC3Statement.getLargeUpdateCount () : long

org/sqlite/jdbc3/JDBC3Statement.getLargeUpdateCount:()J

JDBC3Statement.setLargeMaxRows ( long max ) : void

org/sqlite/jdbc3/JDBC3Statement.setLargeMaxRows:(J)V

JDBC3Statement.unsupported () : SQLException

org/sqlite/jdbc3/JDBC3Statement.unsupported:()Ljava/sql/SQLException;
JDBC3Statement.withConnectionTimeout( JDBC3Statement.SQLCallable<T> callable ) : T
org/sqlite/jdbc3/JDBC3Statement.withConnectionTimeout:(Lorg/sqlite
/jdbc3/JDBC3Statement$SQLCallable;)Ljava/lang/Object;

sqlite-jdbc-3.42.0.0.jar, JDBC4PreparedStatement.class
package org.sqlite.jdbc4
JDBC4PreparedStatement.toString() : String
org/sqlite/jdbc4/JDBC4PreparedStatement.toString():()Ljava/lang/String
;

sqlite-jdbc-3.42.0.0.jar, JDBC4ResultSet.class
package org.sqlite.jdbc4
JDBC4ResultSet.close() : void
org/sqlite/jdbc4/JDBC4ResultSet.close():V
JDBC4ResultSet.unsupported() : SQLException
org/sqlite/jdbc4/JDBC4ResultSet.unsupported():Ljava/sql/SQLException;

sqlite-jdbc-3.42.0.0.jar, JDBC4Statement.class
package org.sqlite.jdbc4
JDBC4Statement.close() : void
org/sqlite/jdbc4/JDBC4Statement.close():V

sqlite-jdbc-3.42.0.0.jar, LibraryLoaderUtil.class
package org.sqlite.util
LibraryLoaderUtil.getNativeLibName() [static] : String
org/sqlite/util/LibraryLoaderUtil.getNativeLibName():Ljava/lang/String;
LibraryLoaderUtil.getNativeLibResourcePath() [static] : String
org/sqlite/util/LibraryLoaderUtil.getNativeLibResourcePath():Ljava/lang/String;
LibraryLoaderUtil.hasNativeLib( String path, String libraryName ) [static] : boolean
org/sqlite/util/LibraryLoaderUtil.hasNativeLib:(Ljava/lang/String;Ljava/
lang/String;)Z
LibraryLoaderUtil.LibraryLoaderUtil()
org/sqlite/util/LibraryLoaderUtil."<init>":()V

sqlite-jdbc-3.42.0.0.jar, NativeDB.class
package org.sqlite.core
NativeDB.backup( String dbName, String destFileName, DB.ProgressObserver observer, int
sleepTimeMillis, int nTimeouts, int pagesPerStep ) : int
org/sqlite/core/NativeDB.backup:(Ljava/lang/String;Ljava/lang/String;
Lorg/sqlite/core/DB$ProgressObserver;III)I
NativeDB.busy_handler ( BusyHandler p1 ) : void
org/sqlite/core/NativeDB.busy_handler:(Lorg/sqlite/BusyHandler;)V
NativeDB.changes ( ) : long
org/sqlite/core/NativeDB.changes:()J
NativeDB.clear_progress_handler ( ) : void
org/sqlite/core/NativeDB.clear_progress_handler:()V
NativeDB.create_collation ( String name, Collation coll ) : int
org/sqlite/core/NativeDB.create_collation:(Ljava/lang/String;Lorg/sqlite/Collation;)I
NativeDB.create_function ( String name, Function func, int nArgs, int flags ) : int
org/sqlite/core/NativeDB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;II)I
NativeDB.destroy_collation ( String name ) : int
org/sqlite/core/NativeDB.destroy_collation:(Ljava/lang/String;)I
NativeDB.limit ( int p1, int p2 ) : int
org/sqlite/core/NativeDB.limit:(II)I
NativeDB.NativeDB ( String url, String fileName, SQLiteConfig config )
org/sqlite/core/NativeDB."<init>":(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite.SQLiteConfig;)V
NativeDB.register_progress_handler ( int p1, ProgressHandler p2 ) : void
org/sqlite/core/NativeDB.register_progress_handler:(ILorg/sqlite/ProgressHandler;)V
NativeDB.restore ( String dbName, String sourceFileName, DB.ProgressObserver observer, int sleepTimeMillis, int nTimeouts, int pagesPerStep ) : int
org/sqlite/core/NativeDB.restore:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III)I
NativeDB.total_changes ( ) : long
org/sqlite/core/NativeDB.total_changes:()J

sqlite-jdbc-3.42.0.0.jar, OSInfo.class
package org.sqlite.util
OSInfo.isAndroid ( ) [static] : boolean
org/sqlite/util/OSInfo.isAndroid:()Z
OSInfo.isAndroidRuntime ( ) [static] : boolean
org/sqlite/util/OSInfo.isAndroidRuntime:()Z
OSInfo.isAndroidTermux ( ) [static] : boolean
org/sqlite/util/OSInfo.isAndroidTermux:()Z
OSInfo.isMusl ( ) [static] : boolean
org/sqlite/util/OSInfo.isMusl:()Z

sqlite-jdbc-3.42.0.0.jar, ProcessRunner.class
package org.sqlite.util
ProcessRunner.ProcessRunner ( )
org/sqlite/util/ProcessRunner."<init>"():V

sqlite-jdbc-3.42.0.0.jar, ProgressHandler.class
package org.sqlite
ProgressHandler.clearHandler ( Connection conn ) [static] : void
org/sqlite/ProgressHandler.clearHandler:(Ljava/sql/Connection;)V
ProgressHandler.progress ( ) [abstract] : int
org/sqlite/ProgressHandler.progress():I
ProgressHandler.ProgressHandler ( )
org/sqlite/ProgressHandler."<init>"():V
ProgressHandler.setHandler ( Connection conn, int vmCalls, ProgressHandler progressHandler ) [static] : void
org/sqlite/ProgressHandler.setHandler:(Ljava/sql/Connection;ILorg/sqlite/ProgressHandler;)V

sqlite-jdbc-3.42.0.0.jar, QueryUtils.class
package org.sqlite.util
QueryUtils.QueryUtils ( )
org/sqlite/util/QueryUtils."<init>"():V
QueryUtils.valuesQuery ( List<String> columns, List<List<Object>> valuesList ) [static] :
  String
org/sqlite/util/QueryUtils.valuesQuery:(Ljava/util/List;Ljava/util/List;)Ljava/lang/String;

sqlite-jdbc-3.42.0.0.jar, SafePtrConsumer.class
package org.sqlite.core
SafeStmtPtr.SafePtrConsumer<E>.run ( DB p1, long p2 ) [abstract] : void
org/sqlite/core/SafeStmtPtr.SafePtrConsumer.run:(Lorg/sqlite/core/DB;J)V

sqlite-jdbc-3.42.0.0.jar, SafePtrDoubleFunction.class
package org.sqlite.core
SafeStmtPtr.SafePtrDoubleFunction<E>.run ( DB p1, long p2 ) [abstract] : double
org/sqlite/core/SafeStmtPtr.SafePtrDoubleFunction.run:(Lorg/sqlite/core/DB;J)D

sqlite-jdbc-3.42.0.0.jar, SafePtrFunction.class
package org.sqlite.core
SafeStmtPtr.SafePtrFunction<T,E>.run ( DB p1, long p2 ) [abstract] : T
org/sqlite/core/SafeStmtPtr.SafePtrFunction.run:(Lorg/sqlite/core/DB; J)Ljava/lang/Object;

sqlite-jdbc-3.42.0.0.jar, SafePtrIntFunction.class
package org.sqlite.core
SafeStmtPtr.SafePtrIntFunction<E>.run ( DB p1, long p2 ) [abstract] : int
org/sqlite/core/SafeStmtPtr.SafePtrIntFunction.run:(Lorg/sqlite/core/DB;J)I

sqlite-jdbc-3.42.0.0.jar, SafePtrLongFunction.class
package org.sqlite.core
SafeStmtPtr.SafePtrLongFunction<E>.run ( DB p1, long p2 ) [abstract] : long
org/sqlite/core/SafeStmtPtr.SafePtrLongFunction.run:(Lorg/sqlite/core/DB;J)Ljava/lang/Object;

sqlite-jdbc-3.42.0.0.jar, SafeStmtPtr.class
package org.sqlite.core
SafeStmtPtr.close () : int
org/sqlite/core/SafeStmtPtr.close():I
SafeStmtPtr.equals ( Object o ) : boolean
org/sqlite/core/SafeStmtPtr.equals:(Ljava/lang/Object;)Z
SafeStmtPtr.hashCode () : int
org/sqlite/core/SafeStmtPtr.hashCode():I
SafeStmtPtr.isClosed () : boolean
org/sqlite/core/SafeStmtPtr.isClosed():Z
SafeStmtPtr.safeRun ( SafeStmtPtr.SafePtrFunction<T,E> run ) : T
org/sqlite/core/SafeStmtPtr.safeRun:(Lorg/sqlite/core/SafeStmtPtr$SafePtrFunction;)Ljava/lang/Object;
SafeStmtPtr.safeRunConsume ( SafeStmtPtr.SafePtrConsumer<E> run ) : void
org/sqlite/core/SafeStmtPtr.safeRunConsume:(Lorg/sqlite/core/SafeStmtPtr$SafePtrConsumer;)V
SafeStmtPtr.safeRunDouble ( SafeStmtPtr.SafePtrDoubleFunction<E> run ) : double
org/sqlite/core/SafeStmtPtr.safeRunDouble:(Lorg/sqlite/core/SafeStmtPtr$SafePtrDoubleFunction;)D
SafeStmtPtr.safeRunInt ( SafeStmtPtr.SafePtrIntFunction<E> run ) : int
org/sqlite/core/SafeStmtPtr.safeRunInt:(Lorg/sqlite/core/SafeStmtPtr$SafePtrIntFunction;)I
SafeStmtPtr.safeRunLong ( SafeStmtPtr.SafePtrLongFunction<E> run ) : long
org/sqlite/core/SafeStmtPtr.safeRunLong:(Lorg/sqlite/core/SafeStmtPtr$SafePtrLongFunction;)J
SafeStmtPtr.SafeStmtPtr ( DB db, long ptr )
org/sqlite/core/SafeStmtPtr."<init>":(Lorg/sqlite/core/DB;J)V

sqlite-jdbc-3.42.0.0.jar, SQLCallable.class
package org.sqlite.jdbc3
JDBC3Statement.SQLCallable<T>.call ( ) [abstract] : T
org/sqlite/jdbc3/JDBC3Statement.SQLCallable.call:(Ljava/lang/Object;

sqlite-jdbc-3.42.0.0.jar, SQLiteCommitListener.class
package org.sqlite
SQLiteCommitListener.onCommit ( ) [abstract] : void
org/sqlite/SQLiteCommitListener.onCommit:()V
SQLiteCommitListener.onRollback ( ) [abstract] : void
org/sqlite/SQLiteCommitListener.onRollback:()V

sqlite-jdbc-3.42.0.0.jar, SQLiteConfig.class
package org.sqlite
SQLiteConfig.defeForeignKeys ( boolean enable ) : void
org/sqlite/SQLiteConfig.defeForeignKeys:(Z)V
SQLiteConfig.getBusyTimeout ( ) : int
org/sqlite/SQLiteConfig.getBusyTimeout:()I
SQLiteConfig.isExplicitReadOnly ( ) : boolean
org/sqlite/SQLiteConfig.isExplicitReadOnly:()Z
SQLiteConfig.newConnectionConfig ( ) : SQLiteConnectionConfig
org/sqlite/SQLiteConfig.newConnectionConfig:()Lorg/sqlite/SQLiteConnectionConfig;
SQLiteConfig.setApplicationId ( int id ) : void
org/sqlite/SQLiteConfig.setApplicationId:(I)V
SQLiteConfig.setBusyTimeout ( int milliseconds ) : void
org/sqlite/SQLiteConfig.setBusyTimeout:(I)V
SQLiteConfig.setExplicitReadOnly ( boolean readOnly ) : void
org/sqlite/SQLiteConfig.setExplicitReadOnly:(Z)V
SQLiteConfig.setHexKeyMode ( SQLiteConfig.HexKeyMode mode ) : void
org/sqlite/SQLiteConfig.setHexKeyMode:(Lorg/sqlite/SQLiteConfig$HexKeyMode;)V
SQLiteConfig.setLegacyAlterTable ( boolean flag ) : void
org/sqlite/SQLiteConfig.setLegacyAlterTable:(Z)V

sqlite-jdbc-3.42.0.0.jar, SQLiteConnection.class
package org.sqlite
SQLiteConnection.addCommitListener ( SQLiteCommitListener listener ) : void
org/sqlite/SQLiteConnection.isClosed():Z
SQLiteConnection.isFirstStatementExecuted(): boolean
org/sqlite/SQLiteConnection.isFirstStatementExecuted():Z
SQLiteConnection.libversion(): String
org/sqlite/SQLiteConnection.libversion():Ljava/lang/String;
SQLiteConnection.removeCommitListener(SQLiteCommitListener listener): void
org/sqlite/SQLiteConnection.removeCommitListener:(Lorg/sqlite/SQLiteCommitListener;)V
SQLiteConnection.removeUpdateListener(SQLiteUpdateListener listener): void
org/sqlite/SQLiteConnection.removeUpdateListener:(Lorg/sqlite/SQLiteUpdateListener;)V
SQLiteConnection.rollback(): void
org/sqlite/SQLiteConnection.rollback():V
SQLiteConnection.setAutoCommit(boolean ac): void
org/sqlite/SQLiteConnection.setAutoCommit:(Z)V
SQLiteConnection.setBusyTimeout(int timeoutMillis): void
org/sqlite/SQLiteConnection.setBusyTimeout:(I)V
SQLiteConnection.setCurrentTransactionMode(SQLiteConfig.TransactionMode currentTransactionMode): void
org/sqlite/SQLiteConnection.setCurrentTransactionMode:(Lorg/sqlite/SQLiteConfig$TransactionMode;)V
SQLiteConnection.setFirstStatementExecuted(boolean firstStatementExecuted): void
org/sqlite/SQLiteConnection.setFirstStatementExecuted:(Z)V
SQLiteConnection.setLimit(SQLiteLimits limit, int value): void
org/sqlite/SQLiteConnection.setLimit:(Lorg/sqlite/SQLiteLimits;I)V
SQLiteConnection.setTransactionIsolation(int level): void
org/sqlite/SQLiteConnection.setTransactionIsolation:(I)V
SQLiteConnection.setTransactionMode(SQLiteConfig.TransactionMode mode): void
org/sqlite/SQLiteConnection.setTransactionMode:(Lorg/sqlite/SQLiteConfig$TransactionMode;)V
SQLiteConnection.SQLiteConnection(DB db)
org/sqlite/SQLiteConnection."<init>":(Lorg/sqlite/core/DB;)V
SQLiteConnection.transactionPrefix(): String
org/sqlite/SQLiteConnection.transactionPrefix():Ljava/lang/String;

sqlite-jdbc-3.42.0.0.jar, SQLiteConnectionConfig.class
package org.sqlite
SQLiteConnectionConfig.copyConfig(): SQLiteConnectionConfig
org/sqlite/SQLiteConnectionConfig.copyConfig():Lorg/sqlite/SQLiteConnectionConfig;
SQLiteConnectionConfig.fromPragmaTable(Properties pragmaTable) [static]
: SQLiteConnectionConfig
org/sqlite/SQLiteConnectionConfig.fromPragmaTable:(Ljava/util/Properties;)Lorg/sqlite/SQLiteConnectionConfig;

SQLiteConnectionConfig.getDateClass () : SQLiteConfig.DateClass

org/sqlite/SQLiteConnectionConfig.getDateClass():Lorg/sqlite/SQLiteConfig$DateClass;

SQLiteConnectionConfig.getDateFormat () : FastDateFormat

org/sqlite/SQLiteConnectionConfig.getDateFormat():(Ljava/date/FastDateFormat;

SQLiteConnectionConfig.getDateMultiplier () : long

org/sqlite/SQLiteConnectionConfig.getDateMultiplier():(J)

SQLiteConnectionConfig.getDatePrecision () : SQLiteConfig.DatePrecision

org/sqlite/SQLiteConnectionConfig.getDatePrecision():(Lorg/sqlite/SQLiteConfig$DatePrecision;

SQLiteConnectionConfig.getDateStringFormat () : String

org/sqlite/SQLiteConnectionConfig.getDateStringFormat():(Ljava/lang/String;

SQLiteConnectionConfig.getTransactionIsolation () : int

org/sqlite/SQLiteConnectionConfig.getTransactionIsolation():(I)

SQLiteConnectionConfig.getTransactionMode () : SQLiteConfig.TransactionMode

org/sqlite/SQLiteConnectionConfig.getTransactionMode():(Lorg/sqlite/SQLiteConfig$TransactionMode;

SQLiteConnectionConfig.isAutoCommit () : boolean

org/sqlite/SQLiteConnectionConfig.isAutoCommit():(Z)

SQLiteConnectionConfig.setAutoCommit ( boolean autoCommit ) : void

org/sqlite/SQLiteConnectionConfig.setAutoCommit:(Z)V

SQLiteConnectionConfig.setDateTimeClass ( SQLiteConfig.DateClass dateClass ) : void

org/sqlite/SQLiteConnectionConfig.setDateTimeClass:(Lorg/sqlite/SQLiteConfig$DateClass;)V

SQLiteConnectionConfig.setDatePrecision ( SQLiteConfig.DatePrecision datePrecision ) :

org/sqlite/SQLiteConnectionConfig.setDatePrecision:(Lorg/sqlite/SQLiteConfig$DatePrecision;)V

SQLiteConnectionConfig.setStringFormat ( String dateStringFormat ) : void

org/sqlite/SQLiteConnectionConfig.setStringFormat:(Ljava/lang/String;)V

SQLiteConnectionConfig.setTransactionIsolation ( int transactionIsolation ) : void

org/sqlite/SQLiteConnectionConfig.setTransactionIsolation:(I)V

SQLiteConnectionConfig.setTransactionMode ( SQLiteConfig.TransactionMode transactionMode ) : void

org/sqlite/SQLiteConnectionConfig.setTransactionMode:(Lorg/sqlite/SQLiteConfig$TransactionMode;)V

SQLiteConnectionConfig.SQLiteConnectionConfig ( SQLiteConfig.DateClass dateClass,
SQLiteLimits.valueOf ( String name ) [static] : SQLiteLimits
org/sqlite/SQLiteLimits.valueOf:(Ljava/lang/String;);}org/sqlite/SQLiteLimits;
SQLiteLimits.values ( ) [static] : SQLiteLimits[]
org/sqlite/SQLiteLimits.values():[Ljava/sqlite/SQLiteLimits;

sqlite-jdbc-3.42.0.0.jar, SQLitePooledConnection.class
package org.sqlite.java
SQLitePooledConnection.getListeners () : List<ConnectionEventListener>
org/sqlite/java/SQLitePooledConnection.getListeners():Ljava/util/List;
SQLitePooledConnection.getPhysicalConn () : SQLiteConnection
org/sqlite/java/SQLitePooledConnection.getPhysicalConn():Lorg/sqlite/SQLiteConnection;
SQLitePooledConnection.SQLitePooledConnection ( SQLiteConnection physicalConn )
org/sqlite/java/SQLitePooledConnection."<init>":(Lorg/sqlite/SQLiteConnection;)V

sqlite-jdbc-3.42.0.0.jar, SQLiteUpdateListener.class
package org.sqlite
SQLiteUpdateListener.onUpdate ( SQLiteUpdateListener.Type p1, String p2, String p3, long p4 ) [abstract] : void
org/sqlite/SQLiteUpdateListener.onUpdate:(Lorg/sqlite/SQLiteUpdateListener$Type;Ljava/lang/String;Ljava/lang/String;J)V

sqlite-jdbc-3.42.0.0.jar, Type.class
package org.sqlite
SQLiteUpdateListener.Type.valueOf ( String name ) [static] : SQLiteUpdateListener.Type
org/sqlite/SQLiteUpdateListener.Type.valueOf:(Ljava/lang/String;;)Lorg/sqlite/SQLiteUpdateListener$Type;
SQLiteUpdateListener.Type.values ( ) [static] : SQLiteUpdateListener.Type[]
org/sqlite/SQLiteUpdateListener.Type.values():[Ljava/sqlite/SQLiteUpdateListener$Type;

sqlite-jdbc-3.42.0.0.jar, VersionHolder.class
package org.sqlite
SQLiteJDBCLoader.VersionHolder.SQLiteJDBCLoader.VersionHolder ( )
org/sqlite/SQLiteJDBCLoader.VersionHolder."<init>"():V

sqlite-jdbc-3.42.0.0.jar, Window.class
package org.sqlite
Function.Window.Function.Window()
org/sqlite/Function.Window."<init>"():V
Function.Window.xInverse() [abstract]: void
org/sqlite/Function.Window.xInverse():V
Function.Window.xValue() [abstract]: void
org/sqlite/Function.Window.xValue():V

to the top

Removed Methods 45

sqlite-jdbc-3.8.11.2.jar, CoreConnection.class
package org.sqlite.core
CoreConnection.checkCursor( int rst, int rsc, int rsh ): void
org/sqlite/core/CoreConnection.checkCursor:(III)V
CoreConnection.checkOpen(): void
org/sqlite/core/CoreConnection.checkOpen():V
CoreConnection.close(): void
org/sqlite/core/CoreConnection.close():V
CoreConnection.CoreConnection( String url, String fileName, Properties prop )
or/SQLite/connection."<init>":(Ljava/lang/String;Ljava/lang/
String;Ljava/util/Properties;):V
CoreConnection.db(): DB
org/sqlite/core/CoreConnection.db():Lorg/sqlite/core/DB;
CoreConnection.getBusyTimeout(): int
org/sqlite/core/CoreConnection.getBusyTimeout():I
CoreConnection.getDriverVersion(): String
org/sqlite/core/CoreConnection.getDriverVersion():Ljava/lang/String;
CoreConnection.libversion(): String
org/sqlite/core/CoreConnection.libversion():Ljava/lang/String;
CoreConnection.setBusyTimeout( int milliseconds ): void
org/sqlite/core/CoreConnection.setBusyTimeout:(I)V
CoreConnection.setTransactionMode( SQLiteConfig.TransactionMode mode ): void
org/sqlite/core/CoreConnection.setTransactionMode:(Lorg/sqlite/SQLi
teConfig$TransactionMode;):V
CoreConnection.url(): String
org/sqlite/core/CoreConnection.url():Ljava/lang/String;

sqlite-jdbc-3.8.11.2.jar, CorePreparedStatement.class
package org.sqlite.core
CorePreparedStatement.checkParameters( ) : void
org/sqlite/core/CorePreparedStatement.checkParameters:()V
CorePreparedStatement.setDateByMilliseconds( int pos, Long value ) : void
org/sqlite/core/CorePreparedStatement.setDateByMilliseconds:(ILjava/lang/Long;)V

sqlite-jdbc-3.8.11.2.jar, DB.class
package org.sqlite.core
DB.changes( ) [abstract] : int
org/sqlite/core/DB.changes():()I
DB.create_function( String p1, Function p2 ) [abstract] : int
org/sqlite/core/DB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;)I
DB.DB( )
org/sqlite/core/DB."<init>"():()V
DB.exec( String sql ) : void
org/sqlite/core/DB.exec:(Ljava/lang/String;)V
DB.executeUpdate( CoreStatement stmt, Object[] vals ) : int
org/sqlite/core/DB.executeUpdate:(Lorg/sqlite/core/CoreStatement;[Ljava/lang/Object;)I
DB.finalize( CoreStatement stmt ) : int
org/sqlite/core/DB.finalize:(Lorg/sqlite/core/CoreStatement;)I
DB.newSQLException( int errorCode, String errorMessage ) [static] : SQLException
org/sqlite/core/DB.newSQLException:(ILjava/lang/String;)Ljava/sql/SQLException;
DB.open( SQLiteConnection conn, String file, int openFlags ) : void
org/sqlite/core/DB.open:(Lorg/sqlite/SQLiteConnection;Ljava/lang/String;)V
DB.prepare( String p1 ) [abstract] : long
org/sqlite/core/DB.prepare:(Ljava/lang/String;)J
DB.total_changes( ) [abstract] : int
org/sqlite/core/DB.total_changes():()I
DB.value_bytes( Function p1, int p2 ) [abstract] : int
org/sqlite/core/DB.value_bytes:(Lorg/sqlite/Function;)I

sqlite-jdbc-3.8.11.2.jar, Function.class
package org.sqlite
Function.value_bytes( int arg ) : int
org/sqlite/Function.value_bytes:(I)I

sqlite-jdbc-3.8.11.2.jar, JDBC.class
package org.sqlite

JDBC.createConnection ( String url, Properties prop ) [static] : Connection

org/sqlite/JDBC.createConnection:(Ljava/lang/String;Ljava/util/Properties;)Ljava/sql/Connection;

sqlite-jdbc-3.8.11.2.jar, JDBC3PreparedStatement.class
package org.sqlite.jdbc3
JDBC3PreparedStatement.unused ( ) : SQLException
org/sqlite/jdbc3/JDBC3PreparedStatement.unused:(Ljava/sql/SQLException);

sqlite-jdbc-3.8.11.2.jar, JDBC3ResultSet.class
package org.sqlite.jdbc3
JDBC3ResultSet.checkCalendar ( Calendar cal ) : void
org/sqlite/jdbc3/JDBC3ResultSet.checkCalendar:(Ljava/util/Calendar;)
V

sqlite-jdbc-3.8.11.2.jar, JDBC3Statement.class
package org.sqlite.jdbc3
JDBC3Statement.unused ( ) : SQLException
org/sqlite/jdbc3/JDBC3Statement.unused:(Ljava/sql/SQLException);

sqlite-jdbc-3.8.11.2.jar, JDBC4ResultSet.class
package org.sqlite.jdbc4
JDBC4ResultSet.unused ( ) : SQLException
org/sqlite/jdbc4/JDBC4ResultSet.unused:(Ljava/sql/SQLException);

sqlite-jdbc-3.8.11.2.jar, NativeDB.class
package org.sqlite.core
NativeDB.create_function ( String p1, Function p2 ) : int
org/sqlite/core_NativeDB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;)I
NativeDB.NativeDB ( )
org/sqlite/core_NativeDB."<init>";()V
NativeDB.value_bytes ( Function p1, int p2 ) : int
org/sqlite/core_NativeDB.value_bytes:(Lorg/sqlite/Function;I)I

sqlite-jdbc-3.8.11.2.jar, PrimaryKeyFinder.class
package org.sqlite.core
CoreDatabaseMetaData.PrimaryKeyFinder.CoreDatabaseMetaData.PrimaryKeyFinder ( CoreDatabaseMetaData table, String p2 )
org/sqlite/core/CoreDatabaseMetaData.PrimaryKeyFinder."<init>":(Lorg/sqlite/core/CoreDatabaseMetaData;Ljava/lang/String;)V
CoreDatabaseMetaData.PrimaryKeyFinder.getColumns ( ) : String[
org/sqlite/core/CoreDatabaseMetaData.PrimaryKeyFinder.getColumns ( )[]Ljava/lang/String;
CoreDatabaseMetaData.PrimaryKeyFinder.getName ( ) : String
org/sqlite/core/CoreDatabaseMetaData.PrimaryKeyFinder.getName():()Ljava/lang/String;

sqlite-jdbc-3.8.11.2.jar, SQLite.class
package org.sqlite
SQLite.main ( String[] p1 ) [static] : void
org/sqlite/SQLite.main:([Ljava/lang/String;)V
SQLite.SQLite ( )
org/sqlite/SQLite."<init>"():()V
SQLite.SQLite ( boolean p1 )
org/sqlite/SQLite."<init>":(Z)V
SQLite.SQLite ( int p1, int p2 )
org/sqlite/SQLite."<init>":(II)V
SQLite.SQLite ( int p1, int p2, boolean p3 )
org/sqlite/SQLite."<init>":(IIZ)V

sqlite-jdbc-3.8.11.2.jar, SQLiteConfig.class
package org.sqlite
SQLiteConfig.setBusyTimeout ( String milliseconds ) : void
org/sqlite/SQLiteConfig.setBusyTimeout:(Ljava/lang/String;)V

sqlite-jdbc-3.8.11.2.jar, SQLiteDataSource.class
package org.sqlite
SQLiteDataSource.setCouncChanges ( boolean enable ) : void
org/sqlite/SQLiteDataSource.setCouncChanges:(Z)V
SQLiteDataSource.setEnforceForeinKeys ( boolean enforce ) : void
org/sqlite/SQLiteDataSource.setEnforceForeinKeys:(Z)V

sqlite-jdbc-3.8.11.2.jar, SQLitePooledConnection.class
package org.sqlite.java
SQLitePooledConnection.SQLitePooledConnection ( Connection physicalConn )
org/sqlite/java/SQLitePooledConnection."<init>":(Ljava/sql/Connection;)V
Problems with Data Types, High Severity

**sqlite-jdbc-3.8.11.2.jar**

**package org.sqlite**

| [+ class SQLite | 1 |

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class has been removed. A client program may be interrupted by NoClassDefFoundError exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 12 (1%)

**SQLite.SQLi te ( )**
*This constructor is from 'SQLite' class.*

**SQLite.SQLi te ( int p1, int p2 )**
*This constructor is from 'SQLite' class.*

**SQLite.SQLi te ( int p1, int p2, boolean p3 )**
*This constructor is from 'SQLite' class.*

**SQLite.SQLi te ( boolean p1 )**
*This constructor is from 'SQLite' class.*

**SQLite._execute ( )**
*This method is from 'SQLite' class.*

**SQLite.entryPoint ( )**
*This method is from 'SQLite' class.*

**SQLite.getCPUP 1 ( )**
*This method is from 'SQLite' class.*

**SQLite.gp ( )**
*This method is from 'SQLite' class.*

**SQLite.heapStart ( )**
This method is from 'SQLite' class.

`SQLite.lookupSymbol` ( String p1 )

This method is from 'SQLite' class.

... 

[+] class `SQLiteConfig` 5

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Access level of field <code>busyTimeout</code> has been changed from public to private.</td>
<td>A client program may be interrupted by <code>IllegalAccessError</code> exception.</td>
</tr>
<tr>
<td>2</td>
<td>Field <code>dateClass</code> of type <code>SQLiteConfig.DateClass</code> has been removed from this class.</td>
<td>A client program may be interrupted by <code>NoSuchFieldError</code> exception.</td>
</tr>
<tr>
<td>3</td>
<td>Field <code>dateMultiplier</code> of type <code>long</code> has been removed from this class.</td>
<td>A client program may be interrupted by <code>NoSuchFieldError</code> exception.</td>
</tr>
<tr>
<td>4</td>
<td>Field <code>datePrecision</code> of type <code>SQLiteConfig.DatePrecision</code> has been removed from this class.</td>
<td>A client program may be interrupted by <code>NoSuchFieldError</code> exception.</td>
</tr>
<tr>
<td>5</td>
<td>Field <code>DateStringFormat</code> of type <code>java.lang.String</code> has been removed from this class.</td>
<td>A client program may be interrupted by <code>NoSuchFieldError</code> exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 48 (3.9%) 

`javax.SQLiteConnectionPoolDataSource.SQLiteConnectionPoolDataSource` ( `SQLiteConfig config` )

1st parameter 'config' of this method is of type 'SQLiteConfig'.

`SQLiteConfig.SQLiteConfig` ( )
This constructor is from 'SQLiteConfig' class.

SQLiteConfig.SQLiteConfig ( Properties prop )
This constructor is from 'SQLiteConfig' class.

SQLiteConfig.apply ( Connection conn )
This method is from 'SQLiteConfig' class.

SQLiteConfig.createConnection ( String url )
This method is from 'SQLiteConfig' class.

SQLiteConfig.enableCaseSensitiveLike ( boolean enable )
This method is from 'SQLiteConfig' class.

SQLiteConfig.enableCountChanges ( boolean enable )
This method is from 'SQLiteConfig' class.

SQLiteConfig.enableEmptyResultCallBacks ( boolean enable )
This method is from 'SQLiteConfig' class.

SQLiteConfig.enableFullColumnNames ( boolean enable )
This method is from 'SQLiteConfig' class.

SQLiteConfig.enableFullSync ( boolean enable )
This method is from 'SQLiteConfig' class.

...

[+] class SQLiteConfig.Pragma 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Field READ_UNCOMMITTED of type</td>
<td>A client program may be interrupted by NoSuchFieldError exception.</td>
</tr>
<tr>
<td>SQLiteConfig.Pragma has been removed</td>
<td></td>
</tr>
<tr>
<td>from this class.</td>
<td></td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)

SQLiteConfig.Pragma.getPragmaName ( )
This method is from 'SQLiteConfig.Pragma' class.

SQLiteConfig.Pragma.valueOf ( String name )
This method is from 'SQLiteConfig.Pragma' class.

SQLiteConfig.Pragma.values ( )
This method is from 'SQLiteConfig.Pragma' class.

SQLiteConfig.setPragma ( SQLiteConfig.Pragma pragma, String value )
1st parameter 'pragma' of this method is of type 'SQLiteConfig.Pragma'.

[+] class SQLiteConnection

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class became abstract.</td>
</tr>
<tr>
<td></td>
<td>A client program may be interrupted by InstantiationException exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 32 (2.6%)

core.CoreDatabaseMetaData.CoreDatabaseMetaData ( SQLiteConnection conn )
1st parameter 'conn' of this method is of type 'SQLiteConnection'.

core.CoreDatabaseMetaData.checkOpen ( )
This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.close ( )
This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.escape ( String val )
This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.finalize ( )
This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.getGeneratedKeys ( )
This abstract method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.quote ( String tableName )
This method is from 'SQLiteConnection' abstract class.

core.CorePreparedStatement.CorePreparedStatement ( SQLiteConnection conn, String sql )
1st parameter 'conn' of this method is of type 'SQLiteConnection'.

```java
core.CoreResultSet.CoreResultSet ( CoreStatement stmt )
1st parameter 'stmt' of this method has base type 'SQLiteConnection'.
```

```java
core.CoreStatement.CoreStatement ( SQLiteConnection c )
1st parameter 'c' of this method is of type 'SQLiteConnection'.
```

...
CoreConnection.getDriverVersion ( )
This method is from 'CoreConnection' abstract class.

CoreConnection.libversion ( )
This method is from 'CoreConnection' abstract class.

CoreConnection.setBusyTimeout ( int milliseconds )
This method is from 'CoreConnection' abstract class.

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class has been removed. A client program may be interrupted by NoClassDefFoundError exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 3 (0.2%)

```java
CoreDatabaseMetaData.PrimaryKeyFinder.CoreDatabaseMetaData.PrimaryKeyFinder ( CoreDatabaseMetaData table, String p2 )
This constructor is from 'CoreDatabaseMetaData.PrimaryKeyFinder' class.

CoreDatabaseMetaData.PrimaryKeyFinder.getColumns ( )
This method is from 'CoreDatabaseMetaData.PrimaryKeyFinder' class.

CoreDatabaseMetaData.PrimaryKeyFinder.getName ( )
This method is from 'CoreDatabaseMetaData.PrimaryKeyFinder' class.
```

[+] class `CoreResultSet` 2

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Type of field <code>maxRows</code> has been changed from <code>int</code> to <code>long</code>. A client program may be interrupted by NoSuchFieldError exception.</td>
</tr>
<tr>
<td>2</td>
<td>Field <code>db</code> of type <code>DB</code> has been removed from this class. A client program may be interrupted by NoSuchFieldError exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 19 (1.5%)

```java
CoreResultSet.CoreResultSet ( CoreStatement stmt )
Field 'stmt.rs' in 1st parameter 'stmt' of this method is of type 'CoreResultSet'.

CoreResultSet.addColumnIndexInCache ( String col, int index )
```
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkMeta ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.close ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.findColumnIndexInCache ( String col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.isOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.markCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreStatement.CoreStatement ( SQLiteConnection c )
Field 'this.rs' in the object of this method is of type 'CoreResultSet'.

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>from this class.</td>
<td>exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 11 (0.9%)

**CoreResultSet.CoreResultSet** (CoreStatement \textit{stmt})

1st parameter 'stmt' of this method is of type 'CoreStatement'.

**CoreStatement.CoreStatement** (SQLiteConnection \textit{c})

This constructor is from 'CoreStatement' abstract class.

**CoreStatement.checkOpen** ()

This method is from 'CoreStatement' abstract class.

**CoreStatement.exec** ()

This method is from 'CoreStatement' abstract class.

**CoreStatement.exec** (String \textit{sql})

This method is from 'CoreStatement' abstract class.

**CoreStatement.executeQuery** (String \textit{p1}, boolean \textit{p2})

This abstract method is from 'CoreStatement' abstract class.

**CoreStatement.internalClose** ()

This method is from 'CoreStatement' abstract class.

**DB.execute** (CoreStatement \textit{stmt}, Object[] \textit{vals})

1st parameter 'stmt' of this method is of type 'CoreStatement'.

**DB.prepare** (CoreStatement \textit{stmt})

1st parameter 'stmt' of this method is of type 'CoreStatement'.

**org.sqlite.jdbc3.JDBC3ResultSet.JDBC3ResultSet** (CoreStatement \textit{stmt})

1st parameter 'stmt' of this method is of type 'CoreStatement'.

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>class.</td>
<td>exception.</td>
</tr>
<tr>
<td><strong>2</strong></td>
<td>Abstract method <code>int create_function ( String, Function )</code> has been removed from this class.</td>
</tr>
<tr>
<td><strong>3</strong></td>
<td>Abstract method <code>long prepare ( String )</code> has been removed from this class.</td>
</tr>
<tr>
<td><strong>4</strong></td>
<td>Abstract method <code>int total_changes ()</code> has been removed from this class.</td>
</tr>
<tr>
<td><strong>5</strong></td>
<td>Abstract method <code>int value_bytes ( Function, int )</code> has been removed from this class.</td>
</tr>
</tbody>
</table>

[+] affected methods: 46 (3.7%)
DB.close ( )
   *This method is from 'DB' abstract class.*

DB.column_blob ( long p1, int p2 )
   *This abstract method is from 'DB' abstract class.*

DB.column_count ( long p1 )
   *This abstract method is from 'DB' abstract class.*

DB.column_decltype ( long p1, int p2 )
   *This abstract method is from 'DB' abstract class.*

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Type of field physicalConn has been changed from java.sql.Connection to org.sqlite.SQLiteConnection.</td>
<td>A client program may be interrupted by NoSuchFieldError exception.</td>
</tr>
<tr>
<td>2</td>
<td>Removed super-interface javax.sql.PooledConnection.</td>
<td>A client program may be interrupted by NoSuchMethodError exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)

SQLitePooledConnection.addConnectionEventListener ( ConnectionEventListener listener )
   *This method is from 'SQLitePooledConnection' class.*

SQLitePooledConnection.close ( )
   *This method is from 'SQLitePooledConnection' class.*

SQLitePooledConnection.getConnection ( )
   *This method is from 'SQLitePooledConnection' class.*
SQLitePooledConnection.removeConnectionEventListener (ConnectionEventListener listener)
    This method is from 'SQLitePooledConnection' class.

sqlite-jdbc-3.8.11.2.jar
package org.sqlite.jdbc3
[+] class JDBC3DatabaseMetaData 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field FK_NAMED_PATTERN of type java.util.regex.Pattern has been removed from this class.</td>
<td>A client program may be interrupted by NoSuchFieldError exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 170 (13.6%)

JDBC3DatabaseMetaData.JDBC3DatabaseMetaData (SQLiteConnection conn)
    This constructor is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.allProceduresAreCallable ( )
    This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.allTablesAreSelectable ( )
    This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.createStruct ( String t, Object[] attr )
    This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.dataDefinitionCausesTransactionCommit ( )
    This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.dataDefinitionIgnoredInTransactions ( )
    This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.deletesAreDetected ( int type )
This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.doesMaxRowSizeIncludeBlobs ( )
This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.finalize ( )
This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.getAttributes ( String c, String s, String t, String a )
This method is from 'JDBC3DatabaseMetaData' abstract class.

...
JDBC4Connection.createSQLXML()  
This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.createStatement(int rst, int rsc, int rsh)  
This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.getClientInfo()  
This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.getClientInfo(String name)  
This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.isClosed()  
This method is from 'JDBC4Connection' abstract class.

...  

[+] class JDBC4DatabaseMetaData  

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed super-interface java.sql.DatabaseMetaData. A client program may be interrupted by NoSuchMethodError exception.</td>
</tr>
</tbody>
</table>

[+] affected methods: 11 (0.9%)  

JDBC4DatabaseMetaData.JDBC4DatabaseMetaData(SQLiteConnection conn)  
This constructor is from 'JDBC4DatabaseMetaData' class.

JDBC4DatabaseMetaData.autoCommitFailureClosesAllResultSets()  
This method is from 'JDBC4DatabaseMetaData' class.

JDBC4DatabaseMetaData.generatedKeyAlwaysReturned()  
This method is from 'JDBC4DatabaseMetaData' class.

JDBC4DatabaseMetaData.getClientInfoProperties()  
This method is from 'JDBC4DatabaseMetaData' class.

JDBC4DatabaseMetaData.getFunctions(String catalog, String schemaPattern, String functionNamePattern)
This method is from 'JDBC4DatabaseMetaData' class.

`JDBC4DatabaseMetaData.getPseudoColumns(String catalog, String schemaPattern, String tableNamePattern, String columnNamePattern)`

This method is from 'JDBC4DatabaseMetaData' class.

`JDBC4DatabaseMetaData.getRowIdLifetime()`

This method is from 'JDBC4DatabaseMetaData' class.

`JDBC4DatabaseMetaData.getSchemas(String catalog, String schemaPattern)`

This method is from 'JDBC4DatabaseMetaData' class.

`JDBC4DatabaseMetaData.isWrapperFor(Class<?> iface)`

This method is from 'JDBC4DatabaseMetaData' class.

`JDBC4DatabaseMetaData.supportsStoredFunctionsUsingCallSyntax()`

This method is from 'JDBC4DatabaseMetaData' class.

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>exception.</td>
</tr>
</tbody>
</table>

```java
[+] DB.executeUpdate ( CoreStatement stmt, Object[] vals ) : int
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from int to long.</td>
</tr>
<tr>
<td></td>
<td>This method has been removed because the return type is part of the method signature. A client program may be interrupted by NoSuchMethodError exception.</td>
</tr>
</tbody>
</table>

```java
[+] DB.newSQLException ( int errorCode, String errorMessage ) [static] : SQLException
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from java.sql.SQLException to org.sqlite.SQLiteException.</td>
</tr>
<tr>
<td></td>
<td>This method has been removed because the return type is part of the method signature. A client program may be interrupted by NoSuchMethodError exception.</td>
</tr>
</tbody>
</table>

```java
[+] DB.prepare ( String p1 ) [abstract] : long
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from long to SafeStmtPtr.</td>
</tr>
<tr>
<td></td>
<td>This method has been removed because the return type is part of the method signature. A client program may be interrupted by</td>
</tr>
</tbody>
</table>
### DB.total_changes()

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>NoSuchMethodError exception.</td>
</tr>
</tbody>
</table>

#### Change

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <code>int</code> to <code>long</code>. This method has been removed because the return type is part of the method signature. A client program may be interrupted by <code>NoSuchMethodError</code> exception.</td>
</tr>
</tbody>
</table>

#### sqlite-jdbc-3.8.11.2.jar, JDBC.class

```java
package org.sqlite;

[+] JDBC.createConnection ( String url, Properties prop ) static : Connection
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <code>java.sql.Connection</code> to <code>SQLiteConnection</code>. This method has been removed because the return type is part of the method signature. A client program may be interrupted by <code>NoSuchMethodError</code> exception.</td>
</tr>
</tbody>
</table>

#### sqlite-jdbc-3.8.11.2.jar, NativeDB.class

```java
package org.sqlite.core;

[+] NativeDB.changes () : int
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>Return value type has been changed from <code>int</code> to <code>long</code>. This method has been removed because the return type is part of the method signature. A client program may be interrupted by <code>NoSuchMethodError</code> exception.</td>
</tr>
</tbody>
</table>

```java
[+] NativeDB.prepare ( String p1 ) : long
org/sqlite/core/NativeDB.prepare:(Ljava/lang/String;)J
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <code>long</code> to <code>SafeStmtPtr</code>. This method has been removed because the return type is part of the method signature. A client program may be interrupted by <code>NoSuchMethodError</code> exception.</td>
</tr>
</tbody>
</table>

```java
[+] NativeDB.total_changes ( ) : int
org/sqlite/core/NativeDB.total_changes:(J)
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <code>int</code> to <code>long</code>. This method has been removed because the return type is part of the method signature. A client program may be interrupted by <code>NoSuchMethodError</code> exception.</td>
</tr>
</tbody>
</table>

```java
sqlite-jdbc-3.8.11.2.jar, SQLiteDataSource.class
package org.sqlite
[+] SQLiteDataSource.getConnection ( String username, String password ) : Connection
org/sqlite.SQLiteDataSource.getConnection:(Ljava/lang/String;Ljava/lang/String;)Ljava/sql/Connection;
```
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from \texttt{java.sql.Connection} to \texttt{SQLiteConnection}. This method has been removed because the return type is part of the method signature. A client program may be interrupted by \texttt{NoSuchMethodError} exception.</td>
</tr>
</tbody>
</table>

Problems with Data Types, Medium Severity 1

\texttt{sqlite-jdbc-3.8.11.2.jar}

package \texttt{org.sqlite}

[+] class \texttt{SQLiteConnection} 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed super-class \texttt{jdbc4.JDBC4Connection}. Access of a client program to the fields or methods of the old super-class may be interrupted by \texttt{NoSuchFieldError} or \texttt{NoSuchMethodError} exceptions.</td>
</tr>
</tbody>
</table>

[+] affected methods: 32 (2.6%)

core.CoreDatabaseMetaData.CoreDatabaseMetaData ( 
\texttt{SQLiteConnection conn} )

1st parameter 'conn' of this method is of type 'SQLiteConnection'.

core.CoreDatabaseMetaData.checkOpen ( )

This method is from 'SQLiteConnection' abstract class.
core.CoreDatabaseMetaData.close ( )
   This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.escape ( String val )
   This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.finalize ( )
   This method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.getGeneratedKeys ( )
   This abstract method is from 'SQLiteConnection' abstract class.

core.CoreDatabaseMetaData.quote ( String tableName )
   This method is from 'SQLiteConnection' abstract class.

core.CorePreparedStatement.CorePreparedStatement ( SQLiteConnection conn, String sql )
   1st parameter 'conn' of this method is of type 'SQLiteConnection'.

core.CoreResultSet.CoreResultSet ( CoreStatement stmt )
   1st parameter 'stmt' of this method has base type 'SQLiteConnection'.

core.CoreStatement.CoreStatement ( SQLiteConnection c )
   1st parameter 'c' of this method is of type 'SQLiteConnection'.

...
[+] affected methods: 48 (3.9%)

```java
javax.SQLiteConnectionPoolDataSource.SQLiteConnectionPoolDataSource ( SQLiteConfig config )
   1st parameter 'config' of this method is of type 'SQLiteConfig'.

SQLiteConfig.SQLiteConfig ()
   This constructor is from 'SQLiteConfig' class.

SQLiteConfig.SQLiteConfig ( Properties prop )
   This constructor is from 'SQLiteConfig' class.

SQLiteConfig.apply ( Connection conn )
   This method is from 'SQLiteConfig' class.

SQLiteConfig.createConnection ( String url )
   This method is from 'SQLiteConfig' class.

SQLiteConfig.enableCaseSensitiveLike ( boolean enable )
   This method is from 'SQLiteConfig' class.

SQLiteConfig.enableCountChanges ( boolean enable )
   This method is from 'SQLiteConfig' class.

SQLiteConfig.enableEmptyResultCallBacks ( boolean enable )
   This method is from 'SQLiteConfig' class.

SQLiteConfig.enableFullColumnNames ( boolean enable )
   This method is from 'SQLiteConfig' class.

SQLiteConfig.enableFullSync ( boolean enable )
   This method is from 'SQLiteConfig' class.
```

```java
sqlite-jdbc-3.8.11.2.jar
package org.sqlite.core
[+] class CorePreparedStatement 3
```
<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method <code>CorePreparedStatement.finalize()</code> has been moved up type hierarchy to <code>java/lang/Object.finalize():V</code></td>
<td>Method <code>java/lang/Object.finalize():V</code> will be called instead of <code>CorePreparedStatement.finalize()</code> in a client program.</td>
</tr>
<tr>
<td>2</td>
<td>Method <code>CorePreparedStatement.getUpdateCount()</code> has been moved up type hierarchy to <code>int org.sqlite.jdbc3.JDBC3Statement.getUpdateCount()</code></td>
<td>Method <code>org.sqlite.jdbc3.JDBC3Statement.getUpdateCount()</code> will be called instead of <code>CorePreparedStatement.getUpdateCount()</code> in a client program.</td>
</tr>
<tr>
<td>3</td>
<td>Method <code>org.sqlite.jdbc3.JDBC3Statement.clearBatch()</code> has been overridden by <code>void CorePreparedStatement.clearBatch()</code></td>
<td>Method <code>CorePreparedStatement.clearBatch()</code> will be called instead of <code>org.sqlite.jdbc3.JDBC3Statement.clearBatch()</code> in a client program.</td>
</tr>
</tbody>
</table>

[+] affected methods: 3 (0.2%)

`CorePreparedStatement.finalize()`  
Method `java/lang/Object.finalize():V` will be called instead of this method in a client program.

`CorePreparedStatement.getUpdateCount()`  
Method `org.sqlite.jdbc3.JDBC3Statement.getUpdateCount()` will be called instead of this method in a client program.

`org.sqlite.jdbc3.JDBC3Statement.clearBatch()`  
Method `CorePreparedStatement.clearBatch()` will be called instead of this method in a client program.

[+] class NativeDB 3
<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method <code>int NativeDB.changes ()</code> has been moved up type hierarchy to <code>long DB.changes ()</code>.</td>
<td>Method <code>DB.changes ()</code> will be called instead of <code>NativeDB.changes ()</code> in a client program.</td>
</tr>
<tr>
<td>2</td>
<td>Method <code>long NativeDB.prepare ( String )</code> has been moved up type hierarchy to SafeStmtPtr <code>DB.prepare ( String )</code>.</td>
<td>Method <code>DB.prepare ( String )</code> will be called instead of <code>NativeDB.prepare ( String )</code> in a client program.</td>
</tr>
<tr>
<td>3</td>
<td>Method <code>int NativeDB.total_changes ()</code> has been moved up type hierarchy to <code>long DB.total_changes ()</code>.</td>
<td>Method <code>DB.total_changes ()</code> will be called instead of <code>NativeDB.total_changes ()</code> in a client program.</td>
</tr>
</tbody>
</table>

`NativeDB.changes ()`

*Method 'DB.changes ()' will be called instead of this method in a client program.*

`NativeDB.prepare ( String p1 )`

*Method 'DB.prepare ( String )' will be called instead of this method in a client program.*

`NativeDB.total_changes ()`

*Method 'DB.total_changes ()' will be called instead of this method in a client program.*

sqlite-jdbc-3.8.11.2.jar
package org.sqlite.jdbc3
[+] class JDBC3Connection 7
<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
</table>
| 1 | Superclass has been changed from org.sqlite.core.CoreConnection to org.sqlite.SQLiteConnection. | 1) Access of a client program to the fields or methods of the old super-class may be interrupted by NoSuchFieldError or NoSuchMethodError exceptions.  
2) A static field from a super-interface of a client class may hide a field (with the same name) inherited from new super-class and cause IncompatibleClassChangeError exception. |
<p>| 2 | Method void JDBC3Connection.commit () has been moved up type hierarchy to void org.sqlite.SQLiteConnection.commit (). | Method org.sqlite.SQLiteConnection.commit () will be called instead of JDBC3Connection.commit () in a client program. |
| 3 | Method boolean JDBC3Connection.getAutoCommit () has been moved up type hierarchy to boolean org.sqlite.SQLiteConnection.getAutoCommit (). | Method org.sqlite.SQLiteConnection.getAutoCommit () will be called instead of JDBC3Connection.getAutoCommit () in a client program. |
| 4 | Method int JDBC3Connection.getTransactionIsolation () has been moved up type hierarchy to int org.sqlite.SQLiteConnection.getTransactionIsolation (). | Method org.sqlite.SQLiteConnection.getTransactionIsolation () will be called instead of JDBC3Connection.getTransactionIsolation () in a client program. |</p>
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>getTransactionIsolation()</td>
<td>nsactionIsolation() in a client program.</td>
</tr>
</tbody>
</table>

5 Method void JDBC3Connection.rollback() has been moved up type hierarchy to void org.sqlite.SQLiteConnection.rollback().

Method org.sqlite.SQLiteConnection.rollback() will be called instead of JDBC3Connection.rollback() in a client program.

6 Method void JDBC3Connection.setAutoCommit(boolean) has been moved up type hierarchy to void org.sqlite.SQLiteConnection.setAutoCommit(boolean).

Method org.sqlite.SQLiteConnection.setAutoCommit(boolean) will be called instead of JDBC3Connection.setAutoCommit(boolean) in a client program.

7 Method void JDBC3Connection.setTransactionIsolation(int) has been moved up type hierarchy to void org.sqlite.SQLiteConnection.setTransactionIsolation(int).

Method org.sqlite.SQLiteConnection.setTransactionIsolation(int) will be called instead of JDBC3Connection.setTransactionIsolation(int) in a client program.

[+] affected methods: 35 (2.8%)

**JDBC3Connection.JDBC3Connection**( String *url*, String *fileName*, Properties *prop*)  
This constructor is from 'JDBC3Connection' abstract class.

**JDBC3Connection.clearWarnings**( )  
This method is from 'JDBC3Connection' abstract class.

**JDBC3Connection.commit**( )  
Method 'org.sqlite.SQLiteConnection.commit ()' will be called instead of this method in a client program.
**JDBC3Connection.createStatement()**
This method is from 'JDBC3Connection' abstract class.

**JDBC3Connection.createStatement(int rsType, int rsConcurr)**
This method is from 'JDBC3Connection' abstract class.

**JDBC3Connection.createStatement(int p1, int p2, int p3)**
This abstract method is from 'JDBC3Connection' abstract class.

**JDBC3Connection.createStruct(String t, Object[] attr)**
This method is from 'JDBC3Connection' abstract class.

**JDBC3Connection.setAutoCommit()**
Method 'org.sqlite.SQLiteConnection.setAutoCommit()' will be called instead of this method in a client program.

**JDBC3Connection.getCatalog()**
This method is from 'JDBC3Connection' abstract class.

**JDBC3Connection.getHoldability()**
This method is from 'JDBC3Connection' abstract class.

... 

[+] class JDBC3PreparedStatement

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
</table>
| 1      | Method `boolean JDBC3Statement.execute(String, int)` has been overridden by `boolean JDBC3PreparedStatement.execute(String, int)`.
|        | Method `JDBC3PreparedStatement.execute(String, int)` will be called instead of `JDBC3Statement.execute(String, int)` in a client program. |
| 2      | Method `boolean JDBC3Statement.execute(String, int[])` has been overridden by `boolean JDBC3PreparedStatement.execute(String, int[])`.
|        | Method `JDBC3PreparedStatement.execute(String, int[])` will be called instead of `JDBC3Statement.execute(String, int[])` in a client program. |
| 3      | Method `boolean` | Method |


<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JDBC3Statement.execute</strong> (String, String[]) has been overridden by</td>
<td><strong>JDBC3PreparedStatement.execute</strong> (String, String[]) will be called</td>
</tr>
<tr>
<td>boolean <strong>JDBC3PreparedStatement.execute</strong> (String, String[])</td>
<td>instead of <strong>JDBC3Statement.execute</strong> (String, String[]) in a client</td>
</tr>
<tr>
<td></td>
<td>program.</td>
</tr>
<tr>
<td><strong>4</strong></td>
<td></td>
</tr>
<tr>
<td>Method int <strong>JDBC3Statement.executeUpdate</strong> (String, int[]) has</td>
<td>Method <strong>JDBC3PreparedStatement.executeUpdate</strong> (String, int[]) will</td>
</tr>
<tr>
<td>been overridden by int <strong>JDBC3PreparedStatement.executeUpdate</strong></td>
<td>be called instead of <strong>JDBC3Statement.executeUpdate</strong> (String, int[])</td>
</tr>
<tr>
<td>(String, int[]).</td>
<td>in a client program.</td>
</tr>
<tr>
<td><strong>5</strong></td>
<td></td>
</tr>
<tr>
<td>Method int <strong>JDBC3Statement.executeUpdate</strong> (String, int[]) has</td>
<td>Method <strong>JDBC3PreparedStatement.executeUpdate</strong> (String, int[]) will</td>
</tr>
<tr>
<td>been overridden by int <strong>JDBC3PreparedStatement.executeUpdate</strong></td>
<td>be called instead of <strong>JDBC3Statement.executeUpdate</strong> (String, int[])</td>
</tr>
<tr>
<td>(String, int[]).</td>
<td>in a client program.</td>
</tr>
<tr>
<td><strong>6</strong></td>
<td></td>
</tr>
<tr>
<td>Method int <strong>JDBC3Statement.executeUpdate</strong> (String, String[])</td>
<td>Method <strong>JDBC3PreparedStatement.executeUpdate</strong> (String, String[]) will</td>
</tr>
<tr>
<td>has been overridden by int <strong>JDBC3PreparedStatement.executeUpdate</strong></td>
<td>be called instead of <strong>JDBC3Statement.executeUpdate</strong> (String, String[])</td>
</tr>
<tr>
<td>(String, String[]).</td>
<td>in a client program.</td>
</tr>
</tbody>
</table>

[+] affected methods: 6 (0.5%)

**JDBC3Statement.executeUpdate** (String sql, int[] colinds)

*Method 'JDBC3PreparedStatement.executeUpdate (String, int[])' will be called instead of this method in a client program.*

**JDBC3Statement.executeUpdate** (String sql, String[] colnames)
Method 'JDBC3PreparedStatement.execute (String, String[])' will be called instead of this method in a client program.

**JDBC3Statement.execute** (String sql, int autokeys)
Method 'JDBC3PreparedStatement.execute (String, int)' will be called instead of this method in a client program.

**JDBC3Statement.executeUpdate** (String sql, int autokeys)
Method 'JDBC3PreparedStatement.executeUpdate (String, int)' will be called instead of this method in a client program.

**JDBC3ResultSet**

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method void JDBC3ResultSet.finalize () has been moved up type hierarchy to java/lang/Object.finalize():V. Method java/lang/Object.finalize():V will be called instead of JDBC3ResultSet.finalize () in a client program.</td>
</tr>
</tbody>
</table>

[+] affected methods: 1 (0.1%)

**JDBC3ResultSet.finalize ()**
Method 'java/lang/Object.finalize():V' will be called instead of this method in a client program.

[+] class **JDBC3Statement**

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method void JDBC3Statement.finalize</td>
</tr>
</tbody>
</table>

[+] class **JDBC3ResultSet**

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method void JDBC3ResultSet.finalize</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>() has been moved up type hierarchy to java/lang/Object.finalize():</td>
<td>() will be called instead of JDBC3Statement.finalize () in a client program.</td>
</tr>
</tbody>
</table>

[+] affected methods: 1 (0.1%)

JDBC3Statement.finalize ( )
Method 'java/lang/Object.finalize():V' will be called instead of this method in a client program.

sqlite-jdbc-3.8.11.2.jar
package org.sqlite.jdbc4
[+] class JDBC4Connection

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method DatabaseMetaData JDBC4Connection.getMetaData () has been moved up type hierarchy to DatabaseMetaData org.sqlite.SQLiteConnection.getMetaData ().</td>
</tr>
<tr>
<td></td>
<td>Method org.sqlite.SQLiteConnection.getMetaData () will be called instead of JDBC4Connection.getMetaData () in a client program.</td>
</tr>
</tbody>
</table>

[+] affected methods: 1 (0.1%)

JDBC4Connection.getMetaData ( )
Method 'org.sqlite.SQLiteConnection.getMetaData ()' will be called instead of this method in a client program.

[+] class JDBC4PreparedStatement

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method void JDBC4PreparedStatement JDBC4Statement.closeO</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>t.closeOnCompletion()</td>
<td>nCompletion() will be called instead of JDBC4PreparedStatement.closeOnCompletion() in a client program.</td>
</tr>
<tr>
<td>2</td>
<td>Method boolean JDBC4PreparedStatement.isCloseOnCompletion() has been moved up type hierarchy to boolean JDBC4Statement.isCloseOnCompletion(). Method JDBC4Statement.isCloseOnCompletion() will be called instead of JDBC4PreparedStatement.isCloseOnCompletion() in a client program.</td>
</tr>
<tr>
<td>+ affected methods: 2 (0.2%)</td>
<td></td>
</tr>
</tbody>
</table>

**JDBC4PreparedStatement.closeOnCompletion()**

Method 'JDBC4Statement.closeOnCompletion()' will be called instead of this method in a client program.

**JDBC4PreparedStatement.isCloseOnCompletion()**

Method 'JDBC4Statement.isCloseOnCompletion()' will be called instead of this method in a client program.

[+] class JDBC4ResultSet

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method void org.sqlite.core.CoreResultSet.close() has been overridden by void JDBC4ResultSet.close(). Method JDBC4ResultSet.close() will be called instead of org.sqlite.core.CoreResultSet.close() in a client program.</td>
</tr>
<tr>
<td>+ affected methods: 1 (0.1%)</td>
<td></td>
</tr>
</tbody>
</table>

**org.sqlite.core.CoreResultSet.close()**

Method 'JDBC4ResultSet.close()' will be called instead of this method in a client program.
[+] class JDBC4Statement

<table>
<thead>
<tr>
<th>Method</th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.sqlite.jdbc3.JDBC3Statement.close() has been overridden by void JDBC4Statement.close()</td>
<td>Method JDBC4Statement.close() will be called instead of org.sqlite.jdbc3.JDBC3Statement.close() in a client program.</td>
<td></td>
</tr>
</tbody>
</table>

[+] affected methods: 1 (0.1%)

org.sqlite.jdbc3.JDBC3Statement.close()
Method 'JDBC4Statement.close()' will be called instead of this method in a client program.

to the top

Problems with Methods, Low Severity

sqlite-jdbc-3.8.11.2.jar, JDBC3Connection.class
package org.sqlite.jdbc3
[+] JDBC3Connection.isReadOnly() : boolean

<table>
<thead>
<tr>
<th>Method</th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
<td></td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, JDBC3DatabaseMetaData.class
package org.sqlite.jdbc3
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added java.sql.SQLException exception thrown.</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added java.sql.SQLException exception thrown.</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added java.sql.SQLException exception thrown.</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>java.sql.SQLException exception thrown.</td>
<td>be interrupted by added exception.</td>
</tr>
</tbody>
</table>

**sqlite-jdbc-3.8.11.2.jar, JDBC3ResultSet.class**

package org.sqlite.jdbc3

[+] JDBC3ResultSet.clearWarnings ( ) : void

org/sqlite/jdbc3/JDBC3ResultSet.clearWarnings:()V

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
<td></td>
</tr>
</tbody>
</table>

[+] JDBC3ResultSet.getColumnDisplaySize ( int col ) : int

org/sqlite/jdbc3/JDBC3ResultSet.getColumnDisplaySize:(I)I

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
<td></td>
</tr>
</tbody>
</table>

[+] JDBC3ResultSet.getConcurrency ( ) : int

org/sqlite/jdbc3/JDBC3ResultSet.getConcurrency:()I

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>JDBC3ResultSet.getCursorName() : String</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
<tr>
<td>JDBC3ResultSet.getFetchSize() : int</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
<tr>
<td>JDBC3ResultSet.getMetaData() : ResultSetMetaData</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

---

**Change**

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>JDBC3ResultSet.getRow() : int</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

```java
[+] JDBC3ResultSet.getSchemaName ( int col ) : String
org/sqlite/jdbc3/JDBC3ResultSet.getSchemaName:(I)Ljava/lang/String;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

```java
[+] JDBC3ResultSet.getType ( ) : int
org/sqlite/jdbc3/JDBC3ResultSet.getType:()I
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

```java
[+] JDBC3ResultSet.getWarnings ( ) : SQLWarning
org/sqlite/jdbc3/JDBC3ResultSet.getWarnings:()Ljava/sql/SQLWarning;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException A client program may change behavior</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>exception thrown.</td>
<td>because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

- **[+]** `JDBC3ResultSet.isAfterLast()` : boolean
  - `org/sqlite/jdbc3/JDBC3ResultSet.isAfterLast:()Z`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
</tr>
<tr>
<td></td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

- **[+]** `JDBC3ResultSet.isBeforeFirst()` : boolean
  - `org/sqlite/jdbc3/JDBC3ResultSet.isBeforeFirst:()Z`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
</tr>
<tr>
<td></td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

- **[+]** `JDBC3ResultSet.isCaseSensitive(int col)` : boolean
  - `org/sqlite/jdbc3/JDBC3ResultSet.isCaseSensitive:(I)Z`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
</tr>
<tr>
<td></td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>handle it.</td>
<td></td>
</tr>
</tbody>
</table>

[+] **JDBC3ResultSet.isCurrency** (int col) : boolean

```java
org/sqlite/jdbc3/JDBC3ResultSet.isCurrency:(I)Z
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

[+] **JDBC3ResultSet.isDefinitelyWritable** (int col) : boolean

```java
org/sqlite/jdbc3/JDBC3ResultSet.isDefinitelyWritable:(I)Z
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

[+] **JDBC3ResultSet.isFirst** () : boolean

```java
org/sqlite/jdbc3/JDBC3ResultSet.isFirst:()Z
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

[+] **JDBC3ResultSet.isReadOnly** (int col) : boolean

```java
org/sqlite/jdbc3/JDBC3ResultSet.isReadOnly:(I)Z
```
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.

**[+] JDBC3ResultSet.isSearchable ( int col ) : boolean**

org/sqlite/jdbc3/JDBC3ResultSet.isSearchable:(I)Z

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.

**[+] JDBC3ResultSet.isWritable ( int col ) : boolean**

org/sqlite/jdbc3/JDBC3ResultSet.isWritable:(I)Z

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.

**[+] JDBC3ResultSet.rowDeleted ( ) : boolean**

org/sqlite/jdbc3/JDBC3ResultSet.rowDeleted:(I)Z

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

A client program may change behavior because the removed exception will not be
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td><strong>client will not catch and handle it.</strong></td>
</tr>
</tbody>
</table>

**sqlite-jdbc-3.8.11.2.jar, JDBC4Connection.class**

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Removed java.sql.SQLException exception thrown.</strong> A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td><strong>Removed java.sql.SQLException exception thrown.</strong> A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

**sqlite-jdbc-3.8.11.2.jar, JDBC4DatabaseMetaData.class**

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Added java.lang.ClassCastException exception thrown.</strong> A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td><strong>Removed java.sql.SQLException exception thrown.</strong> A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

`jdbc4DatabaseMetaData.unwrap(Class<T> iface) : T`  
`org/sqlite/jdbc4/JDBC4DatabaseMetaData.unwrap:(Ljava/lang/Class;)Ljava/lang/Object;`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added <code>java.lang.ClassCastException</code> exception thrown. A client program may be interrupted by added exception.</td>
</tr>
<tr>
<td>2</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, `JDBC4ResultSet.class`  
`jdbc4ResultSet.isWrapperFor(Class<*> iface) : boolean`  
`org/sqlite/jdbc4/JDBC4ResultSet.isWrapperFor:(Ljava/lang/Class;);Z`
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>handle it.</td>
<td></td>
</tr>
</tbody>
</table>

```java
JDBC4ResultSet.unwrap ( Class<T> iface ) : T
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added java.lang.ClassCastException exception thrown.</td>
<td>A client program may be interrupted by added exception.</td>
</tr>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, JDBC4Statement.class

package org.sqlite.jdbc4

```java
JDBC4Statement.isClosed () : boolean
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

```java
JDBC4Statement.isWrapperFor ( Class<?>) : boolean
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException</td>
<td>A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>exception thrown.</td>
<td>exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

`JDBC4Statement.unwrap` (Class<T> iface) : T

org/sqlite/jdbc4/JDBC4Statement.unwrap:(Ljava/lang/Class;)Ljava/lang/Object;

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added java.lang.ClassCastException exception thrown. A client program may be interrupted by added exception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Removed java.sql.SQLException exception thrown. A client program may change behavior because the removed exception will not be thrown any more and client will not catch and handle it.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, NativeDB.class
package org.sqlite.core

[+] NativeDB.backup (String p1, String p2, DB.ProgressObserver p3) : int

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method became non-synchronized. A multi-threaded client program may change behavior.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added java.sql.SQLException A client program may be interrupted by</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>exception thrown.</td>
<td>added exception.</td>
</tr>
</tbody>
</table>

### sqlite-jdbc-3.8.11.2.jar, SQLiteConfig.class

**package** org.sqlite

[+][1] SQLiteConfig.setDatePrecision ( String datePrecision ) : void

org/sqlite/SQLiteConfig.setDatePrecision:(Ljava/lang/String;)V

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>

### sqlite-jdbc-3.8.11.2.jar, SQLiteJDBCLoader.class

**package** org.sqlite

[+][1] SQLiteJDBCLoader.initialize () [static] : boolean

org/sqlite/SQLiteJDBCLoader.initialize:()Z

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Method became synchronized.</td>
</tr>
</tbody>
</table>

[to the top](#)

### Other Changes in Data Types 118

**sqlite-jdbc-3.8.11.2.jar**

**package** org.sqlite

[+][1] class **Function**
1 | Field FLAG_DETERMINISTIC has been added to this class. | No effect.

[+] affected methods: 30 (2.4%)

```plaintext
core.DB.value_blob ( Function p1, int p2 )
  1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_double ( Function p1, int p2 )
  1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_int ( Function p1, int p2 )
  1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_long ( Function p1, int p2 )
  1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_text ( Function p1, int p2 )
  1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_type ( Function p1, int p2 )
  1st parameter 'p1' of this abstract method is of type 'Function'.

core.NativeDB.value_blob ( Function p1, int p2 )
  1st parameter 'p1' of this method is of type 'Function'.

core.NativeDB.value_double ( Function p1, int p2 )
  1st parameter 'p1' of this method is of type 'Function'.

core.NativeDB.value_int ( Function p1, int p2 )
  1st parameter 'p1' of this method is of type 'Function'.

core.NativeDB.value_long ( Function p1, int p2 )
  1st parameter 'p1' of this method is of type 'Function'.

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field <strong>APPLICATION_ID</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>2</td>
<td>Field <strong>DEFER_FOREIGN_KEYS</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>3</td>
<td>Field <strong>HEXKEY_MODE</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>4</td>
<td>Field <strong>JDBC_EXPILICT_READONLY</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>5</td>
<td>Field <strong>LEGACY_ALTER_TABLE</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>6</td>
<td>Field <strong>LIMIT_ATTACHED</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>7</td>
<td>Field <strong>LIMIT_COLUMN</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>8</td>
<td>Field <strong>LIMIT_COMPOUND_SELECT</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>9</td>
<td>Field <strong>LIMIT_EXPR_DEPTH</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>10</td>
<td>Field</td>
</tr>
<tr>
<td></td>
<td>Change</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>LIMIT_FUNCTION_ARG has been added to this class.</td>
</tr>
<tr>
<td>11</td>
<td>Field LIMIT_LENGTH has been added to this class.</td>
</tr>
<tr>
<td>12</td>
<td>Field LIMITLIKE_PATTERNLLENGTH has been added to this class.</td>
</tr>
<tr>
<td>13</td>
<td>Field LIMIT_PAGE_COUNT has been added to this class.</td>
</tr>
<tr>
<td>14</td>
<td>Field LIMIT_SQL_LENGTH has been added to this class.</td>
</tr>
<tr>
<td>15</td>
<td>Field LIMIT_TRIGGERDEPTH has been added to this class.</td>
</tr>
<tr>
<td>16</td>
<td>Field LIMIT_VARIABLENUMBER has been added to this class.</td>
</tr>
<tr>
<td>17</td>
<td>Field LIMIT_VDBE_OP has been added to this class.</td>
</tr>
<tr>
<td>18</td>
<td>Field LIMIT_WORKER_THREADS has been added to this class.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>19</strong></td>
<td>Field <strong>MMAP_SIZE</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td><strong>20</strong></td>
<td>Field <strong>PASSWORD</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td><strong>21</strong></td>
<td>Field <strong>READ_UNCOMMITTED</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td><strong>22</strong></td>
<td>Field <strong>SECURE_DELETE</strong> has been added to this class. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)  

**SQLiteConfig.Pragma.getPragmaName** ( )  
This method is from 'SQLiteConfig.Pragma' class.

**SQLiteConfig.Pragma.valueOf** ( String name )  
This method is from 'SQLiteConfig.Pragma' class.

**SQLiteConfig.Pragma.values** ( )  
This method is from 'SQLiteConfig.Pragma' class.

**SQLiteConfig.setPragma** ( SQLiteConfig.Pragma pragma, String value )  
1st parameter 'pragma' of this method is of type 'SQLiteConfig.Pragma'.

[+] class **SQLiteConfig.TransactionMode** 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>Field <strong>DEFERRED</strong> has been added to this class. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 6 (0.5%)
SQLiteConfig.getTransactionMode ( )

Return value of this method is of type 'SQLiteConfig.TransactionMode'.

SQLiteConfig.setTransactionMode ( SQLiteConfig.TransactionMode transactionMode )

1st parameter 'transactionMode' of this method is of type 'SQLiteConfig.TransactionMode'.

SQLiteConfig.TransactionMode.getMode ( String mode )

This method is from 'SQLiteConfig.TransactionMode' class.

SQLiteConfig.TransactionMode.getValue ( )

This method is from 'SQLiteConfig.TransactionMode' class.

SQLiteConfig.TransactionMode.valueOf ( String name )

This method is from 'SQLiteConfig.TransactionMode' class.

SQLiteConfig.TransactionMode.values ( )

This method is from 'SQLiteConfig.TransactionMode' class.

[+] class SQLiteErrorCode 76

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field SQLITE_ABORT_ROLLBACK has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>2</td>
<td>Field SQLITE_AUTH_USER has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>3</td>
<td>Field SQLITE_BUSY_RECOVERY has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>4</td>
<td>Field SQLITE_BUSY_SNAPSHOT has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Field SQLITE_BUSY_TIMEOUT has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Field SQLITE_CANTOPEN_CONNPATH has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Field SQLITE_CANTOPEN_DIRTYWAL has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Field SQLITE_CANTOPEN_FULLPATH has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Field SQLITE_CANTOPEN_ISR has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Field SQLITE_CANTOPEN_NOTEMPDIR has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Field SQLITE_CANTOPEN_SYMLINK has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Field SQLITE_CONSTRAINT_CHECK has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Field SQLITE_CONSTRAINT_C</td>
<td></td>
</tr>
</tbody>
</table>

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OMMITHOOK</strong> has been added to this class.</td>
<td></td>
</tr>
<tr>
<td>14 Field <strong>SQLITE_CONSTRAINT_DATETYPE</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>15 Field <strong>SQLITE_CONSTRAINT_FOREIGNKEY</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>16 Field <strong>SQLITE_CONSTRAINT_FUNCTION</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>17 Field <strong>SQLITE_CONSTRAINT_NOTNULL</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>18 Field <strong>SQLITE_CONSTRAINT_PINNED</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>19 Field <strong>SQLITE_CONSTRAINT_PRIMARYKEY</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>20 Field <strong>SQLITE_CONSTRAINT_ROWID</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>21 Field <strong>SQLITE_CONSTRAINT_TRIGGER</strong> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>22 Field SQLITE_CONSTRAINT_UNIQUE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>23 Field SQLITE_CONSTRAINT_VTAB has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>24 Field SQLITE_CORRUPT_INDEX has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>25 Field SQLITE_CORRUPT_SEQUENCE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>26 Field SQLITE_CORRUPT_VTAB has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>27 Field SQLITE_ERROR_MISSING_COLLSEQ has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>28 Field SQLITE_ERROR_RETRY has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>29 Field SQLITE_ERROR_SNAPSHOT HOT has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>30 Field SQLITE_IOERR_ACCESS</td>
<td>No effect.</td>
</tr>
<tr>
<td></td>
<td>Change</td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td>has been added to this class.</td>
</tr>
<tr>
<td>31</td>
<td>Field SQLITE_IOERR_AUTH has been added to this class.</td>
</tr>
<tr>
<td>32</td>
<td>Field SQLITE_IOERR_BEGIN_ATOMIC has been added to this class.</td>
</tr>
<tr>
<td>33</td>
<td>Field SQLITE_IOERR_BLOCKED has been added to this class.</td>
</tr>
<tr>
<td>34</td>
<td>Field SQLITE_IOERR_CHECKRESERVEDLOCK has been added to this class.</td>
</tr>
<tr>
<td>35</td>
<td>Field SQLITE_IOERR_CLOSE has been added to this class.</td>
</tr>
<tr>
<td>36</td>
<td>Field SQLITE_IOERR_COMMIT_ATOMIC has been added to this class.</td>
</tr>
<tr>
<td>37</td>
<td>Field SQLITE_IOERR_CONVPATH has been added to this class.</td>
</tr>
<tr>
<td>38</td>
<td>Field SQLITE_IOERR_CORRUPTFS has been added</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_DATA has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_DELETE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_DELETE _NOENT has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_DIR_CLOSE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_DIR_FSYNC has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_FSTAT has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_FSYNC has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_GETTEMP PATH has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>47</td>
<td>Field <code>SQLITE_IOERR_LOCK</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>48</td>
<td>Field <code>SQLITE_IOERR_MMAP</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>49</td>
<td>Field <code>SQLITE_IOERR_NOMEM</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>50</td>
<td>Field <code>SQLITE_IOERR_RDLOCK</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>51</td>
<td>Field <code>SQLITE_IOERR_READ</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>52</td>
<td>Field <code>SQLITE_IOERR_ROLLBACK_ATOMIC</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>53</td>
<td>Field <code>SQLITE_IOERR_SEEK</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>54</td>
<td>Field <code>SQLITE_IOERR_SHMLOCK</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>55</td>
<td>Field <code>SQLITE_IOERR_SHMMA</code> No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>P has been added to this class.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_IOERR_SHMOPEN has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_SHMSIZE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_SHORT_READ has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_TRUNCATE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_UNLOCK has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_VNODE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_IOERR_WRITE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_LOCKED_SHAREDCACHE has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td></td>
<td>Change</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>64</td>
<td>Field SQLITE_LOCKED_VTAB has been added to this class.</td>
</tr>
<tr>
<td>65</td>
<td>Field SQLITE_NOTICE has been added to this class.</td>
</tr>
<tr>
<td>66</td>
<td>Field SQLITE_NOTICE_RECOVER.Rollback has been added to this class.</td>
</tr>
<tr>
<td>67</td>
<td>Field SQLITE_NOTICE_RECOVER.Wal has been added to this class.</td>
</tr>
<tr>
<td>68</td>
<td>Field SQLITE_OK_LOAD_PERMANENTLY has been added to this class.</td>
</tr>
<tr>
<td>69</td>
<td>Field SQLITE_READONLY.CANTINIT has been added to this class.</td>
</tr>
<tr>
<td>70</td>
<td>Field SQLITE_READONLY.CANTLOCK has been added to this class.</td>
</tr>
<tr>
<td>71</td>
<td>Field SQLITE_READONLY.DBMOVED has been added to this class.</td>
</tr>
<tr>
<td>72</td>
<td>Field SQLITE_READONLY.DIR</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>ECTORY has been added to this class.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_READONLY_RECOVERY has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_READONLY_ROLLBACK has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_WARNING has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Field SQLITE_WARNING_AUTOINDEX has been added to this class.</td>
<td>No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)

**SQLiteErrorCode.getErrorCode** (int errorCode)
*This method is from 'SQLiteErrorCode' class.*

**SQLiteErrorCode.toString** ( )
*This method is from 'SQLiteErrorCode' class.*

**SQLiteErrorCode.valueOf** (String name)
*This method is from 'SQLiteErrorCode' class.*

**SQLiteErrorCode.values** ( )
*This method is from 'SQLiteErrorCode' class.*

**sqlite-jdbc-3.8.11.2.jar**
package org.sqlite.core
[+] class CoreDatabaseMetaData 1
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added super-interface java.sql.DatabaseMetaData. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 7 (0.6%)  

CoreDatabaseMetaData.CoreDatabaseMetaData (  
SQLiteConnection conn )  
This constructor is from 'CoreDatabaseMetaData' abstract class.

CoreDatabaseMetaData.checkOpen ( )  
This method is from 'CoreDatabaseMetaData' abstract class.

CoreDatabaseMetaData.close ( )  
This method is from 'CoreDatabaseMetaData' abstract class.

CoreDatabaseMetaData.escape ( String val )  
This method is from 'CoreDatabaseMetaData' abstract class.

CoreDatabaseMetaData.finalize ( )  
This method is from 'CoreDatabaseMetaData' abstract class.

CoreDatabaseMetaData.getGeneratedKeys ( )  
This abstract method is from 'CoreDatabaseMetaData' abstract class.

CoreDatabaseMetaData.quote ( String tableName )  
This method is from 'CoreDatabaseMetaData' abstract class.

[+] class CorePreparedStatement 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field batchQueryCount has been added to this class. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 3 (0.2%)  

CorePreparedStatement.CorePreparedStatement (  
SQLiteConnection conn, String sql )  
This constructor is from 'CorePreparedStatement' abstract class.
CorePreparedStatement.batch ( int pos, Object value )
This method is from 'CorePreparedStatement' abstract class.

CorePreparedStatement.executeBatch ( )
This method is from 'CorePreparedStatement' abstract class.

[+] class CoreResultSet 2

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Field emptyResultSet has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>2 Field pastLastRow has been added to this class.</td>
<td>No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 19 (1.5%)

CoreResultSet.CoreResultSet ( CoreStatement stmt )
Field 'stmt.rs' in 1st parameter 'stmt' of this method is of type 'CoreResultSet'.

CoreResultSet.addColumnIndexInCache ( String col, int index )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkMeta ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.close ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.findColumnIndexInCache ( String col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.isOpen ( )
This method is from 'CoreResultSet' abstract class.
CoreResultSet.markCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreStatement.CoreStatement ( SQLiteConnection c )
Field 'this.rs' in the object of this method is of type 'CoreResultSet'.
...

<p>| 1 | Abstract method int backup ( String, String, DB.ProgressObserver, int, int, int ) has been added to this class. | No effect. |
| 2 | Abstract method void busy_handler ( BusyHandler ) has been added to this class. | No effect. |
| 3 | Abstract method long changes () has been added to this class. | No effect. |
| 4 | Abstract method void clear_progress_handler ( ) has been added to this class. | No effect. |
| 5 | Abstract method int create_collation ( String, Collation ) has been added to this class. | No effect. |
| 6 | Abstract method int create_function ( String, Function, int, int ) has been added to this class. | No effect. |
| 7 | Abstract method int destroy_collation ( String ) has been added to this | No effect. |</p>
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>class.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Abstract method <code>int limit ( int, int )</code> has been added to this class.</td>
</tr>
<tr>
<td>9</td>
<td>Abstract method <code>SafeStmtPtr prepare ( String )</code> has been added to this class.</td>
</tr>
<tr>
<td>10</td>
<td>Abstract method <code>void register_progress_handler ( int, ProgressHandler )</code> has been added to this class.</td>
</tr>
<tr>
<td>11</td>
<td>Abstract method <code>int restore ( String, String, DB.ProgressObserver, int, int, int )</code> has been added to this class.</td>
</tr>
<tr>
<td>12</td>
<td>Abstract method <code>long total_changes ( )</code> has been added to this class.</td>
</tr>
</tbody>
</table>

[+] affected methods: 46 (3.7%)
This abstract method is from 'DB' abstract class.

DB.clear_bindings ( long p1 )
This abstract method is from 'DB' abstract class.

DB.close ( )
This method is from 'DB' abstract class.

DB.column_blob ( long p1, int p2 )
This abstract method is from 'DB' abstract class.

DB.column_count ( long p1 )
This abstract method is from 'DB' abstract class.

DB.column_dectype ( long p1, int p2 )
This abstract method is from 'DB' abstract class.

...
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.close ()
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.execute ( String sql )
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.execute ( String sql, int[ ] colinds )
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.execute ( String sql, String[ ] colnames )
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.execute ( String sql, int autokeys )
This method is from 'JDBC3Statement' abstract class.

...

sqlite-jdbc-3.8.11.2.jar
package org.sqlite.util
[+] class OSInfo 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field processRunner has been added to this class. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 5 (0.4%)

OSInfo.OSInfo ()
This constructor is from 'OSInfo' class.

OSInfo.getArchName ()
This method is from 'OSInfo' class.

OSInfo.getNativeLibFolderPathForCurrentOS ()
This method is from 'OSInfo' class.

OSInfo.getOSName ()
This method is from 'OSInfo' class.

OSInfo.main ( String[ ] args )
This method is from 'OSInfo' class.

Java Archives 1

sqlite-jdbc-3.8.11.2.jar

Test Info

<table>
<thead>
<tr>
<th>Library Name</th>
<th>sqlite-jdbc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version #1</td>
<td>3.8.11.2</td>
</tr>
<tr>
<td>Version #2</td>
<td>3.42.0.0</td>
</tr>
<tr>
<td>Subject</td>
<td>Source Compatibility</td>
</tr>
</tbody>
</table>

Test Results

<table>
<thead>
<tr>
<th>Total Java Modules</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Methods / Classes</td>
<td>1246 / 81</td>
</tr>
<tr>
<td>Compatibility</td>
<td>61.2%</td>
</tr>
</tbody>
</table>

Problem Summary

<table>
<thead>
<tr>
<th></th>
<th>Severity</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Added Methods</td>
<td>-</td>
<td>203</td>
</tr>
<tr>
<td>Removed Methods</td>
<td>High</td>
<td>45</td>
</tr>
<tr>
<td>Problems with Data Types</td>
<td>Severity</td>
<td>Count</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Problems with Methods</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other Changes in Data Types</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
package org.sqlite.core
CorePreparedStatement.clearBatch( ) : void
org/sqlite/core/CorePreparedStatement.clearBatch():()V
CorePreparedStatement.executeLargeBatch( ) : long
org/sqlite/core/CorePreparedStatement.executeLargeBatch():()[J
CorePreparedStatement.setDateByMilliseconds( int pos, Long value, Calendar calendar ) : void
org/sqlite/core/CorePreparedStatement.setDateByMilliseconds:(ILjava/lang/Long;Ljava/util/Calendar;)V

package org.sqlite.core
CoreResultSet.getConnectionConfig( ) : SQLiteConnectionConfig
org/sqlite/core/CoreResultSet.getConnectionConfig():()Lorg/sqlite/SQLiteConnectionConfig;
CoreResultSet.getDatabase( ) : DB
org/sqlite/core/CoreResultSet.getDatabase():()Lorg/sqlite/core/DB;

package org.sqlite.core
CoreStatement.checkIndex( int index ) : void
org/sqlite/core/CoreStatement.checkIndex:(I)V
CoreStatement.getConnectionConfig( ) : SQLiteConnectionConfig
org/sqlite/core/CoreStatement.getConnectionConfig():()Lorg/sqlite/SQLiteConnectionConfig;
CoreStatement.getDatabase( ) : DB
org/sqlite/core/CoreStatement.getDatabase():()Lorg/sqlite/core/DB;
CoreStatement.notifyFirstStatementExecuted( ) : void
org/sqlite/core/CoreStatement.notifyFirstStatementExecuted():()V

package org.sqlite.core
DB.addCommitListener( SQLiteCommitListener listener ) : void
org/sqlite/core/DB.addCommitListener:(Lorg/sqlite/SQLiteCommitListener;)V
DB.addUpdateListener( SQLiteUpdateListener listener ) : void
org/sqlite/core/DB.addUpdateListener:(Lorg/sqlite/SQLiteUpdateListener;)V
DB.backup( String p1, String p2, DB.ProgressObserver p3, int p4, int p5, int p6 ) [abstract] :
int
org/sqlite/core/DB.backup:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III)I
DB.busy_handler ( BusyHandler p1 ) [abstract] : void
org/sqlite/core/DB.busy_handler:(Lorg/sqlite/BusyHandler;)V
DB.changes ( ) [abstract] : long
org/sqlite/core/DB.changes:()J
DB.clear_progress_handler ( ) [abstract] : void
org/sqlite/core/DB.clear_progress_handler:()V
DB.create_collation ( String p1, Collation p2 ) [abstract] : int
org/sqlite/core/DB.create_collation:(Ljava/lang/String;Lorg/sqlite/Collation;)I
DB.create_function ( String p1, Function p2, int p3, int p4 ) [abstract] : int
org/sqlite/core/DB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;II)I
DB.DB ( String url, String fileName, SQLiteConfig config )
org/sqlite/core/DB."<init>":(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/SQLiteConfig;)V
DB.destroy_collation ( String p1 ) [abstract] : int
org/sqlite/core/DB.destroy_collation:(Ljava/lang/String;)I
DB.exec ( String sql, boolean autoCommit ) : void
org/sqlite/core/DB.exec:(Ljava/lang/String;Z)V
DB.executeUpdate ( CoreStatement stmt, Object[] vals ) : long
org/sqlite/core/DB.executeUpdate:(Lorg/sqlite/core/CoreStatement;[Ljava/lang/Object;)J
DB.finalize ( SafeStmtPtr safePtr, long ptr ) : int
org/sqlite/core/DB.finalize:(Lorg/sqlite/core/SafeStmtPtr;J)I
DB.getConfig ( ) : SQLiteConfig
org/sqlite/core/DB.getConfig:()Lorg/sqlite/SQLiteConfig;
DB.getUrl ( ) : String
org/sqlite/core/DB.getUrl:()Ljava/lang/String;
DB.isClosed ( ) : boolean
org/sqlite/core/DB.isClosed:()Z
DB.limit ( int p1, int p2 ) [abstract] : int
org/sqlite/core/DB.limit:(II)I
DB.newSQLException ( int errorCode, String errorMessage ) [static] : SQLiteException
org/sqlite/core/DB.newSQLException:(ILjava/lang/String;)Lorg/sqlite/SQLiteException;
DB.open ( String file, int openFlags ) : void
org/sqlite/core/DB.open:(Ljava/lang/String;I)V
DB.prepare ( String p1 ) [abstract] : SafeStmtPtr
org/sqlite/core/DB.prepare:(Ljava/lang/String;)Lorg/sqlite/core/SafeStmtPtr;
DB.register_progress_handler ( int p1, ProgressHandler p2 ) [abstract] : void

org/sqlite/core/DB.register_progress_handler:(IIorg/sqlite/ProgressHandler;)V

DB.removeCommitListener ( SQLiteCommitListener listener ) : void

org/sqlite/core/DB.removeCommitListener:(Lorg/sqlite/SQLiteCommitListener;)V

DB.removeUpdateListener ( SQLiteUpdateListener listener ) : void

org/sqlite/core/DB.removeUpdateListener:(Lorg/sqlite/SQLiteUpdateListener;)V

DB.restore ( String p1, String p2, DB.ProgressObserver p3, int p4, int p5, int p6 ) [abstract] : int

org/sqlite/core/DB.restore:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III)I

DB.total_changes ( ) [abstract] : long

org/sqlite/core/DB.total_changes:()J

sqlite-jdbc-3.42.0.0.jar, Function.class
package org.sqlite
Function.create ( Connection conn, String name, Function f, int flags ) [static] : void

org/sqlite/Function.create:(Ljava/sql/Connection;Ljava/lang/String;Lorg/sqlite/Function;)V

Function.create ( Connection conn, String name, Function f, int nArgs, int flags ) [static] : void

org/sqlite/Function.create:(Ljava/sql/Connection;Ljava/lang/String;Lorg/sqlite/Function;II)V

Function.destroy ( Connection conn, String name, int nArgs ) [static] : void

org/sqlite/Function.destroy:(Ljava/sql/Connection;Ljava/lang/String;I)V

sqlite-jdbc-3.42.0.0.jar, HexKeyMode.class
package org.sqlite
SQLiteConfig.HexKeyMode.getValue ( ) : String

org/sqlite/SQLiteConfig.HexKeyMode.getValue:()Ljava/lang/String;
SQLiteConfig.HexKeyMode.valueOf ( String name ) [static] : SQLiteConfig.HexKeyMode
org/sqlite/SQLiteConfig.HexKeyMode.valueOf:(Ljava/lang/String;)Lorg/sqlite/SQLiteConfig$HexKeyMode;
SQLiteConfig.HexKeyMode.values ( ) [static] : SQLiteConfig.HexKeyMode[]
org/sqlite/SQLiteConfig.HexKeyMode.values:()[Lorg/sqlite/SQLiteConfig$HexKeyMode;

sqlite-jdbc-3.42.0.0.jar, JDBC.class
package org.sqlite
JDBC.createConnection (String url, Properties prop) [static] : SQLiteConnection

org/sqlite/JDBC.createConnection:(Ljava/lang/String;Ljava/util/Properties;)Lorg/sqlite/SQLiteConnection;

sqlite-jdbc-3.42.0.0.jar, JDBC3Connection.class
package org.sqlite.jdbc3
JDBC3Connection.tryEnforceTransactionMode () : void
org/sqlite/jdbc3/JDBC3Connection.tryEnforceTransactionMode():V

sqlite-jdbc-3.42.0.0.jar, JDBC3PreparedStatement.class
package org.sqlite.jdbc3
JDBC3PreparedStatement.execute (String sql, int autoGeneratedKeys) : boolean
JDBC3PreparedStatement.execute (String sql, int[] colinds) : boolean
JDBC3PreparedStatement.execute (String sql, String[] colnames) : boolean
JDBC3PreparedStatement.executeLargeUpdate () : long
org/sqlite/jdbc3/JDBC3PreparedStatement.executeLargeUpdate():J
JDBC3PreparedStatement.executeLargeUpdate (String sql) : long
org/sqlite/jdbc3/JDBC3PreparedStatement.executeLargeUpdate:(Ljava/lang/String;)J
JDBC3PreparedStatement.executeLargeUpdate (String sql, int autoGeneratedKeys) : long
org/sqlite/jdbc3/JDBC3PreparedStatement.executeLargeUpdate:(Ljava/lang/String;I)J
JDBC3PreparedStatement.executeLargeUpdate (String sql, int[] colinds) : long
JDBC3PreparedStatement.executeLargeUpdate (String sql, String[] cols) : long
JDBC3PreparedStatement.executeUpdate (String sql, int autoGeneratedKeys) : int
JDBC3PreparedStatement.executeUpdate (String sql, int[] colinds) : int
JDBC3PreparedStatement.executeUpdate (String sql, String[] cols) : int
sqlite-jdbc-3.42.0.0.jar  JDBC3ResultSet.class
package org.sqlite.jdbc3
JDBC3ResultSet.safeGetColumnType ( int col ) : int
org/sqlite/jdbc3/JDBC3ResultSet.safeGetColumnType:(I)I

sqlite-jdbc-3.42.0.0.jar  JDBC3Statement.class
package org.sqlite.jdbc3
JDBC3Statement.executeLargeBatch ( ) : long[
org/sqlite/jdbc3/JDBC3Statement.executeLargeBatch():[J
JDBC3Statement.executeLargeUpdate ( String sql ) : long
org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;)J
JDBC3Statement.executeLargeUpdate ( String sql, int autoGeneratedKeys ) : long
org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;)I)J
JDBC3Statement.executeLargeUpdate ( String sql, int[] colinds ) : long
org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;[I)J
JDBC3Statement.executeLargeUpdate ( String sql, String[] cols ) : long
org/sqlite/jdbc3/JDBC3Statement.executeLargeUpdate:(Ljava/lang/String;Ljava/lang/String;)J
JDBC3Statement.getLargeMaxRows ( ) : long
org/sqlite/jdbc3/JDBC3Statement.getLargeMaxRows():J
JDBC3Statement.getLargeUpdateCount ( ) : long
org/sqlite/jdbc3/JDBC3Statement.getLargeUpdateCount():J
JDBC3Statement.setLargeMaxRows ( long max ) : void
org/sqlite/jdbc3/JDBC3Statement.setLargeMaxRows:(J)V
JDBC3Statement.unsupported ( ) : SQLException
org/sqlite/jdbc3/JDBC3Statement.unsupported():Ljava/sql/SQLException;
JDBC3Statement.withConnectionTimeout ( JDBC3Statement.SQLCallable<T> callable ) : T
org/sqlite/jdbc3/JDBC3Statement.withConnectionTimeout:(Lorg/sqlite
java/lang/Object

package org.sqlite.jdbc4

JDBC4PreparedStatement.toString(): String
org/sqlite/jdbc4/JDBC4PreparedStatement.toString():()Ljava/lang/String;

JDBC4ResultSet.close(): void
org/sqlite/jdbc4/JDBC4ResultSet.close():()V
JDBC4ResultSet.unsupported(): SQLException
org/sqlite/jdbc4/JDBC4ResultSet.unsupported():()Ljava/sql/SQLException;

JDBC4Statement.close(): void
org/sqlite/jdbc4/JDBC4Statement.close():()V

LibraryLoaderUtil.getNativeLibName(): String
org/sqlite/util/LibraryLoaderUtil.getNativeLibName():()Ljava/lang/String;

LibraryLoaderUtil.getNativeLibResourcePath(): String
org/sqlite/util/LibraryLoaderUtil.getNativeLibResourcePath():()Ljava/lang/String;

LibraryLoaderUtil.hasNativeLib(String path, String libraryName): boolean
org/sqlite/util/LibraryLoaderUtil.hasNativeLib:(Ljava/lang/String;Ljava/lang/String;)Z

LibraryLoaderUtil.LibraryLoaderUtil():
org/sqlite/util/LibraryLoaderUtil."<init>":()V

NativeDB.backup(): int
org/sqlite/core/NativeDB.backup:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III):I
NativeDB.busy_handler(): void
org/sqlite/core/NativeDB.busy_handler:()V
org/sqlite/core/NativeDB.busy_handler:(Lorg/sqlite/BusyHandler;)V
NativeDB.changes ( ) : long
org/sqlite/core/NativeDB.changes:()J
NativeDB.clear_progress_handler ( ) : void
org/sqlite/core/NativeDB.clear_progress_handler:()V
NativeDB.create_collation ( String name, Collation coll ) : int
org/sqlite/core/NativeDB.create_collation:(Ljava/lang/String;Lorg/sqlite/Collation;)I
NativeDB.create_function ( String name, Function func, int nArgs, int flags ) : int
org/sqlite/core/NativeDB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;II)I
NativeDB.destroy_collation ( String name ) : int
org/sqlite/core/NativeDB.destroy_collation:(Ljava/lang/String;I)
NativeDB.limit ( int p1, int p2 ) : int
org/sqlite/core/NativeDB.limit:(II)I
NativeDB.NativeDB ( String url, String fileName, SQLiteConfig config )
org/sqlite/core/NativeDB."<init>":(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/SQLiteConfig;)V
NativeDB.register_progress_handler ( int p1, ProgressHandler p2 ) : void
org/sqlite/core/NativeDB.register_progress_handler:(ILorg/sqlite/ProgressHandler;)V
NativeDB.restore ( String dbName, String sourceFileName, DB.ProgressObserver observer, int sleepTimeMillis, int nTimeouts, int pagesPerStep ) : int
org/sqlite/core/NativeDB.restore:(Ljava/lang/String;Ljava/lang/String;Lorg/sqlite/core/DB$ProgressObserver;III)I
NativeDB.total_changes ( ) : long
org/sqlite/core/NativeDB.total_changes:()J

sqlite-jdbc-3.42.0.0.jar, OSInfo.class
package org.sqlite.util
OSInfo.isAndroid ( ) [static] : boolean
org/sqlite/util/OSInfo.isAndroid:(Z
OSInfo.isAndroidRuntime ( ) [static] : boolean
org/sqlite/util/OSInfo.isAndroidRuntime:(Z
OSInfo.isAndroidTermux ( ) [static] : boolean
org/sqlite/util/OSInfo.isAndroidTermux:(Z
OSInfo.isMusl ( ) [static] : boolean
org/sqlite/util/OSInfo.isMusl:(Z

sqlite-jdbc-3.42.0.0.jar, ProcessRunner.class
package org.sqlite.util
ProcessRunner.ProcessRunner ( )
org/sqlite/util/ProcessRunner."<init>":()V

sqlite-jdbc-3.42.0.0.jar, ProgressHandler.class
package org.sqlite
ProgressHandler.clearHandler ( Connection conn ) [static] : void
org/sqlite/ProgressHandler.clearHandler:(Ljava/sql/Connection;)V
ProgressHandler.progress ( ) [abstract] : int
org/sqlite/ProgressHandler.progress:()I
ProgressHandler.ProgressHandler ( )
org/sqlite/ProgressHandler."<init>":()V
ProgressHandler.setHandler ( Connection conn , int vmCalls , ProgressHandler progressHandler ) [static] : void
org/sqlite/ProgressHandler.setHandler:(Ljava/sql/Connection;ILorg/sqlite/ProgressHandler;)V

sqlite-jdbc-3.42.0.0.jar, QueryUtils.class
package org.sqlite.util
QueryUtils.QueryUtils ( )
org/sqlite/util/QueryUtils."<init>":()V
QueryUtils.valuesQuery ( List<String> columns , List<List<Object>> valuesList ) [static] : String
org/sqlite/util/QueryUtils.valuesQuery:(Ljava/util/List;Ljava/util/List;)Ljava/lang/String;

sqlite-jdbc-3.42.0.0.jar, SafePtrConsumer.class
package org.sqlite.core
SafeStmtPtr.SafePtrConsumer<E>.run ( DB p1 , long p2 ) [abstract] : void
org/sqlite/core/SafeStmtPtr.SafePtrConsumer.run:(Lorg/sqlite/core/DB;J)V

sqlite-jdbc-3.42.0.0.jar, SafePtrDoubleFunction.class
package org.sqlite.core
SafeStmtPtr.SafePtrDoubleFunction<E>.run ( DB p1 , long p2 ) [abstract] : double
org/sqlite/core/SafeStmtPtr.SafePtrDoubleFunction.run:(Lorg/sqlite/core/DB;J)D

sqlite-jdbc-3.42.0.0.jar, SafePtrFunction.class
package org.sqlite.core
SafeStmtPtr.SafePtrFunction<T,E>.run ( DB p1 , long p2 ) [abstract] : T
org/sqlite/core/SafeStmtPtr.SafePtrFunction.run:(Lorg/sqlite/core/DB;J)Ljava/lang/Object;
package org.sqlite.core

SafeStmtPtr.SafePtrIntFunction<E>.run (DB p1, long p2) [abstract] : int

package org.sqlite.core

SafeStmtPtr.SafePtrLongFunction<E>.run (DB p1, long p2) [abstract] : long

package org.sqlite.core

SafeStmtPtr.close () : int

SafeStmtPtr.safeRun (SafeStmtPtr.SafePtrFunction<T,E> run) : T

SafeStmtPtr.safeRunConsume (SafeStmtPtr.SafePtrConsumer<E> run) : void

SafeStmtPtr.safeRunDouble (SafeStmtPtr.SafePtrDoubleFunction<E> run) : double

SafeStmtPtr.safeRunInt (SafeStmtPtr.SafePtrIntFunction<E> run) : int

SafeStmtPtr.safeRunLong (SafeStmtPtr.SafePtrLongFunction<E> run) : long

SafeStmtPtr.<init> (DB db, long ptr) : (Lorg/sqlite/core/DB;J)V
package org.sqlite.jdbc3
JDBC3Statement.SQLCallable<T>.call ( ) [abstract] : T
org/sqlite/jdbc3/JDBC3Statement.SQLCallable.call:()Ljava/lang/Object;

package org.sqlite
SQLiteCommitListener.onCommit ( ) [abstract] : void
org/sqlite/SQLiteCommitListener.onCommit:()V
SQLiteCommitListener.onRollback ( ) [abstract] : void
org/sqlite/SQLiteCommitListener.onRollback:()V

package org.sqlite
SQLiteConfig.deferForeignKeys ( boolean enable ) : void
org/sqlite/SQLiteConfig.deferForeignKeys:(Z)V
SQLiteConfig.getBusyTimeout ( ) : int
org/sqlite/SQLiteConfig.getBusyTimeout:()I
SQLiteConfig.isExplicitReadOnly ( ) : boolean
org/sqlite/SQLiteConfig.isExplicitReadOnly:(Z)
SQLiteConfig.newConnectionConfig ( ) : SQLiteConnectionConfig
org/sqlite/SQLiteConfig.newConnectionConfig:()Lorg/sqlite/SQLiteConnectionConfig;
SQLiteConfig.setApplicationId ( int id ) : void
org/sqlite/SQLiteConfig.setApplicationId:(I)V
SQLiteConfig.setBusyTimeout ( int milliseconds ) : void
org/sqlite/SQLiteConfig.setBusyTimeout:(I)V
SQLiteConfig.setExplicitReadOnly ( boolean readOnly ) : void
org/sqlite/SQLiteConfig.setExplicitReadOnly:(Z)V
SQLiteConfig.setHexKeyMode ( SQLiteConfig.HexKeyMode mode ) : void
org/sqlite/SQLiteConfig.setHexKeyMode:(Lorg/sqlite/SQLiteConfig$HexKeyMode;)V
SQLiteConfig.setLegacyAlterTable ( boolean flag ) : void
org/sqlite/SQLiteConfig.setLegacyAlterTable:(Z)V

package org.sqlite
SQLiteConnection.addCommitListener ( SQLiteCommitListener listener ) : void
org/sqlite/SQLiteConnection.addCommitListener:(Lorg/sqlite/SQLiteCommitListener;)V
SQLiteConnection.addUpdateListener (SQLiteUpdateListener listener): void
org/sqlite/SQLiteConnection.addUpdateListener:(Lorg/sqlite/SQLiteUpdateListener;)V
SQLiteConnection.checkCursor (int rst, int rsc, int rsh): void
org/sqlite/SQLiteConnection.checkCursor:(III)V
SQLiteConnection.checkOpen(): void
org/sqlite/SQLiteConnection.checkOpen:()V
SQLiteConnection.close(): void
org/sqlite/SQLiteConnection.close:()V
SQLiteConnection.commit(): void
org/sqlite/SQLiteConnection.commit:()V
SQLiteConnection.extractPragmasFromFilename (String url, String filename, Properties prop) [static]: String
org/sqlite/SQLiteConnection.extractPragmasFromFilename:(Ljava/lang/String;Ljava/lang/String;Ljava/util/Properties;)Ljava/lang/String;
SQLiteConnection.getAutoCommit(): boolean
org/sqlite/SQLiteConnection.getAutoCommit:()Z
SQLiteConnection.getBusyTimeout(): int
org/sqlite/SQLiteConnection.getBusyTimeout:()I
SQLiteConnection.getConnectionConfig(): SQLiteConnectionConfig
org/sqlite/SQLiteConnection.getConnectionConfig:()Lorg/sqlite/SQLiteConnectionConfig;
SQLiteConnection.getCurrentTransactionMode(): SQLiteConfig.TransactionMode
org/sqlite/SQLiteConnection.getCurrentTransactionMode:()Lorg/sqlite/SQLiteConfig$TransactionMode;
SQLiteConnection.getDatabase(): DB
org/sqlite/SQLiteConnection.getDatabase:()Lorg/sqlite/core/DB;
SQLiteConnection.getLimit (SQLiteLimits limit): void
org/sqlite/SQLiteConnection.getLimit:(Lorg/sqlite/SQLiteLimits;)V
SQLiteConnection.getMetaData(): DatabaseMetaData
org/sqlite/SQLiteConnection.getMetaData:()Ljava/sql/DatabaseMetaData;
SQLiteConnection.getSQLiteDatabaseMetaData(): CoreDatabaseMetaData
org/sqlite/SQLiteConnection.getSQLiteDatabaseMetaData:()Lorg/sqlite/core/CoreDatabaseMetaData;
SQLiteConnection.getTransactionIsolation(): int
org/sqlite/SQLiteConnection.getTransactionIsolation:()I
SQLiteConnection.getUrl(): String
org/sqlite/SQLiteConnection.getUrl:()Ljava/lang/String;
SQLiteConnection.isClosed(): boolean
org/sqlite/SQLiteConnection.isClosed:()Z
SQLiteConnection isFirstStatementExecuted(): boolean
org/sqlite/SQLiteConnection.isFirstStatementExecuted:()Z
org/sqlite/SQLiteConnection.isFirstStatementExecuted():Z
SQLiteConnection.libversion (): String
org/sqlite/SQLiteConnection.libversion():Ljava/lang/String;
SQLiteConnection.removeCommitListener (SQLiteCommitListener listener): void
org/sqlite/SQLiteConnection.removeCommitListener:(Lorg/sqlite/SQLiteCommitListener;)V
SQLiteConnection.removeUpdateListener (SQLiteUpdateListener listener): void
org/sqlite/SQLiteConnection.removeUpdateListener:(Lorg/sqlite/SQLiteUpdateListener;)V
SQLiteConnection.rollback (): void
org/sqlite/SQLiteConnection.rollback():V
SQLiteConnection.setAutoCommit (boolean ac): void
org/sqlite/SQLiteConnection.setAutoCommit:(Z)V
SQLiteConnection.setBusyTimeout (int timeoutMillis): void
org/sqlite/SQLiteConnection.setBusyTimeout:(I)V
SQLiteConnection.setCurrentTransactionMode (SQLiteConfig.TransactionMode currentTransactionMode): void
org/sqlite/SQLiteConnection.setCurrentTransactionMode:(Lorg/sqlite/SQLiteConfig$TransactionMode;)V
SQLiteConnection.setFirstStatementExecuted (boolean firstStatementExecuted): void
org/sqlite/SQLiteConnection.setFirstStatementExecuted:(Z)V
SQLiteConnection.setLimit (SQLiteLimits limit, int value): void
org/sqlite/SQLiteConnection.setLimit:(Lorg/sqlite/SQLiteLimits;I)V
SQLiteConnection.setTransactionIsolation (int level): void
org/sqlite/SQLiteConnection.setTransactionIsolation:(I)V
SQLiteConnection.setTransactionMode (SQLiteConfig.TransactionMode mode): void
org/sqlite/SQLiteConnection.setTransactionMode:(Lorg/sqlite/SQLiteConfig$TransactionMode;)V
SQLiteConnection.SQLiteConnection (DB db)
org/sqlite/SQLiteConnection."<init>":(Lorg/sqlite/core/DB;)V
SQLiteConnection.transactionPrefix (): String
org/sqlite/SQLiteConnection.transactionPrefix():Ljava/lang/String;

sqlite-jdbc-3.42.0.0.jar, SQLiteConnectionConfig.class
package org.sqlite
SQLiteConnectionConfig.copyConfig (): SQLiteConnectionConfig
org/sqlite/SQLiteConnectionConfig.copyConfig():Lorg/sqlite/SQLiteConnectionConfig;
SQLiteConnectionConfig.fromPragmaTable (Properties pragmaTable) [static]
: SQLiteConnectionConfig
org/sqlite/SQLiteConnectionConfig.fromPragmaTable:(Ljava/util/Properties;)Lorg/sqlite/SQLiteConnectionConfig;
SQLiteConnectionConfig.getDateClass() : SQLiteConfig.DateClass
org/sqlite.SQLiteConnectionConfig.getDateClass():(Lorg/sqlite/SQLite
Config$DateClass;
SQLiteConnectionConfig.getDateFormat() : FastDateFormat
org/sqlite SQLiteConnectionConfig.getDateFormat():(Lorg/sqlite/date/F
astDateFormat;
SQLiteConnectionConfig.getDateMultiplier() : long
org/sqlite SQLiteConnectionConfig.getDateMultiplier():(J
SQLiteConnectionConfig.getDatePrecision() : SQLiteConfig.DatePrecision
org/sqlite SQLiteConnectionConfig.getDatePrecision():(Lorg/sqlite/SQ
LiteConfig$DatePrecision;
SQLiteConnectionConfig.getDateStringFormat() : String
org/sqlite SQLiteConnectionConfig.getDateStringFormat():(Ljava/lang/
String;
SQLiteConnectionConfig.getTransactionIsolation() : int
org/sqlite SQLiteConnectionConfig.getTransactionIsolation():(l
SQLiteConnectionConfig.getTransactionMode() : SQLiteConfig.TransactionMode
org/sqlite SQLiteConnectionConfig.getTransactionMode():(Lorg/sqlite/
SQLiteConfig$TransactionMode;
SQLiteConnectionConfig.isAutoCommit() : boolean
org/sqlite SQLiteConnectionConfig.isAutoCommit():(Z
SQLiteConnectionConfig.setAutoCommit( boolean autoCommit ) : void
org/sqlite SQLiteConnectionConfig.setAutoCommit:(Z)V
SQLiteConnectionConfig.setDateClass( SQLiteConfig.DateClass dateClass ) : void
org/sqlite SQLiteConnectionConfig.setDateClass:(Lorg/sqlite/SQLiteC
onfig$DateClass;)V
SQLiteConnectionConfig.setDatePrecision( SQLiteConfig.DatePrecision datePrecision )
 : void
org/sqlite SQLiteConnectionConfig.setDatePrecision:(Lorg/sqlite/SQL
iteConfig$DatePrecision;)V
SQLiteConnectionConfig.setDateStringFormat( String dateStringFormat ) : void
org/sqlite SQLiteConnectionConfig.setDateStringFormat:(Ljava/lang/S
tring;)V
SQLiteConnectionConfig.getTransactionIsolation( int transactionIsolation ) : void
org/sqlite SQLiteConnectionConfig.getTransactionIsolation:(l)V
SQLiteConnectionConfig.getTransactionMode( SQLiteConfig.TransactionMode transactionMode ) : void
org/sqlite SQLiteConnectionConfig.getTransactionMode:(Lorg/sqlite/S
SQLiteConfig$TransactionMode;)V
SQLiteConnectionConfig.SQLiteConnectionConfig( SQLiteConfig.DateClass dateClass,
SQLiteConfig.DatePrecision datePrecision, String dateStringFormat, int transactionIsolation,
SQLiteConfig.TransactionMode transactionMode, boolean autoCommit )
org/sqlite/SQLiteConnectionConfig.<init>:(Lorg/sqlite/SQLiteConfig$DateClass;Lorg/sqlite/SQLiteConfig$DatePrecision;Ljava/lang/String;ILorg/sqlite/SQLiteConfig$TransactionMode;Z)V

**sqlite-jdbc-3.42.0.0.jar, SQLiteDataSource.class**

package org.sqlite

SQLiteDataSource.getConnection ( String username, String password ) : SQLiteConnection

org/sqlite/SQLiteDataSource.getConnection:(Ljava/lang/String;Ljava/lang/String;)Lorg/sqlite/SQLiteConnection;

SQLiteDataSource.setBusyTimeout ( int milliseconds ) : void

org/sqlite/SQLiteDataSource.setBusyTimeout:(I)V

SQLiteDataSource.setCountChanges ( boolean enable ) : void

org/sqlite/SQLiteDataSource.setCountChanges:(Z)V

SQLiteDataSource.setEnforceForeignKeys ( boolean enforce ) : void

org/sqlite/SQLiteDataSource.setEnforceForeignKeys:(Z)V

SQLiteDataSource.setLegacyAlterTable ( boolean flag ) : void

org/sqlite/SQLiteDataSource.setLegacyAlterTable:(Z)V

**sqlite-jdbc-3.42.0.0.jar, SQLiteException.class**

package org.sqlite

SQLiteException.getResultCode ( ) : SQLiteErrorCode

org/sqlite/SQLiteException.getResultCode:()Lorg/sqlite/SQLiteErrorCode;

SQLiteException.SQLiteException ( String message, SQLiteErrorCode resultCode )

org/sqlite/SQLiteException."<init>":(Ljava/lang/String;Lorg/sqlite/SQLiteErrorCode;)V

**sqlite-jdbc-3.42.0.0.jar, SqliteJdbcFeature.class**

package org.sqlite.nativeimage

SqliteJdbcFeature.beforeAnalysis ( Feature.BeforeAnalysisAccess a ) : void

org/sqlite/nativeimage/SqliteJdbcFeature.beforeAnalysis:(Lorg/graalvm/nativeimage/hosted/Feature$BeforeAnalysisAccess;)V

SqliteJdbcFeature.SqliteJdbcFeature ( )

org/sqlite/nativeimage/SqliteJdbcFeature."<init>":()V

**sqlite-jdbc-3.42.0.0.jar, SQLiteLimits.class**

package org.sqlite

SQLiteLimits.getId ( ) : int

org/sqlite/SQLiteLimits.getId:(I)

SQLiteLimits.valueOf ( String name ) [static] : SQLiteLimits

org/sqlite/SQLiteLimits.valueOf:(Ljava/lang/String;)Lorg/sqlite/SQLiteLimits;
Limits;
SQLiteLimits.values ( ) [static] : SQLiteLimits[]
org/sqlite/SQLiteLimits.values:()[Lorg/sqlite/SQLiteLimits;

sqlite-jdbc-3.42.0.0.jar, SQLitePooledConnection.class
package org.sqlite.javex
SQLitePooledConnection.getListeners ( ) : List<ConnectionEventListener>
org/sqlite/javex/SQLitePooledConnection.getListeners:()Ljava/util/List;
SQLitePooledConnection.getPhysicalConn ( ) : SQLiteConnection
org/sqlite/javex/SQLitePooledConnection.getPhysicalConn:()Lorg/sqlite/SQLiteConnection;
SQLitePooledConnection.SQLitePooledConnection ( SQLiteConnection physicalConn )
org/sqlite/javex/SQLitePooledConnection."<init>":(Lorg/sqlite/SQLiteConnection;)V

sqlite-jdbc-3.42.0.0.jar, SQLiteUpdateListener.class
package org.sqlite
SQLiteUpdateListener.onUpdate ( SQLiteUpdateListener.Type p1, String p2, String p3, long p4 ) [abstract] : void
org/sqlite/SQLiteUpdateListener.onUpdate:(Lorg/sqlite/SQLiteUpdateListener$Type;Ljava/lang/String;Ljava/lang/String;J)V

sqlite-jdbc-3.42.0.0.jar, Type.class
package org.sqlite
SQLiteUpdateListener.Type.valueOf ( String name ) [static] : SQLiteUpdateListener.Type
org/sqlite/SQLiteUpdateListener.Type.valueOf:(Ljava/lang/String;)Lorg/sqlite/SQLiteUpdateListener$Type;
SQLiteUpdateListener.Type.values ( ) [static] : SQLiteUpdateListener.Type[]
org/sqlite/SQLiteUpdateListener.Type.values:()[Lorg/sqlite/SQLiteUpdateListener$Type;

sqlite-jdbc-3.42.0.0.jar, VersionHolder.class
package org.sqlite
SQLiteJDBCLoader.VersionHolder.SQLiteJDBCLoader.VersionHolder ( )
org/sqlite/SQLiteJDBCLoader.VersionHolder."<init>":()V

sqlite-jdbc-3.42.0.0.jar, Window.class
package org.sqlite
Function.Window.Function.Window ( )
org/sqlite/Function.Window."<init>":()V
Function.Window.xInverse ( ) [abstract] : void
org/sqlite/Function.Window.xInverse:()V
Function.Window.xValue ( ) [abstract] : void
org/sqlite/Function.Window.xValue:()V

to the top

### Removed Methods 45

**sqlite-jdbc-3.8.11.2.jar, CoreConnection.class**

package org.sqlite.core
CoreConnection.checkCursor ( int rst, int rsc, int rsh ) : void
org/sqlite/core/CoreConnection.checkCursor:(III)V
CoreConnection.checkOpen ( ) : void
org/sqlite/core/CoreConnection.checkOpen:()V
CoreConnection.close ( ) : void
org/sqlite/core/CoreConnection.close:()V
CoreConnection.CoreConnection ( String url, String fileName, Properties prop )
org/sqlite/core/CoreConnection."<init>":(Ljava/lang/String;Ljava/lang/String;Ljava/util/Properties;)V
CoreConnection.db ( ) : DB
org/sqlite/core/CoreConnection.db:()Lorg/sqlite/core/DB;
CoreConnection.getBusyTimeout ( ) : int
org/sqlite/core/CoreConnection.getBusyTimeout:()I
CoreConnection.getDriverVersion ( ) : String
org/sqlite/core/CoreConnection.getDriverVersion:()Ljava/lang/String;
CoreConnection.libversion ( ) : String
org/sqlite/core/CoreConnection.libversion:()Ljava/lang/String;
CoreConnection.setBusyTimeout ( int milliseconds ) : void
org/sqlite/core/CoreConnection.setBusyTimeout:(I)V
CoreConnection.setTransactionMode ( SQLiteConfig.TransactionMode mode ) : void
org/sqlite/core/CoreConnection.setTransactionMode:(Lorg/sqlite/SQLiteConfig$TransactionMode;)V
CoreConnection.url ( ) : String
org/sqlite/core/CoreConnection.url:()Ljava/lang/String;

**sqlite-jdbc-3.8.11.2.jar, CorePreparedStatement.class**

package org.sqlite.core
CorePreparedStatement.checkParameters ( ) : void
org/sqlite/core/CorePreparedStatement.checkParameters:()V
CorePreparedStatement.setDbDateByMilliseconds ( int pos, Long value ) : void
org/sqlite/core/CorePreparedStatement.setDbDateByMilliseconds:(ILjava/lang/Long;)V

sqlite-jdbc-3.8.11.2.jar, DB.class
package org.sqlite.core
DB.changes ( ) [abstract] : int
org/sqlite/core/DB.changes:(I)V
DB.create_function ( String p1, Function p2 ) [abstract] : int
org/sqlite/core/DB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;)I
DB.DB ( )
org/sqlite/core/DB."<init>";():V
DB.exec ( String sql ) : void
org/sqlite/core/DB.exec:(Ljava/lang/String;)V
DB.executeUpdate ( CoreStatement stmt, Object[] vals ) : int
org/sqlite/core/DB.executeUpdate:(Lorg/sqlite/core/CoreStatement;[Ljava/lang/Object;)I
DB.finalize ( CoreStatement stmt ) : int
org/sqlite/core/DB.finalize:(Lorg/sqlite/core/CoreStatement;)I
DB.newSQLException ( int errorCode, String errorMessage ) [static] : SQLException
org/sqlite/core/DB.newSQLException:(ILjava/lang/String;)Ljava/sql/SQLException;
DB.open ( SQLiteConnection conn, String file, int openFlags ) : void
org/sqlite/core/DB.open:(Lorg/sqlite/SQLiteConnection;Ljava/lang/String;)V
DB.prepare ( String p1 ) [abstract] : long
org/sqlite/core/DB.prepare:(Ljava/lang/String;)J
DB.total_changes ( ) [abstract] : int
org/sqlite/core/DB.total_changes:(I)V
DB.value_bytes ( Function p1, int p2 ) [abstract] : int
org/sqlite/core/DB.value_bytes:(Lorg/sqlite/Function;I)I

sqlite-jdbc-3.8.11.2.jar, Function.class
package org.sqlite
Function.value_bytes ( int arg ) : int
org/sqlite/Function.value_bytes:(I)V

sqlite-jdbc-3.8.11.2.jar, JDBC.class
package org.sqlite
JDBC.createConnection ( String url, Properties prop ) [static] : Connection
org/sqlite/JDBC.createConnection:(Ljava/lang/String;Ljava/util/Properties;)Ljava/sql/Connection;

**sqlite-jdbc-3.8.11.2.jar, JDBC3PreparedStatement.class**
package org.sqlite.jdbc3
JDBC3PreparedStatement.unused () : SQLException
org/sqlite/jdbc3/JDBC3PreparedStatement.unused:()Ljava/sql/SQLException;

**sqlite-jdbc-3.8.11.2.jar, JDBC3ResultSet.class**
package org.sqlite.jdbc3
JDBC3ResultSet.checkCalendar ( Calendar cal ) : void
org/sqlite/jdbc3/JDBC3ResultSet.checkCalendar:(Ljava/util/Calendar;)V

**sqlite-jdbc-3.8.11.2.jar, JDBC3Statement.class**
package org.sqlite.jdbc3
JDBC3Statement.unused () : SQLException
org/sqlite/jdbc3/JDBC3Statement.unused:()Ljava/sql/SQLException;

**sqlite-jdbc-3.8.11.2.jar, JDBC4ResultSet.class**
package org.sqlite.jdbc4
JDBC4ResultSet.unused () : SQLException
org/sqlite/jdbc4/JDBC4ResultSet.unused:()Ljava/sql/SQLException;

**sqlite-jdbc-3.8.11.2.jar, NativeDB.class**
package org.sqlite.core
NativeDB.create_function ( String p1, Function p2 ) : int
org/sqlite/core/NativeDB.create_function:(Ljava/lang/String;Lorg/sqlite/Function;)I
NativeDB.NativeDB ()
org/sqlite/core/NativeDB."<init>"():V
NativeDB.value_bytes ( Function p1, int p2 ) : int
org/sqlite/core/NativeDB.value_bytes:(Lorg/sqlite/Function;I)I

**sqlite-jdbc-3.8.11.2.jar, PrimaryKeyFinder.class**
package org.sqlite.core
CoreDatabaseMetaData.PrimaryKeyFinder.CoreDatabaseMetaData.PrimaryKeyFinder ( CoreDatabaseMetaData table, String p2 )
org/sqlite/core/CoreDatabaseMetaData.PrimaryKeyFinder."<init>":(Lor
g/sqlite/core/CoreDatabaseMetaData;Ljava/lang/String;)V
CoreDatabaseMetaData.PrimaryKeyFinder.getColumns ( ) : String[]
org/sqlite/core/CoreDatabaseMetaData.PrimaryKeyFinder.getColumns ( ):[Ljava/lang/String;
CoreDatabaseMetaData.PrimaryKeyFinder.getName ( ) : String
org/sqlite/core/CoreDatabaseMetaData.PrimaryKeyFinder.getName:()Ljava/lang/String;

sqlite-jdbc-3.8.11.2.jar, SQLite.class
package org.sqlite
SQLite.main ( String[] p1 ) [static] : void
org/sqlite/SQLite.main:([Ljava/lang/String;)V
SQLite.SQLite ()
org/sqlite/SQLite."<init>";():V
SQLite.SQLite ( boolean p1 )
org/sqlite/SQLite."<init>";:(Z)V
SQLite.SQLite ( int p1, int p2 )
org/sqlite/SQLite."<init>";:(II)V
SQLite.SQLite ( int p1, int p2, boolean p3 )
org/sqlite/SQLite."<init>";:(IIZ)V

sqlite-jdbc-3.8.11.2.jar, SQLiteConfig.class
package org.sqlite
SQLiteConfig.setBusyTimeout ( String milliseconds ) : void
org/sqlite/SQLiteConfig.setBusyTimeout:(Ljava/lang/String;)V

sqlite-jdbc-3.8.11.2.jar, SQLiteDataSource.class
package org.sqlite
SQLiteDataSource.setCouncChanges ( boolean enable ) : void
org/sqlite/SQLiteDataSource.setCouncChanges:(Z)V
SQLiteDataSource.setEnforceForeinKeys ( boolean enforce ) : void
org/sqlite/SQLiteDataSource.setEnforceForeinKeys:(Z)V

sqlite-jdbc-3.8.11.2.jar, SQLitePooledConnection.class
package org.sqlite javax
SQLitePooledConnection.SQLitePooledConnection ( Connection physicalConn )
org/sqlite/javax/SQLitePooledConnection."<init>";:(Ljava/sql/Connection;)V

to the top
### Problems with Data Types, High Severity

**sqlite-jdbc-3.8.11.2.jar**

package org.sqlite

[+] class SQLite 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class has been removed. Recompilation of a client program may be terminated with the message: cannot find class SQLite.</td>
</tr>
</tbody>
</table>

[+] affected methods: 12 (1%)

**SQLite.SQLite ( )**

*This constructor is from 'SQLite' class.*

**SQLite.SQLite ( int p1, int p2 )**

*This constructor is from 'SQLite' class.*

**SQLite.SQLite ( int p1, int p2, boolean p3 )**

*This constructor is from 'SQLite' class.*

**SQLite.SQLite ( boolean p1 )**

*This constructor is from 'SQLite' class.*

**SQLite._execute ( )**

*This method is from 'SQLite' class.*

**SQLite.entryPoint ( )**

*This method is from 'SQLite' class.*

**SQLite.getCPUSState ( Runtime.CPUSState p1 )**

*This method is from 'SQLite' class.*

**SQLite.gp ( )**

*This method is from 'SQLite' class.*

**SQLite.heapStart ( )**

*This method is from 'SQLite' class.*
This method is from 'SQLite' class.

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Access level of field <code>busyTimeout</code> has been changed from public to private.</td>
<td>Recompilation of a client program may be terminated with the message: <code>busyTimeout</code> has private access in <code>SQLiteConfig</code>.</td>
</tr>
<tr>
<td>2</td>
<td>Field <code>dateClass</code> of type <code>SQLiteConfig.DateClass</code> has been removed from this class.</td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable <code>dateClass</code> in <code>SQLiteConfig</code>.</td>
</tr>
<tr>
<td>3</td>
<td>Field <code>dateMultiplier</code> of type <code>long</code> has been removed from this class.</td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable <code>dateMultiplier</code> in <code>SQLiteConfig</code>.</td>
</tr>
<tr>
<td>4</td>
<td>Field <code>datePrecision</code> of type <code>SQLiteConfig.DatePrecision</code> has been removed from this class.</td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable <code>datePrecision</code> in <code>SQLiteConfig</code>.</td>
</tr>
<tr>
<td>5</td>
<td>Field <code>dateStringFormat</code> of type <code>java.lang.String</code> has been removed from this class.</td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable <code>dateStringFormat</code> in <code>SQLiteConfig</code>.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>dateStringFormat in SQLiteConfig.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>type SQLiteConfig.Pragma has been removed</td>
<td>terminated with the message: cannot find variable READ_UNCOMMITTED in</td>
<td></td>
</tr>
<tr>
<td>from this class.</td>
<td>SQLiteConfig.Pragma.</td>
<td></td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)

**SQLiteConfig.Pragma.getPragmaName ( )**

*This method is from 'SQLiteConfig.Pragma' class.*

**SQLiteConfig.Pragma.valueOf ( String name )**

*This method is from 'SQLiteConfig.Pragma' class.*

**SQLiteConfig.Pragma.values ( )**

*This method is from 'SQLiteConfig.Pragma' class.*

**SQLiteConfig.setPragma ( SQLiteConfig.Pragma pragma, String value )**

*1st parameter 'pragma' of this method is of type 'SQLiteConfig.Pragma'.*

[+] class **SQLiteConnection** 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class became abstract.</td>
</tr>
</tbody>
</table>

[+] affected methods: 32 (2.6%)

**core.CoreDatabaseMetaData.CoreDatabaseMetaData ( SQLiteConnection conn )**

*1st parameter 'conn' of this method is of type 'SQLiteConnection'.*

**core.CoreDatabaseMetaData.checkOpen ( )**
```java
package org.sqlite.core

[+] class CoreConnection 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class has been removed.</td>
</tr>
</tbody>
</table>

Recompilation of a client program may be terminated with the message: cannot find class CoreConnection.

[+] affected methods: 12 (1%)

CoreConnection.CoreConnection ( String url, String fileName,
```
Properties prop )
   This constructor is from 'CoreConnection' abstract class.

CoreConnection.checkCursor ( int rst, int rsc, int rsh )
   This method is from 'CoreConnection' abstract class.

CoreConnection.checkOpen ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.close ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.db ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.finalize ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.getBusyTimeout ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.getDriverVersion ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.libversion ( )
   This method is from 'CoreConnection' abstract class.

CoreConnection.setBusyTimeout ( int milliseconds )
   This method is from 'CoreConnection' abstract class.

...

[+] class CoreDatabaseMetaData 2

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added super-interface java.sql.DatabaseMetaData.</td>
<td>Recompilation of a client program may be terminated with the message: a client class C is not abstract and does not override abstract method in java.sql.DatabaseMetaData.</td>
</tr>
</tbody>
</table>
### Change and Effect Table

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Field <code>refCount</code> of type <code>int</code> has been removed from this class. Recompilation of a client program may be terminated with the message: cannot find variable <code>refCount</code> in <code>CoreDatabaseMetaData</code>.</td>
</tr>
</tbody>
</table>

[+] affected methods: 7 (0.6%)

- `CoreDatabaseMetaData.CoreDatabaseMetaData (SQLiteConnection conn)`
  - This constructor is from `'CoreDatabaseMetaData' abstract class.`

- `CoreDatabaseMetaData.checkOpen ( )`
  - This method is from `'CoreDatabaseMetaData' abstract class.`

- `CoreDatabaseMetaData.close ( )`
  - This method is from `'CoreDatabaseMetaData' abstract class.`

- `CoreDatabaseMetaData.escape ( String val )`
  - This method is from `'CoreDatabaseMetaData' abstract class.`

- `CoreDatabaseMetaData.finalize ( )`
  - This method is from `'CoreDatabaseMetaData' abstract class.`

- `CoreDatabaseMetaData.getGeneratedKeys ( )`
  - This abstract method is from `'CoreDatabaseMetaData' abstract class.`

- `CoreDatabaseMetaData.quote ( String tableName )`
  - This method is from `'CoreDatabaseMetaData' abstract class.`

[+] class `CoreDatabaseMetaData.PrimaryKeyFinder` 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This class has been removed. Recompilation of a client program may be terminated with the message: cannot find class <code>CoreDatabaseMetaData.Predicate</code>.</td>
</tr>
</tbody>
</table>

[+] class `CoreDatabaseMetaData.Predicate` 1
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>primaryKeyFinder.</td>
</tr>
</tbody>
</table>

[+] affected methods: 3 (0.2%)

**CoreDatabaseMetaData.PrimaryKeyFinder.CoreDatabaseMetaData.PrimaryKeyFinder** (CoreDatabaseMetaData table, String p2)

*This constructor is from 'CoreDatabaseMetaData.PrimaryKeyFinder' class.*

**CoreDatabaseMetaData.PrimaryKeyFinder.getColumns** ()

*This method is from 'CoreDatabaseMetaData.PrimaryKeyFinder' class.*

**CoreDatabaseMetaData.PrimaryKeyFinder.getName** ()

*This method is from 'CoreDatabaseMetaData.PrimaryKeyFinder' class.*

[+] class **CoreResultSet** 2

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Type of field <code>maxRows</code> has been changed from <code>int</code> to <code>long</code>. Recompilation of a client program may be terminated with the message: incompatible types, found: <code>int</code>, required: <code>long</code>.</td>
</tr>
<tr>
<td>2</td>
<td>Field <code>db</code> of type <code>DB</code> has been removed from this class. Recompilation of a client program may be terminated with the message: cannot find variable <code>db</code> in <code>CoreResultSet</code>.</td>
</tr>
</tbody>
</table>

[+] affected methods: 19 (1.5%)

**CoreResultSet.CoreResultSet** (CoreStatement `stmt`)

*Field 'stmt.rs' in 1st parameter 'stmt' of this method is of type 'CoreResultSet'.*
CoreResultSet.addColumnIndexInCache ( String col, int index )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkMeta ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.close ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.findColumnIndexInCache ( String col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.isOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.markCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreStatement.CoreStatement ( SQLiteConnection c )
Field 'this.rs' in the object of this method is of type 'CoreResultSet'.

[+ class CoreStatement 3

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Type of field <strong>pointer</strong> has been changed from <strong>long</strong> to <strong>SafeStmtPtr</strong>.</td>
<td>Recompilation of a client program may be terminated with the message: incompatible types, found: <strong>long</strong>, required: <strong>SafeStmtPtr</strong>.</td>
</tr>
<tr>
<td>2</td>
<td>Field <strong>db</strong> of type <strong>DB</strong> has been removed from this class.</td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable <strong>db</strong> in</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>Field <strong>metadata</strong> of type <strong>CoreDatabaseMetaData</strong> has been removed from this class.</td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable <strong>metadata</strong> in <strong>CoreStatement</strong>.</td>
<td></td>
</tr>
</tbody>
</table>

[+] affected methods: 11 (0.9%)
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Abstract method <code>int backup(String, String, DB.ProgressObserver, int, int)</code> has been added to this class.</td>
</tr>
<tr>
<td>2</td>
<td>Abstract method <code>void busy_handler(BusyHandler)</code> has been added to this class.</td>
</tr>
<tr>
<td>3</td>
<td>Abstract method <code>long changes()</code> has been added to this class.</td>
</tr>
<tr>
<td>4</td>
<td>Abstract method <code>void clear_progress_handler()</code> has been added to this class.</td>
</tr>
<tr>
<td>Change (Int)</td>
<td>Effect</td>
</tr>
<tr>
<td>-------------</td>
<td>--------</td>
</tr>
<tr>
<td>5</td>
<td>Abstract method <code>int create_collation ( String, Collation )</code> has been added to this class. Recompilation of a client program may be terminated with the message: a client class C is not abstract and does not override abstract method <code>create_collation ( String, Collation )</code> in DB.</td>
</tr>
<tr>
<td>6</td>
<td>Abstract method <code>int create_function ( String, Function, int, int )</code> has been added to this class. Recompilation of a client program may be terminated with the message: a client class C is not abstract and does not override abstract method <code>create_function ( String, Function, int, int )</code> in DB.</td>
</tr>
<tr>
<td>7</td>
<td>Abstract method <code>int destroy_collation ( String )</code> has been added to this class. Recompilation of a client program may be terminated with the message: a client class C is not abstract and does not override abstract method <code>destroy_collation ( String )</code> in DB.</td>
</tr>
<tr>
<td>8</td>
<td>Abstract method <code>int limit ( int, int )</code> has been added to this class. Recompilation of a client program may be terminated with the message: a client class C is not abstract and does not override abstract method <code>limit ( int, int )</code> in DB.</td>
</tr>
<tr>
<td></td>
<td>Change</td>
</tr>
<tr>
<td>---</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td>Abstract method <code>SafeStmtPtr prepare ( String )</code> has been added to this class.</td>
</tr>
<tr>
<td>9</td>
<td>Abstract method <code>void register_progress_handler ( int, ProgressHandler )</code> has been added to this class.</td>
</tr>
<tr>
<td>10</td>
<td>Abstract method <code>int restore ( String, String, DB.ProgressObserver, int, int, int )</code> has been added to this class.</td>
</tr>
<tr>
<td>11</td>
<td>Abstract method <code>long total_changes ()</code> has been added to this class.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>13</strong></td>
<td>Abstract method <code>int changes()</code> has been removed from this class.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find method <code>changes()</code> in class <code>DB</code>.</td>
</tr>
<tr>
<td><strong>14</strong></td>
<td>Abstract method <code>int create_function(String, Function)</code> has been removed from this class.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find method <code>create_function(String, Function)</code> in class <code>DB</code>.</td>
</tr>
<tr>
<td><strong>15</strong></td>
<td>Abstract method <code>long prepare(String)</code> has been removed from this class.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find method <code>prepare(String)</code> in class <code>DB</code>.</td>
</tr>
<tr>
<td><strong>16</strong></td>
<td>Abstract method <code>int total_changes()</code> has been removed from this class.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find method <code>total_changes()</code> in class <code>DB</code>.</td>
</tr>
<tr>
<td><strong>17</strong></td>
<td>Abstract method <code>int value_bytes(Function, int)</code> has been removed from this class.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find method <code>value_bytes(Function, int)</code> in class <code>DB</code>.</td>
</tr>
</tbody>
</table>

[+] affected methods: 46 (3.7%)
DB._close ( )
This abstract method is from 'DB' abstract class.

DB._exec ( String p1 )
This abstract method is from 'DB' abstract class.

DB._open ( String p1, int p2 )
This abstract method is from 'DB' abstract class.

DB.backup ( String p1, String p2, DB.ProgressObserver p3 )
This abstract method is from 'DB' abstract class.

DB.busy_timeout ( int p1 )
This abstract method is from 'DB' abstract class.

DB.clear_bindings ( long p1 )
This abstract method is from 'DB' abstract class.

DB.close ( )
This method is from 'DB' abstract class.

DB.column_blob ( long p1, int p2 )
This abstract method is from 'DB' abstract class.

DB.column_count ( long p1 )
This abstract method is from 'DB' abstract class.

DB.column_dectype ( long p1, int p2 )
This abstract method is from 'DB' abstract class.

...
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>required: org.sqlite.SQLiteConnection.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Removed super-interface javax.sql.PooledConnection.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find method in class SQLitePooledConnection.</td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)

```java
SQLitePooledConnection.addConnectionEventListener ( 
ConnectionEventListener listener )
This method is from 'SQLitePooledConnection' class.

SQLitePooledConnection.close ( )
This method is from 'SQLitePooledConnection' class.

SQLitePooledConnection.getConnection ( )
This method is from 'SQLitePooledConnection' class.

SQLitePooledConnection.removeConnectionEventListener ( 
ConnectionEventListener listener )
This method is from 'SQLitePooledConnection' class.
```

### sqlite-jdbc-3.8.11.2.jar

**package org.sqlite.jdbc3**

[+] **class JDBC3DatabaseMetaData** 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field FK_NAMED_PATTERN of type java.util.regex.Pattern has been removed</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot find variable</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>from this class.</td>
<td>FK_NAMED_PATTERN in JDBC3DatabaseMetaData</td>
</tr>
</tbody>
</table>

[+] affected methods: 170 (13.6%)

```java
JDBC3DatabaseMetaData.JDBC3DatabaseMetaData (SQLiteConnection conn )
   This constructor is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.allProceduresAreCallable ( )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.allTablesAreSelectable ( )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.createStruct ( String t, Object[ ] attr )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.dataDefinitionCausesTransactionCommit ( )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.dataDefinitionIgnoredInTransactions ( )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.deletesAreDetected ( int type )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.doesMaxRowSizeIncludeBlobs ( )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.finalize ( )
   This method is from 'JDBC3DatabaseMetaData' abstract class.

JDBC3DatabaseMetaData.getAttributes ( String c, String s, String t, String a )
   This method is from 'JDBC3DatabaseMetaData' abstract class.
...
sqlite-jdbc-3.8.11.2.jar
package org.sqlite.jdbc4
[+] class JDBC4Connection

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed super-interface java.sql.Connection.</td>
</tr>
</tbody>
</table>

[+] affected methods: 16 (1.3%)

JDBC4Connection.JDBC4Connection ( String url, String fileName, Properties prop )
   This constructor is from 'JDBC4Connection' abstract class.

JDBC4Connection.createArrayOf ( String typeName, Object[ ] elements )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.createBlob ( )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.createClob ( )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.createNClob ( )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.createSQLXML ( )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.createStatement ( int rst, int rsc, int rsh )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.getClientInfo ( )
   This method is from 'JDBC4Connection' abstract class.

JDBC4Connection.getClientInfo ( String name )
   This method is from 'JDBC4Connection' abstract class.
JDBC4Connection.isClosed ( )

This method is from 'JDBC4Connection' abstract class.

...
This method is from 'JDBC4DatabaseMetaData' class.

**JDBC4DatabaseMetaData.getSchemas** ( String catalog, String schemaPattern )  
This method is from 'JDBC4DatabaseMetaData' class.

**JDBC4DatabaseMetaData.isWrapperFor** ( Class<?> iface )  
This method is from 'JDBC4DatabaseMetaData' class.

**JDBC4DatabaseMetaData.supportsStoredFunctionsUsingCallableSyntax** ( )  
This method is from 'JDBC4DatabaseMetaData' class.

...
Problems with Methods, Medium Severity 50

SQLiteConnection

sqlite-jdbc-3.8.11.2.jar, DB.class
package org.sqlite.core

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from int to long.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td>types: <strong>long</strong> cannot be converted to <strong>int</strong>.</td>
</tr>
</tbody>
</table>

```java
[+] DB.executeUpdate ( CoreStatement stmt, Object[] vals ) : int
org/sqlite/core/DB.executeUpdate:(Lorg/sqlite/core/CoreStatement;[Ljava/lang/Object;)I
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <strong>int</strong> to <strong>long</strong>.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: incompatible types: <strong>long</strong> cannot be converted to <strong>int</strong>.</td>
</tr>
</tbody>
</table>

```java
[+] DB.newSQLException ( int errorCode, String errorMessage ) [static] : SQLException
org/sqlite/core/DB.newSQLException:(ILjava/lang/String;)Ljava/sql/SQLException;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <strong>java.sql.SQLException</strong> to <strong>org.sqlite.SQLiteException</strong>.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: incompatible types: <strong>org.sqlite.SQLiteException</strong> cannot be converted to <strong>java.sql.SQLException</strong>.</td>
</tr>
</tbody>
</table>

```java
[+] DB.prepare ( String p1 ) [abstract] : long
org/sqlite/core/DB.prepare:(Ljava/lang/String;)J
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from <strong>long</strong> to <strong>SafeStmtPtr</strong>.</td>
</tr>
</tbody>
</table>
|        | Recompilation of a client program may be terminated with the message: incompatible types: **SafeStmtPtr** cannot be converted to
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.total_changes</td>
<td>long.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Return value type has been changed from int to long.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: incompatible types: long cannot be converted to int.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>JDBC.createConnection</td>
<td>boolean</td>
</tr>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>override isReadOnly () in JDBC3Connection; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
</tbody>
</table>

**sqlite-jdbc-3.8.11.2.jar. JDBC3DatabaseMetaData.class**

package org.sqlite.jdbc3

[+] JDBC3DatabaseMetaData.getDatabaseMajorVersion () : int

org/sqlite/jdbc3/JDBC3DatabaseMetaData.getDatabaseMajorVersion:()Z

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Added java.sql.SQLException exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: unreported exception java.sql.SQLException must be caught or declared to be thrown.</td>
</tr>
</tbody>
</table>

[+] JDBC3DatabaseMetaData.getDatabaseMinorVersion () : int

org/sqlite/jdbc3/JDBC3DatabaseMetaData.getDatabaseMinorVersion:()Z

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Added java.sql.SQLException exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: unreported exception java.sql.SQLException must be caught or declared to be thrown.</td>
</tr>
</tbody>
</table>

[+] JDBC3DatabaseMetaData.supportsFullOuterJoins () : boolean

org/sqlite/jdbc3/JDBC3DatabaseMetaData.supportsFullOuterJoins:()Z
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added java.sql.SQLException exception thrown. Recompilation of a client program may be terminated with the message: unreported exception java.sql.SQLException must be caught or declared to be thrown.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, JDBC3PreparedStatement.class

package org.sqlite.jdbc3

[+] JDBC3PreparedStatement.getParameterType ( int pos ) : int

org/sqlite/jdbc3/JDBC3PreparedStatement.getParameterType:(l)

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added java.sql.SQLException exception thrown. Recompilation of a client program may be terminated with the message: unreported exception java.sql.SQLException must be caught or declared to be thrown.</td>
</tr>
</tbody>
</table>

[+] JDBC3PreparedStatement.getParameterTypeName ( int pos ) : String

org/sqlite/jdbc3/JDBC3PreparedStatement.getParameterTypeName:(l)

Ljava/lang/String;

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Added java.sql.SQLException exception thrown. Recompilation of a client program may be terminated with the message: unreported exception java.sql.SQLException must be caught or declared to be thrown.</td>
</tr>
</tbody>
</table>
### sqlite-jdbc-3.8.11.2.jar, JDBC3ResultSet.class
package org.sqlite.jdbc3

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown. Recompilation of a client program may be terminated with the message: cannot override <code>clearWarnings</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
</tr>
</tbody>
</table>

### JDBC3ResultSet.getColumnNameDisplaySize (int col) : int

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown. Recompilation of a client program may be terminated with the message: cannot override <code>getColumnNameDisplaySize</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
</tr>
</tbody>
</table>

### JDBC3ResultSet.getConcurrency () : int

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown. Recompilation of a client program may be terminated with the message: cannot override <code>getConcurrency</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</strong></td>
<td></td>
</tr>
</tbody>
</table>

**JDBC3ResultSet.getCursorName() : String**

```java
org/sqlite/jdbc3/JDBC3ResultSet.getCursorName:()Ljava/lang/String;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override getCursorName () in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

**JDBC3ResultSet.getFetchSize() : int**

```java
org/sqlite/jdbc3/JDBC3ResultSet.getFetchSize:()I
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override getFetchSize () in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

**JDBC3ResultSet.getMetaData() : ResultSetMetaData**

```java
org/sqlite/jdbc3/JDBC3ResultSet.getMetaData:()Ljava/sql/ResultSetMetaData;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed ResultSetMetaData</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>java.sql.SQLException exception thrown.</td>
<td>client program may be terminated with the message: cannot override getMetaData() in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

```java
[+] JDBC3ResultSet.getRow() : int
org/sqlite/jdbc3/JDBC3ResultSet.getRow():I
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override getRow() in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

```java
[+] JDBC3ResultSet.getSchemaName(int col) : String
org/sqlite/jdbc3/JDBC3ResultSet.getSchemaName:(I)Ljava/lang/String;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override getSchemaName(int) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

```java
[+] JDBC3ResultSet.getType() : int
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override getType() in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: cannot override <code>getType()</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
</tr>
</tbody>
</table>

```
[+] JDBC3ResultSet.getWarnings() : SQLWarning
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: cannot override <code>getWarnings()</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
</tr>
</tbody>
</table>

```
[+] JDBC3ResultSet.isAfterLast() : boolean
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: cannot override <code>isAfterLast()</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
</tr>
<tr>
<td>Method</td>
<td>Change</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>JDBC3ResultSet.isBeforeFirst() : boolean</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>JDBC3ResultSet.isCaseSensitive(int col) : boolean</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>JDBC3ResultSet.isCurrency(int col) : boolean</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
</tbody>
</table>
### JDBC3ResultSet.isDefinitelyWritable

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: cannot override isDefinitelyWritable(int) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

### JDBC3ResultSet.isFirst

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: cannot override isFirst() in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

### JDBC3ResultSet.isReadOnly

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td>Recompilation of a client program may be terminated with the message: cannot override isReadOnly(int) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>int ) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
</tbody>
</table>

## JDBC3ResultSet.isSearchable ( int col ) : boolean

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>Recompilation of a client program may be terminated with the message: cannot override isSearchable ( int ) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
</tbody>
</table>

## JDBC3ResultSet.isWritable ( int col ) : boolean

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>Recompilation of a client program may be terminated with the message: cannot override isWritable ( int ) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
</tbody>
</table>

## JDBC3ResultSet.rowDeleted ( ) : boolean

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td>Recompilation of a client program may be terminated with the message: cannot override rowDeleted ( ) in JDBC3ResultSet; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>message: cannot override <code>rowDeleted()</code> in <code>JDBC3ResultSet</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
<td></td>
</tr>
</tbody>
</table>

```
[+] JDBC3ResultSet.rowInserted() : boolean 1
org/sqlite/jdbc3/JDBC3ResultSet.rowInserted():Z
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
<td></td>
</tr>
</tbody>
</table>

Recompilation of a client program may be terminated with the message: cannot override `rowInserted()` in `JDBC3ResultSet`; overridden method does not throw `java.sql.SQLException`.

```
[+] JDBC3ResultSet.rowUpdated() : boolean 1
org/sqlite/jdbc3/JDBC3ResultSet.rowUpdated():Z
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
<td></td>
</tr>
</tbody>
</table>

Recompilation of a client program may be terminated with the message: cannot override `rowUpdated()` in `JDBC3ResultSet`; overridden method does not throw `java.sql.SQLException`.

```
[+] JDBC3Statement.setEscapeProcessing(boolean enable) : void 1
org/sqlite/jdbc3/JDBC3Statement.setEscapeProcessing:(Z)V
```
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown. Recompilation of a client program may be terminated with the message: cannot override setEscapeProcessing ( boolean ) in JDBC3Statement; overridden method does not throw java.sql.SQLException.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>override <code>unwrap (Class&lt;T&gt;)</code> in <code>JDBC4Connection</code>; overridden method does not throw <code>java.sql.SQLException</code>.</td>
<td></td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, `JDBC4DatabaseMetaData.class`

package `org.sqlite.jdbc4`

```java
[+] JDBC4DatabaseMetaData.isWrapperFor (Class<?> iface) : boolean
```

```java
org/sqlite/jdbc4/JDBC4DatabaseMetaData.isWrapperFor:(Ljava/lang/Class;)Z
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
</tr>
</tbody>
</table>

Recompilation of a client program may be terminated with the message: cannot override `isWrapperFor (Class<?>)` in `JDBC4DatabaseMetaData`; overridden method does not throw `java.sql.SQLException`.

```java
[+] JDBC4DatabaseMetaData.unwrap (Class<T> iface) : T
```

```java
org/sqlite/jdbc4/JDBC4DatabaseMetaData.unwrap:(Ljava/lang/Class;)Ljava/lang/Object;
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed <code>java.sql.SQLException</code> exception thrown.</td>
</tr>
</tbody>
</table>

Recompilation of a client program may be terminated with the message: cannot override `unwrap (Class<T>)` in `JDBC4DatabaseMetaData`; overridden method does not throw `java.sql.SQLException`. |
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, JDBC4ResultSet.class
package org.sqlite.jdbc4

[+] JDBC4ResultSet.isWrapperFor (Class<? iface) : boolean

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override isWrapperFor (Class&lt;?&gt;) in JDBC4ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

[+] JDBC4ResultSet.unwrap (Class<T iface) : T

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Removed java.sql.SQLException exception thrown.</td>
</tr>
<tr>
<td></td>
<td>Recompilation of a client program may be terminated with the message: cannot override unwrap (Class&lt;T&gt;) in JDBC4ResultSet; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, JDBC4Statement.class
package org.sqlite.jdbc4
### JDBC4Statement.isClosed()

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed</td>
<td>Recompilation of a client program may be terminated with the message: cannot override isClosed() in JDBC4Statement; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

### JDBC4Statement.isWrapperFor(Class<?> iface)

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed</td>
<td>Recompilation of a client program may be terminated with the message: cannot override isWrapperFor(Class&lt;?&gt;) in JDBC4Statement; overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

### JDBC4Statement.unwrap(Class<T> iface)

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed</td>
<td>Recompilation of a client program may be terminated with the message: cannot override unwrap(Class&lt;T&gt;) in JDBC4Statement; overridden method does not throw java.sql.SQLException.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td>overridden method does not throw java.sql.SQLException.</td>
</tr>
</tbody>
</table>

sqlite-jdbc-3.8.11.2.jar, **NativeDB.class**

package `org.sqlite.core`

[+] `NativeDB.changes ( ) : int 1`

`org/sqlite/core/NativeDB.changes():I`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
</table>
| 1      | Return value type has been changed from `int` to `long`. Recompilation of a client program may be terminated with the message: incompatible types: `long` cannot be converted to `int`.

[+] `NativeDB.destroy_function ( String p1 ) : int 1`

`org/sqlite/core/NativeDB.destroy_function:(Ljava/lang/String;)I`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
</table>
| 1      | Added `java.sql.SQLException` exception thrown. Recompilation of a client program may be terminated with the message: unreported exception `java.sql.SQLException` must be caught or declared to be thrown.

[+] `NativeDB.prepare ( String p1 ) : long 1`

`org/sqlite/core/NativeDB.prepare:(Ljava/lang/String;)J`

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
</table>
| 1      | Return value type has been changed from `long` to `SafeStmtPtr`. Recompilation of a client program may be terminated with the message: incompatible types: `SafeStmtPtr`
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>cannot be converted to long.</td>
<td></td>
</tr>
</tbody>
</table>

**sqlite-jdbc-3.8.11.2.jar, SQLiteConfig.class**

```java
package org.sqlite;

 [+] SQLiteConfig.setDatePrecision ( String datePrecision ) : void
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td></td>
</tr>
<tr>
<td>Recompilation of a client program may be terminated with the message: cannot override <code>setDatePrecision ( String )</code> in SQLiteConfig; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
</tbody>
</table>

**sqlite-jdbc-3.8.11.2.jar, SQLiteDataSource.class**

```java
package org.sqlite;

 [+] SQLiteDataSource.getConnection ( String username, String password ) : Connection
```

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Removed java.sql.SQLException exception thrown.</td>
<td></td>
</tr>
<tr>
<td>Recompilation of a client program may be terminated with the message: cannot override <code>getConnection ( String username, String password )</code> in SQLiteDataSource; overridden method does not throw java.sql.SQLException.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>Return value type has been changed from java.sql.Connection to SQLiteConnection. Recompilation of a client program may be terminated with the message: incompatible types: SQLiteConnection cannot be converted to java.sql.Connection.</td>
</tr>
</tbody>
</table>

Problems with Data Types, Low Severity

sqlite-jdbc-3.8.11.2.jar
package org.sqlite.jdbc3
[+] class JDBC3Connection 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Superclass has been changed from org.sqlite.core.CoreConnection to org.sqlite.SQLiteConnection. 1) Recompilation of a client program may be terminated with the message: cannot find variable (or method) in JDBC3Connection. 2) A static field from a super-interface of a client class may hide a field (with the same name) inherited from new super-class. Recompilation of a client class may be terminated with the message: reference to variable is ambiguous.</td>
</tr>
</tbody>
</table>

[+] affected methods: 29 (2.3%)
JDBC3Connection.JDBC3Connection(String url, String fileName, Properties prop)
This constructor is from 'JDBC3Connection' abstract class.

JDBC3Connection.clearWarnings()
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.createStatement()
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.createStatement(int rsType, int rsConcurr)
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.createStatement(int p1, int p2, int p3)
This abstract method is from 'JDBC3Connection' abstract class.

JDBC3Connection.createStruct(String t, Object[] attr)
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.getCatalog()
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.getHoldability()
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.getTypeMap()
This method is from 'JDBC3Connection' abstract class.

JDBC3Connection.getWarnings()
This method is from 'JDBC3Connection' abstract class.

...
Field `FLAG_DETERMINISTIC` has been added to this class.

No effect.

[+] affected methods: 30 (2.4%)

```c
core.DB.value_blob ( Function p1, int p2 )
1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_double ( Function p1, int p2 )
1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_int ( Function p1, int p2 )
1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_long ( Function p1, int p2 )
1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_text ( Function p1, int p2 )
1st parameter 'p1' of this abstract method is of type 'Function'.

core.DB.value_type ( Function p1, int p2 )
1st parameter 'p1' of this abstract method is of type 'Function'.

core.NativeDB.value_blob ( Function p1, int p2 )
1st parameter 'p1' of this method is of type 'Function'.

core.NativeDB.value_double ( Function p1, int p2 )
1st parameter 'p1' of this method is of type 'Function'.

core.NativeDB.value_int ( Function p1, int p2 )
1st parameter 'p1' of this method is of type 'Function'.

core.NativeDB.value_long ( Function p1, int p2 )
1st parameter 'p1' of this method is of type 'Function'.

...
```

[+] class SQLiteConfig 1
<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field <code>busyTimeout</code> became non-final.</td>
</tr>
</tbody>
</table>

[+] affected methods: 48 (3.9%)

```
javax.SQLiteConnectionPoolDataSource.SQLiteConnectionPoolDataSource ( SQLiteConfig `config` )
  1st parameter 'config' of this method is of type 'SQLiteConfig'.

SQLiteConfig.SQLiteConfig ( )
  This constructor is from 'SQLiteConfig' class.

SQLiteConfig.SQLiteConfig ( Properties `prop` )
  This constructor is from 'SQLiteConfig' class.

SQLiteConfig.apply ( Connection `conn` )
  This method is from 'SQLiteConfig' class.

SQLiteConfig.createConnection ( String `url` )
  This method is from 'SQLiteConfig' class.

SQLiteConfig.enableCaseSensitiveLike ( boolean `enable` )
  This method is from 'SQLiteConfig' class.

SQLiteConfig.enableCountChanges ( boolean `enable` )
  This method is from 'SQLiteConfig' class.

SQLiteConfig.enableEmptyResultSetCallBacks ( boolean `enable` )
  This method is from 'SQLiteConfig' class.

SQLiteConfig.enableFullColumnNames ( boolean `enable` )
  This method is from 'SQLiteConfig' class.

SQLiteConfig.enableFullSync ( boolean `enable` )
  This method is from 'SQLiteConfig' class.

... 
```

[+] class `SQLiteConfig.Pragma` 22

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field <code>APPLICATION_ID</code> has been added to this</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>class.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Field <code>DEFER_FOREIGN_KEYS</code> has been added to this class.</td>
</tr>
<tr>
<td>3</td>
<td>Field <code>HEXKEY_MODE</code> has been added to this class.</td>
</tr>
<tr>
<td>4</td>
<td>Field <code>JDBC_EXPLICIT_READONLY</code> has been added to this class.</td>
</tr>
<tr>
<td>5</td>
<td>Field <code>LEGACY_ALTER_TABLE</code> has been added to this class.</td>
</tr>
<tr>
<td>6</td>
<td>Field <code>LIMIT_ATTACHED</code> has been added to this class.</td>
</tr>
<tr>
<td>7</td>
<td>Field <code>LIMIT_COLUMN</code> has been added to this class.</td>
</tr>
<tr>
<td>8</td>
<td>Field <code>LIMIT_COMPOUND_SELECT</code> has been added to this class.</td>
</tr>
<tr>
<td>9</td>
<td>Field <code>LIMIT_EXPR_DEPTH</code> has been added to this class.</td>
</tr>
<tr>
<td>10</td>
<td>Field <code>LIMIT_FUNCTION_ARG</code> has been added to this class.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>class.</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Field <code>LIMIT_LENGTH</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>12</td>
<td>Field <code>LIMIT_LIKE_PATTERN_LENGTH</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>13</td>
<td>Field <code>LIMIT_PAGE_COUNT</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>14</td>
<td>Field <code>LIMIT_SQL_LENGTH</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>15</td>
<td>Field <code>LIMIT_TRIGGER_DEPTH</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>16</td>
<td>Field <code>LIMIT_VARIABLE_NUMBER</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>17</td>
<td>Field <code>LIMIT_VDBE_OP</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>18</td>
<td>Field <code>LIMIT_WORKER_THREADS</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>19</td>
<td>Field <code>MMAP_SIZE</code> has been added to this class. No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>class.</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Field <strong>PASSWORD</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>21</td>
<td>Field <strong>READ_UNCOMMITTED</strong> has been added to this class. No effect.</td>
</tr>
<tr>
<td>22</td>
<td>Field <strong>SECURE_DELETE</strong> has been added to this class. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%)

**SQLiteConfig.Pragma.getPragmaName**( )  
*This method is from 'SQLiteConfig.Pragma' class.*

**SQLiteConfig.Pragma.valueOf**( String name )  
*This method is from 'SQLiteConfig.Pragma' class.*

**SQLiteConfig.Pragma.values**( )  
*This method is from 'SQLiteConfig.Pragma' class.*

**SQLiteConfig.setPragma**( SQLiteConfig.Pragma pragma, String value )  
1st parameter 'pragma' of this method is of type 'SQLiteConfig.Pragma'.

[+] class **SQLiteConfig.TransactionMode** [ ]

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field <strong>DEFERRED</strong> has been added to this class. No effect.</td>
</tr>
</tbody>
</table>

[+] affected methods: 6 (0.5%)

**SQLiteConfig.getTransactionMode**( )
Return value of this method is of type 'SQLiteConfig.TransactionMode'.

SQLiteConfig.setTransactionMode ( SQLiteConfig.TransactionMode transactionMode )
1st parameter 'transactionMode' of this method is of type 'SQLiteConfig.TransactionMode'.

SQLiteConfig.TransactionMode.getMode ( String mode )
This method is from 'SQLiteConfig.TransactionMode' class.

SQLiteConfig.TransactionMode.getValue ( )
This method is from 'SQLiteConfig.TransactionMode' class.

SQLiteConfig.TransactionMode.valueOf ( String name )
This method is from 'SQLiteConfig.TransactionMode' class.

SQLiteConfig.TransactionMode.values ( )
This method is from 'SQLiteConfig.TransactionMode' class.

[+] class SQLiteErrorCode  76

<table>
<thead>
<tr>
<th></th>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field SQLITE_ABORT_ROLLBACK has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>2</td>
<td>Field SQLITE_AUTH_USER has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>3</td>
<td>Field SQLITE_BUSY_RECOVERY has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>4</td>
<td>Field SQLITE_BUSY_SNAPSHOT has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>5</td>
<td>Field</td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>SQLITE_BUSY_TIMEOUT has been added to this class.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CANTOPEN_CONVPATH has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CANTOPEN_DIRTYWAL has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CANTOPEN_FULLPATH has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CANTOPEN_DIRIR has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CANTOPEN_NOTEMPDIR has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CANTOPEN_SYMLINK has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CONSTRAINT_CHECK has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field SQLITE_CONSTRAINT_COMMITHOOK has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>14</strong> Field SQLITE_CONSTRAINT_DATATYPE has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>15</strong> Field SQLITE_CONSTRAINT_FOREIGNKEY has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>16</strong> Field SQLITE_CONSTRAINT_FUNCTION has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>17</strong> Field SQLITE_CONSTRAINT_NOTNULL has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>18</strong> Field SQLITE_CONSTRAINT_PINNED has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>19</strong> Field SQLITE_CONSTRAINT_PRIMARYKEY has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>20</strong> Field SQLITE_CONSTRAINT_RowID has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>21</strong> Field SQLITE_CONSTRAINT_TRIGGER has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>22</strong> Field</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td><code>SQLITE_CONSTRAINT_UNIQUE</code> has been added to this class.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>23</strong></td>
<td>Field <code>SQLITE_CONSTRAINT_VTAB</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>24</strong></td>
<td>Field <code>SQLITE_CORRUPT_INDEX</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>25</strong></td>
<td>Field <code>SQLITE_CORRUPT_SEQUENCE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>26</strong></td>
<td>Field <code>SQLITE_CORRUPT_VTAB</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>27</strong></td>
<td>Field <code>SQLITE_ERROR_MISSING_COLLSEQ</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>28</strong></td>
<td>Field <code>SQLITE_ERROR_RETRY</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>29</strong></td>
<td>Field <code>SQLITE_ERROR_SNAPSHOT</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td><strong>30</strong></td>
<td>Field <code>SQLITE_IOERR_ACCESS</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>Field <code>SQLITE_IOERR_AUTH</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>32</td>
<td>Field <code>SQLITE_IOERR_BEGIN_ATOMIC</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>33</td>
<td>Field <code>SQLITE_IOERR_BLOCKED</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>34</td>
<td>Field <code>SQLITE_IOERR_CHECKRESERVEDLOCK</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>35</td>
<td>Field <code>SQLITE_IOERR_CLOSE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>36</td>
<td>Field <code>SQLITE_IOERR_COMMIT_ATOMIC</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>37</td>
<td>Field <code>SQLITE_IOERR_CONVERTPATH</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>38</td>
<td>Field <code>SQLITE_IOERR_CORRUPTFS</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>39</td>
<td>Field <code>SQLITE_IOERR_DATA</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>40</td>
<td>Field <code>SQLITE_IOERR_DELETE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>41</td>
<td>Field <code>SQLITE_IOERR_DELETE_NOENT</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>42</td>
<td>Field <code>SQLITE_IOERR_DIR_CLOSE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>43</td>
<td>Field <code>SQLITE_IOERR_DIR_FSYNC</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>44</td>
<td>Field <code>SQLITE_IOERR_FSTAT</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>45</td>
<td>Field <code>SQLITE_IOERR_FSYNC</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>46</td>
<td>Field <code>SQLITE_IOERR_GETTEMP</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>47</td>
<td>Field <code>SQLITE_IOERR_LOCK</code></td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>has been added to this class.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_MMAP</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_NOMEM</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_RDLOCK</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_READ</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_ROLLBACK_ATOMIC</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_SEEK</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_SHMLOCK</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td>Field <code>SQLITE_IOERR_SHMMAP</code> has been added to this class.</td>
<td>No effect.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------------------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>56</td>
<td>Field <code>SQLITE_IOERR_SHMOPEN</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>57</td>
<td>Field <code>SQLITE_IOERR_SHMSIZE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>58</td>
<td>Field <code>SQLITE_IOERR_SHORT_READ</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>59</td>
<td>Field <code>SQLITE_IOERR_TRUNCATE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>60</td>
<td>Field <code>SQLITE_IOERR_UNLOCK</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>61</td>
<td>Field <code>SQLITE_IOERR_VNODE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>62</td>
<td>Field <code>SQLITE_IOERR_WRITE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>63</td>
<td>Field <code>SQLITE_LOCKED_SHAREDCACHE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>64</td>
<td>Field <code>SQLITE_LOCKED_VTAB</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td></td>
<td>has been added to this class.</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>Field <code>SQLITE_NOTICE</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>66</td>
<td>Field <code>SQLITE_NOTICE_RECOV_ER_ROLLBACK</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>67</td>
<td>Field <code>SQLITE_NOTICE_RECOV_ER_WAL</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>68</td>
<td>Field <code>SQLITE_OK_LOAD_PERMANENTLY</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>69</td>
<td>Field <code>SQLITE_READONLY_CANTINIT</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>70</td>
<td>Field <code>SQLITE_READONLY_CANTLOCK</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>71</td>
<td>Field <code>SQLITE_READONLY_DBMOVED</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>72</td>
<td>Field <code>SQLITE_READONLY_DIRECTORY</code> has been added to this class.</td>
<td>No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td><strong>73</strong></td>
<td>Field SQLITE_READONLY_RECOVERY has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>74</strong></td>
<td>Field SQLITE_READONLY_ROLLBACK has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>75</strong></td>
<td>Field SQLITE_WARNING has been added to this class. No effect.</td>
<td></td>
</tr>
<tr>
<td><strong>76</strong></td>
<td>Field SQLITE_WARNING_AUTOINDEX has been added to this class. No effect.</td>
<td></td>
</tr>
</tbody>
</table>

[+] affected methods: 4 (0.3%) |

SQLiteErrorCode.getErrorCode ( int errorCode )
*This method is from 'SQLiteErrorCode' class.*

SQLiteErrorCode.toString ( )
*This method is from 'SQLiteErrorCode' class.*

SQLiteErrorCode.valueOf ( String name )
*This method is from 'SQLiteErrorCode' class.*

SQLiteErrorCode.values ( )
*This method is from 'SQLiteErrorCode' class.*

[sqlite-jdbc-3.8.11.2.jar]
package org.sqlite.core
[+] class CorePreparedStatement 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>Field batchQueryCount No effect.</td>
</tr>
<tr>
<td>Change</td>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>has been added to this class.</td>
<td></td>
</tr>
</tbody>
</table>

[+] affected methods: 3 (0.2%)

CorePreparedStatement.CorePreparedStatement ( SQLiteConnection conn, String sql )
This constructor is from 'CorePreparedStatement' abstract class.

CorePreparedStatement.batch ( int pos, Object value )
This method is from 'CorePreparedStatement' abstract class.

CorePreparedStatement.executeBatch ( )
This method is from 'CorePreparedStatement' abstract class.

[+] class CoreResultSet 2

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field emptyResultSet has been added to this class.</td>
</tr>
<tr>
<td>2</td>
<td>Field pastLastRow has been added to this class.</td>
</tr>
</tbody>
</table>

[+] affected methods: 19 (1.5%)

CoreResultSet.CoreResultSet ( CoreStatement stmt )
Field 'stmt.rs' in 1st parameter 'stmt' of this method is of type 'CoreResultSet'.

CoreResultSet.addColumnIndexInCache ( String col, int index )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.checkMeta ( )
This method is from 'CoreResultSet' abstract class.
CoreResultSet.checkOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.close ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.findColumnIndexInCache ( String col )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.isOpen ( )
This method is from 'CoreResultSet' abstract class.

CoreResultSet.markCol ( int col )
This method is from 'CoreResultSet' abstract class.

CoreStatement.CoreStatement ( SQLiteConnection c )
Field 'this.rs' in the object of this method is of type 'CoreResultSet'.

sqlite-jdbc-3.8.11.2.jar
package org.sqlite.jdbc3
[+] class JDBC3Statement 1

<table>
<thead>
<tr>
<th>Change</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Field updateCount has been added to this class.</td>
</tr>
</tbody>
</table>

[+] affected methods: 39 (3.1%)

JDBC3Statement.JDBC3Statement ( SQLiteConnection conn )
This constructor is from 'JDBC3Statement' abstract class.

JDBC3Statement.addBatch ( String sql )
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.cancel ( )
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.clearBatch ( )
This method is from 'JDBC3Statement' abstract class.

JDBC3Statement.clearWarnings ( )
This method is from 'JDBC3Statement' abstract class.

**JDBC3Statement.close ( )**
This method is from 'JDBC3Statement' abstract class.

**JDBC3Statement.execute ( String sql )**
This method is from 'JDBC3Statement' abstract class.

**JDBC3Statement.execute ( String sql, int[ ] colinds )**
This method is from 'JDBC3Statement' abstract class.

**JDBC3Statement.execute ( String sql, String[ ] colnames )**
This method is from 'JDBC3Statement' abstract class.

**JDBC3Statement.execute ( String sql, int autokeys )**
This method is from 'JDBC3Statement' abstract class.

...
This method is from 'OSInfo' class.