Faculty of Engineering, Alexandria University
Computers and Systems Engineering Department
Second year: (CS 221) Programming II
Assignment IV - Report

JDBC API

Made by:

Bishoy Nader Fathy (22)

Amr Mohamed NasrEldine (46)

Marc Magdi Kamel (52)

Michael Raafat Mikhail (55)

I. <u>Design Description</u>

We divided the application into some main packages:

1- **console**: containing the main class to run the application it uses the DBMS directly.

2- **commandLineInterface:** is the cli which uses only the driver implemented without the usage of any of the DBMS internal components.

3- **consolePortal:** contains the jdbcAdapter which connects the driver implementation with the DBMS so it's not connected directly to our implementation, not strongly connected to the parser or the engine.

4- **jdbc:** contains classes that implements the required interfaces (ResultSet, Driver … etc).

5- **parser**: responsible for parsing a given string and returns a strongly datatype query or a syntax error.

6- **syntaxManagement:** contains pattern manager class for each type of queries.

7- **queryBuilders:** contains the query factory class and the query build unit.

8- **queries**: contains classes for every query type (insertQuery, DatabseCreationQuery, ...).

9- **databaseManagement**: containing the engine class which performs the given query.

10- **actionManagement:** contains action taker class for each query which does this specific query.

11- **fileManager:** responsible for directories and Xml files reading, writing and deleting.

12- **fileManager.savableModels:** contains models of the serialized models to be saved.

13- **models**: contains interfaces for the models (Database, Table, Row …).

14- **models.implementation**: containing implementation for models interfaces.

15- **logicalComponents:** contains models that are used by the condition service.

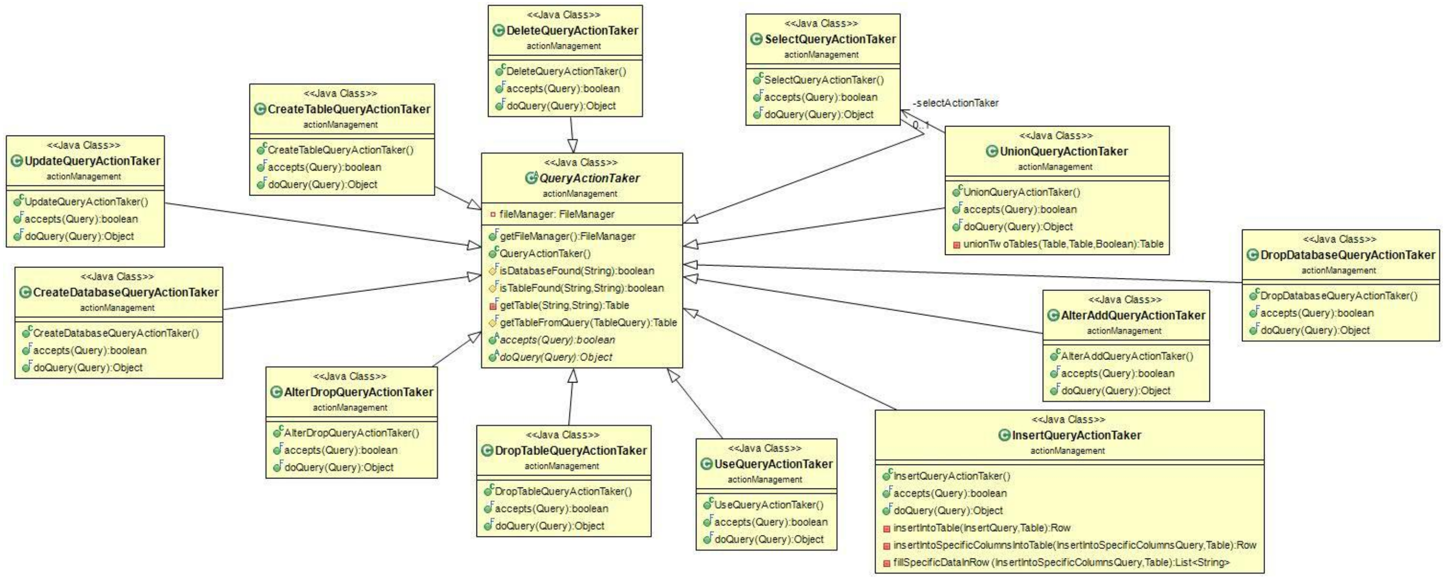
16- **conditionService**: responsible for processing simple conditions like >, < or = and complex ones like AND or OR.

17- **printingService:** contains classes responsible for printing table or a resultset in a readable way.

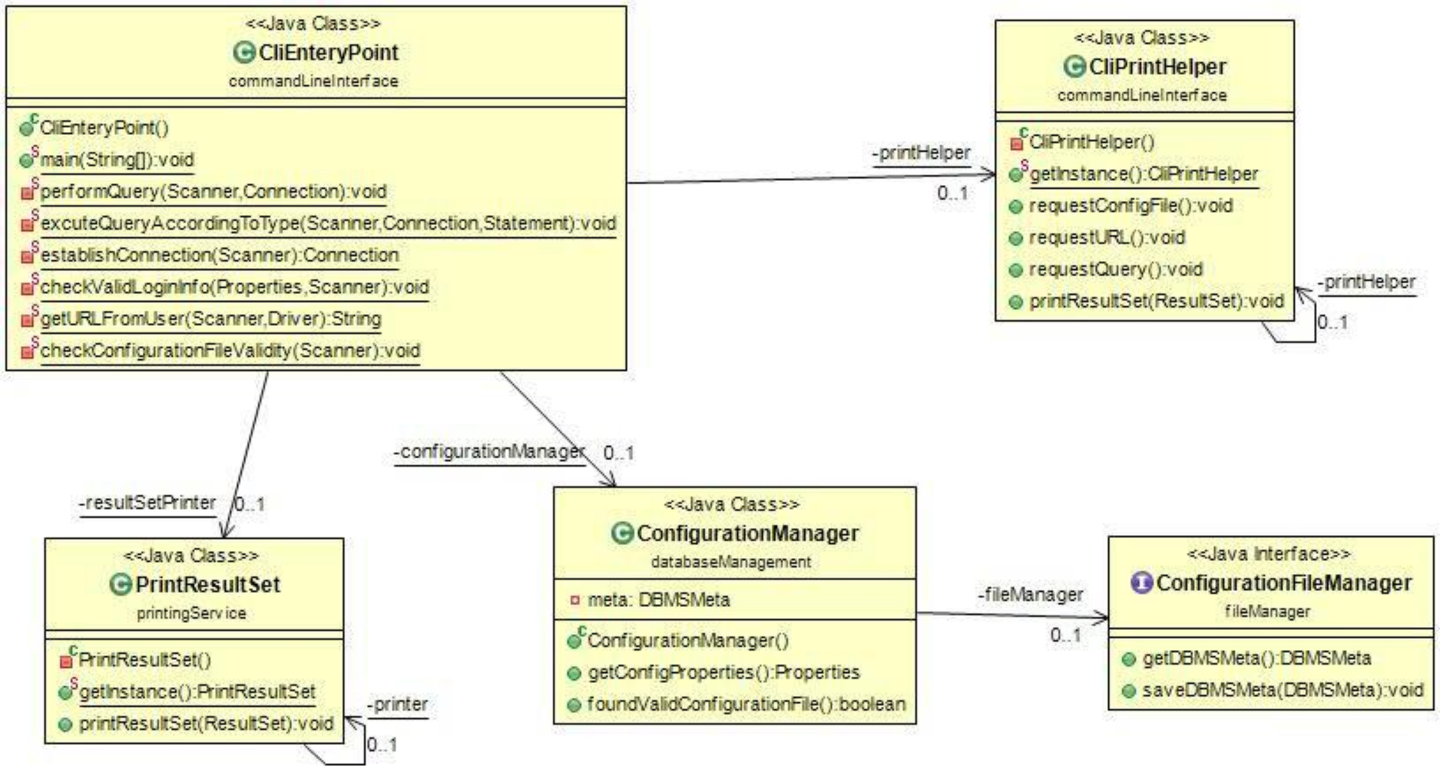
18- **logs:** contains the logging service code that uses the log4j.

19- **exceptions**: contains our own defined exceptions.

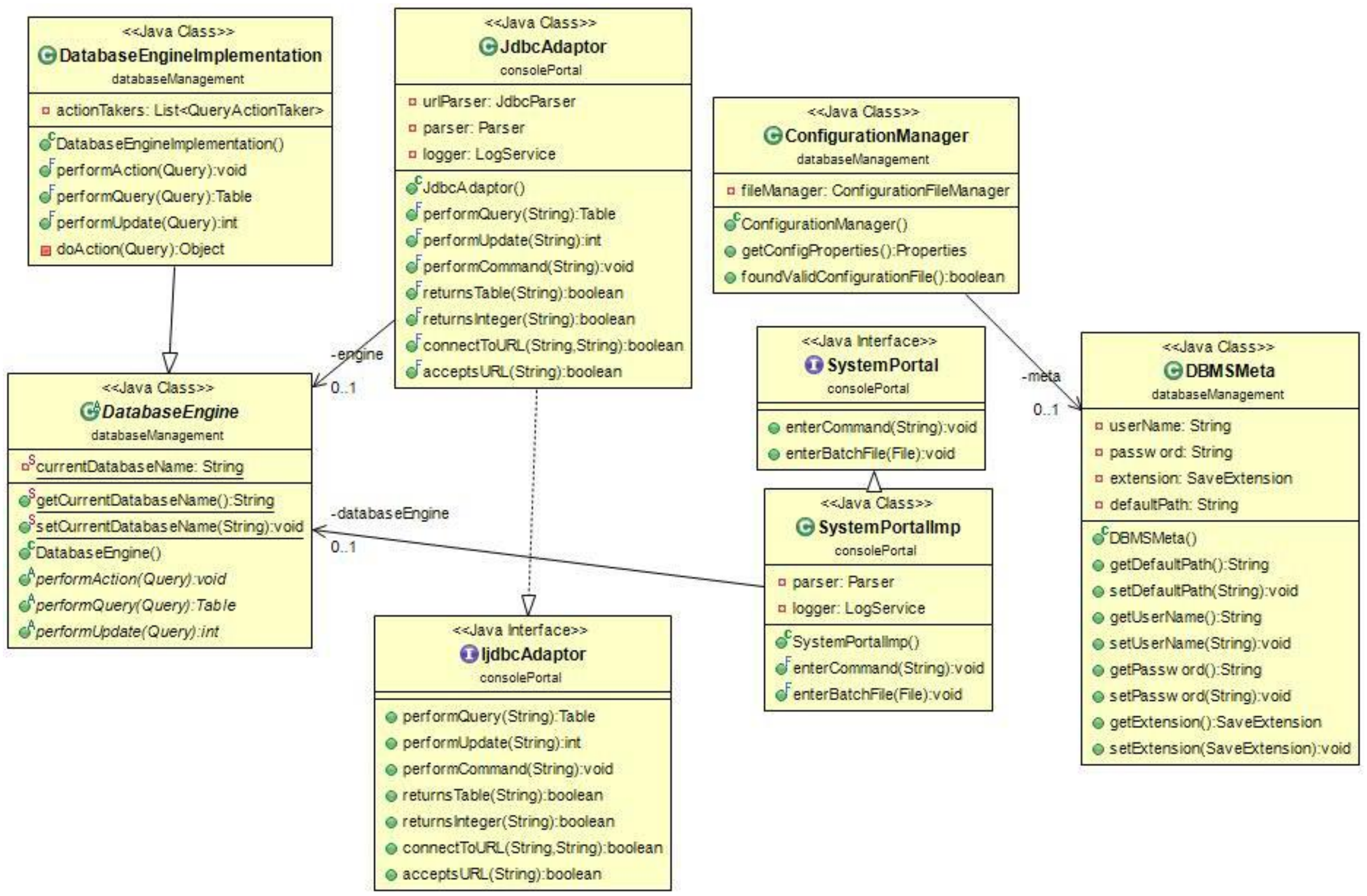
II. UML Diagram



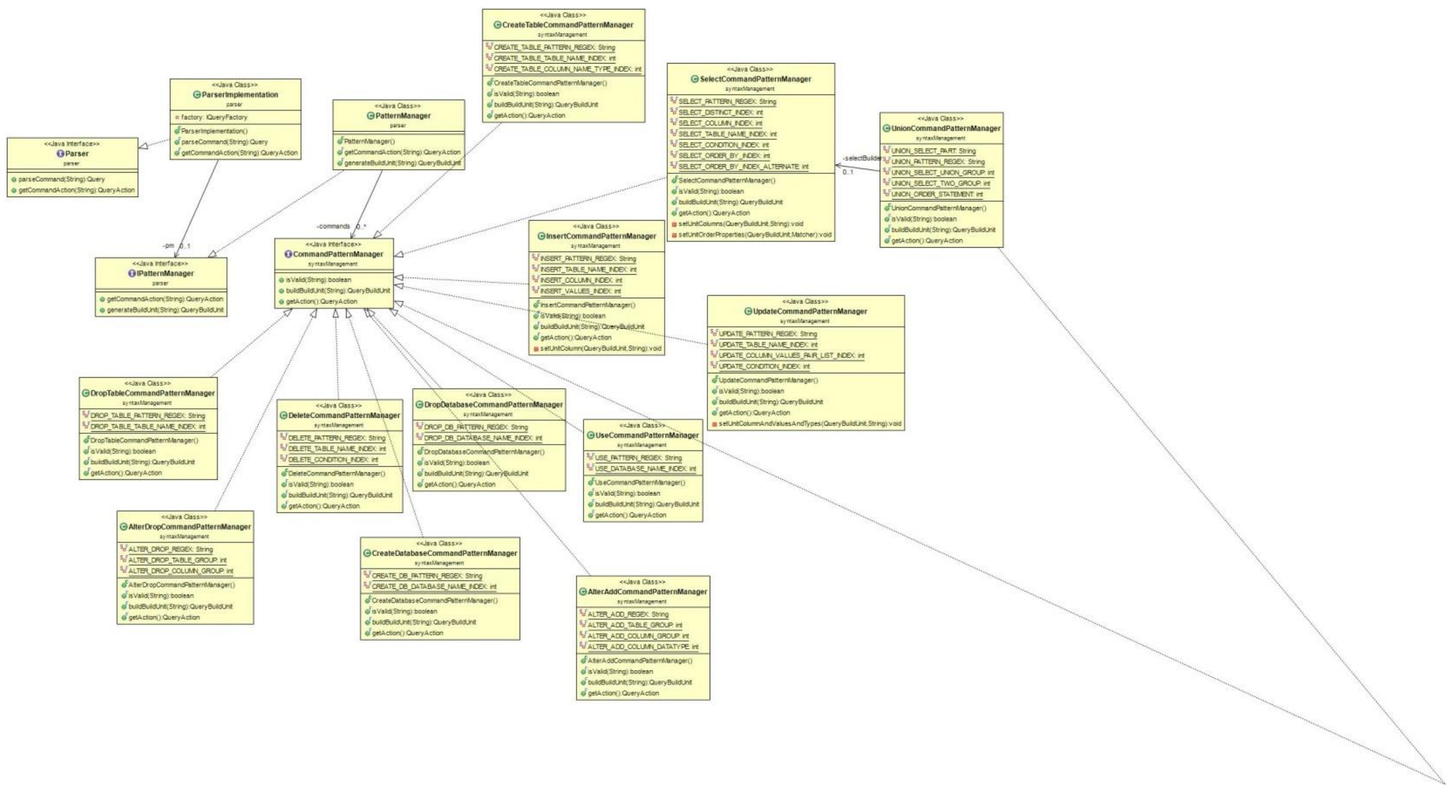
<<Java Class>> DeleteQueryActionTaker actionManagement
DeleteQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> SelectQueryActionTaker actionManagement
SelectQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
-selectActionTaker
0..1
<<Java Class>> CreateTableQueryActionTaker actionManagement
CreateTableQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> UpdateQueryActionTaker actionManagement
UpdateQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> QueryActionTaker actionManagement
fileManager: FileManager
getFileManager():FileManager
QueryActionTaker()
isDatabaseFound(String):boolean
isTableFound(String,String):boolean
getTable(String,String):Table
getTableFromQuery(TableQuery):Table
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> UnionQueryActionTaker actionManagement
UnionQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
unionTwoTables(Table,Table,Boolean):Table
<<Java Class>> DropDatabaseQueryActionTaker actionManagement
DropDatabaseQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> CreateDatabaseQueryActionTaker actionManagement
CreateDatabaseQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> AlterAddQueryActionTaker actionManagement
AlterAddQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> AlterDropQueryActionTaker actionManagement
AlterDropQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> DropTableQueryActionTaker actionManagement
DropTableQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> UseQueryActionTaker actionManagement
UseQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
<<Java Class>> InsertQueryActionTaker actionManagement
InsertQueryActionTaker()
accepts(Query):boolean
doQuery(Query):Object
insertIntoTable(InsertQuery,Table):Row
insertIntoSpecificColumnsIntoTable(InsertIntoSpecificColumnsQuery,Table):Row
fillSpecificDataInRow(InsertIntoSpecificColumnsQuery,Table):List<String>



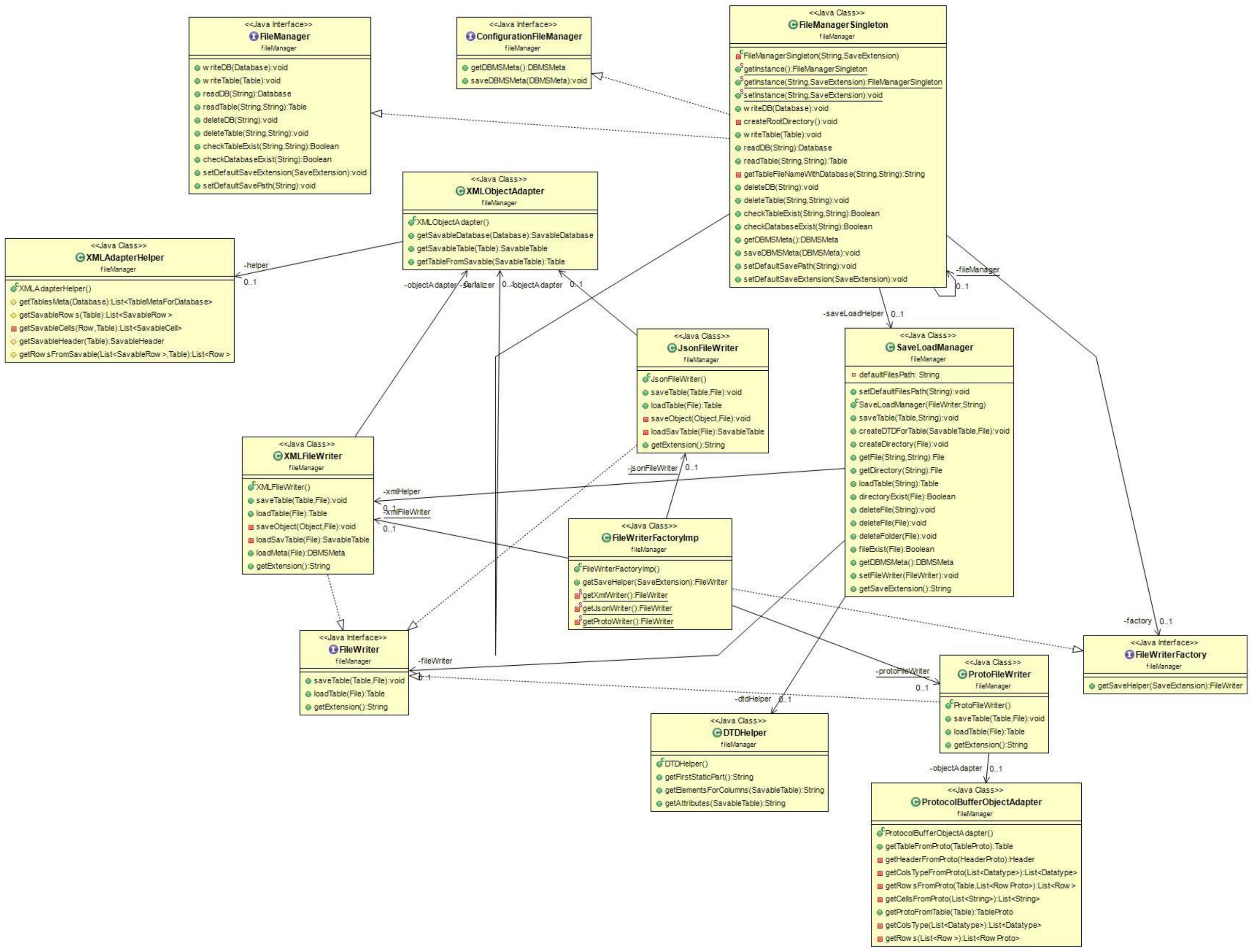
<<Java Class>> CliEnteryPoint commandLineInterface
CliEnteryPoint()
main(String[]):void
performQuery(Scanner,Connection):void
excuteQueryAccordingToType(Scanner,Connection,Statement):void
establishConnection(Scanner):Connection
checkValidLoginInfo(Properties,Scanner):void
getURLFromUser(Scanner,Driver):String
checkConfigurationFileValidity(Scanner):void
-printHelper
0..1
<<Java Class>> CliPrintHelper commandLineInterface
CliPrintHelper()
getInstance():CliPrintHelper
requestConfigFile():void
requestURL():void
requestQuery():void
printResultSet(ResultSet):void
-printHelper
0..1
-configurationManager
0..1
-resultSetPrinter
0..1
<<Java Class>> PrintResultSet printingService
PrintResultSet()
getInstance():PrintResultSet
printResultSet(ResultSet):void
-printer
0..1
<<Java Class>> ConfigurationManager databaseManagement
meta: DBMSMeta
ConfigurationManager()
getConfigProperties():Properties
foundValidConfigurationFile():boolean
-fileManager
0..1
<<Java Interface>> ConfigurationFileManager fileManager
getDBMSMeta():DBMSMeta
saveDBMSMeta(DBMSMeta):void



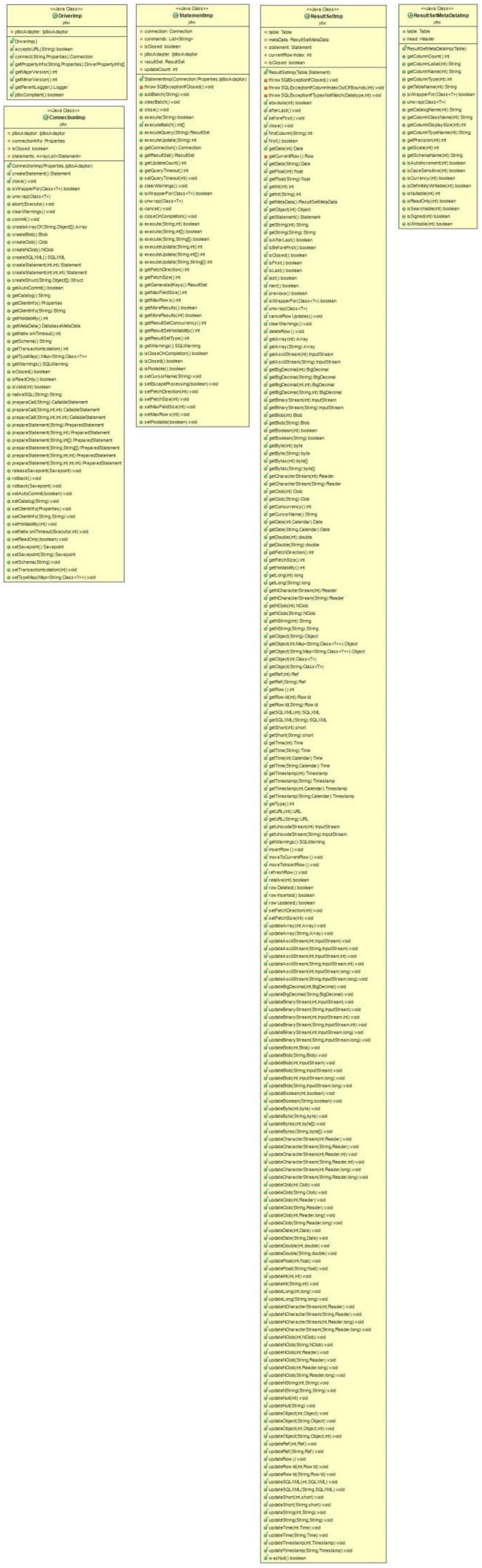
<<Java Class>>
DatabaseEngineImplementation
databaseManagement
actionTakers: List<QueryActionTaker>
DatabaseEngineImplementation()
performAction(Query):void
performQuery(Query):Table
performUpdate(Query):int
doAction(Query):Object
<<Java Class>>
JdbcAdaptor
consolePortal
urlParser: JdbcParser
parser: Parser
logger: LogService
JdbcAdaptor()
performQuery(String):Table
performUpdate(String):int
performCommand(String):void
returnsTable(String):boolean
returnsInteger(String):boolean
connectToURL(String,String):boolean
acceptsURL(String):boolean
<<Java Class>>
ConfigurationManager
databaseManagement
fileManager: ConfigurationFileManager
ConfigurationManager()
getConfigProperties():Properties
foundValidConfigurationFile():boolean
-engine
0..1
<<Java Class>>
DatabaseEngine
databaseManagement
currentDatabaseName: String
getCurrentDatabaseName():String
setCurrentDatabaseName(String):void
DatabaseEngine()
performAction(Query):void
performQuery(Query):Table
performUpdate(Query):int
<<Java Interface>>
SystemPortal
consolePortal
enterCommand(String):void
enterBatchFile(File):void
-meta
0..1
<<Java Class>>
DBMSMeta
databaseManagement
userName: String
password: String
extension: SaveExtension
defaultPath: String
DBMSMeta()
getDefaultPath():String
setDefaultPath(String):void
getUserName():String
setUserName(String):void
getPassword():String
setPassword(String):void
getExtension():SaveExtension
setExtension(SaveExtension):void
-databaseEngine
0..1
<<Java Class>>
SystemPortalImp
consolePortal
parser: Parser
logger: LogService
SystemPortalImp()
enterCommand(String):void
enterBatchFile(File):void
<<Java Interface>>
IjdbcAdaptor
consolePortal
performQuery(String):Table
performUpdate(String):int
performCommand(String):void
returnsTable(String):boolean
returnsInteger(String):boolean
connectToURL(String,String):boolean
acceptsURL(String):boolean



<<Java Class>>
CreateTableCommandPatternManager
syntaxManagement
CREATE_TABLE_PATTERN_REGEX: String
CREATE_TABLE_TABLE_NAME_INDEX: int
CREATE_TABLE_COLUMN_NAME_TYPE_INDEX: int
CreateTableCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
SelectCommandPatternManager
syntaxManagement
SELECT_PATTERN_REGEX: String
SELECT_DISTINCT_INDEX: int
SELECT_COLUMN_INDEX: int
SELECT_TABLE_NAME_INDEX: int
SELECT_CONDITION_INDEX: int
SELECT_ORDER_BY_INDEX: int
SELECT_ORDER_BY_INDEX_ALTERNATE: int
SelectCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
setUnitColumns(QueryBuildUnit,String):void
setUnitOrderProperties(QueryBuildUnit,Matcher):void
<<Java Class>>
ParserImplementation
parser
factory: IQueryFactory
ParserImplementation()
parseCommand(String):Query
getCommandAction(String):QueryAction
<<Java Class>>
PatternManager
parser
PatternManager()
getCommandAction(String):QueryAction
generateBuildUnit(String):QueryBuildUnit
-selectBuilder
0..1
<<Java Class>>
UnionCommandPatternManager
syntaxManagement
UNION_SELECT_PART: String
UNION_PATTERN_REGEX: String
UNION_SELECT_UNION_GROUP: int
UNION_SELECT_TWO_GROUP: int
UNION_ORDER_STATEMENT: int
UnionCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Interface>>
Parser
parser
parseCommand(String):Query
getCommandAction(String):QueryAction
-pm
0..1
-commands
0..*
<<Java Interface>>
IPatternManager
parser
getCommandAction(String):QueryAction
generateBuildUnit(String):QueryBuildUnit
<<Java Interface>>
CommandPatternManager
syntaxManagement
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
InsertCommandPatternManager
syntaxManagement
INSERT_PATTERN_REGEX: String
INSERT_TABLE_NAME_INDEX: int
INSERT_COLUMN_INDEX: int
INSERT_VALUES_INDEX: int
InsertCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
setUnitColumn(QueryBuildUnit,String):void
<<Java Class>>
UpdateCommandPatternManager
syntaxManagement
UPDATE_PATTERN_REGEX: String
UPDATE_TABLE_NAME_INDEX: int
UPDATE_COLUMN_VALUES_PAIR_LIST_INDEX: int
UPDATE_CONDITION_INDEX: int
UpdateCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
setUnitColumnAndValuesAndTypes(QueryBuildUnit,String):void
<<Java Class>>
DropTableCommandPatternManager
syntaxManagement
DROP_TABLE_PATTERN_REGEX: String
DROP_TABLE_TABLE_NAME_INDEX: int
DropTableCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
DeleteCommandPatternManager
syntaxManagement
DELETE_PATTERN_REGEX: String
DELETE_TABLE_NAME_INDEX: int
DELETE_CONDITION_INDEX: int
DeleteCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
DropDatabaseCommandPatternManager
syntaxManagement
DROP_DB_PATTERN_REGEX: String
DROP_DB_DATABASE_NAME_INDEX: int
DropDatabaseCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
UseCommandPatternManager
syntaxManagement
USE_PATTERN_REGEX: String
USE_DATABASE_NAME_INDEX: int
UseCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
AlterDropCommandPatternManager
syntaxManagement
ALTER_DROP_REGEX: String
ALTER_DROP_TABLE_GROUP: int
ALTER_DROP_COLUMN_GROUP: int
AlterDropCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
CreateDatabaseCommandPatternManager
syntaxManagement
CREATE_DB_PATTERN_REGEX: String
CREATE_DB_DATABASE_NAME_INDEX: int
CreateDatabaseCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction
<<Java Class>>
AlterAddCommandPatternManager
syntaxManagement
ALTER_ADD_REGEX: String
ALTER_ADD_TABLE_GROUP: int
ALTER_ADD_COLUMN_GROUP: int
ALTER_ADD_COLUMN_DATATYPE: int
AlterAddCommandPatternManager()
isValid(String):boolean
buildBuildUnit(String):QueryBuildUnit
getAction():QueryAction

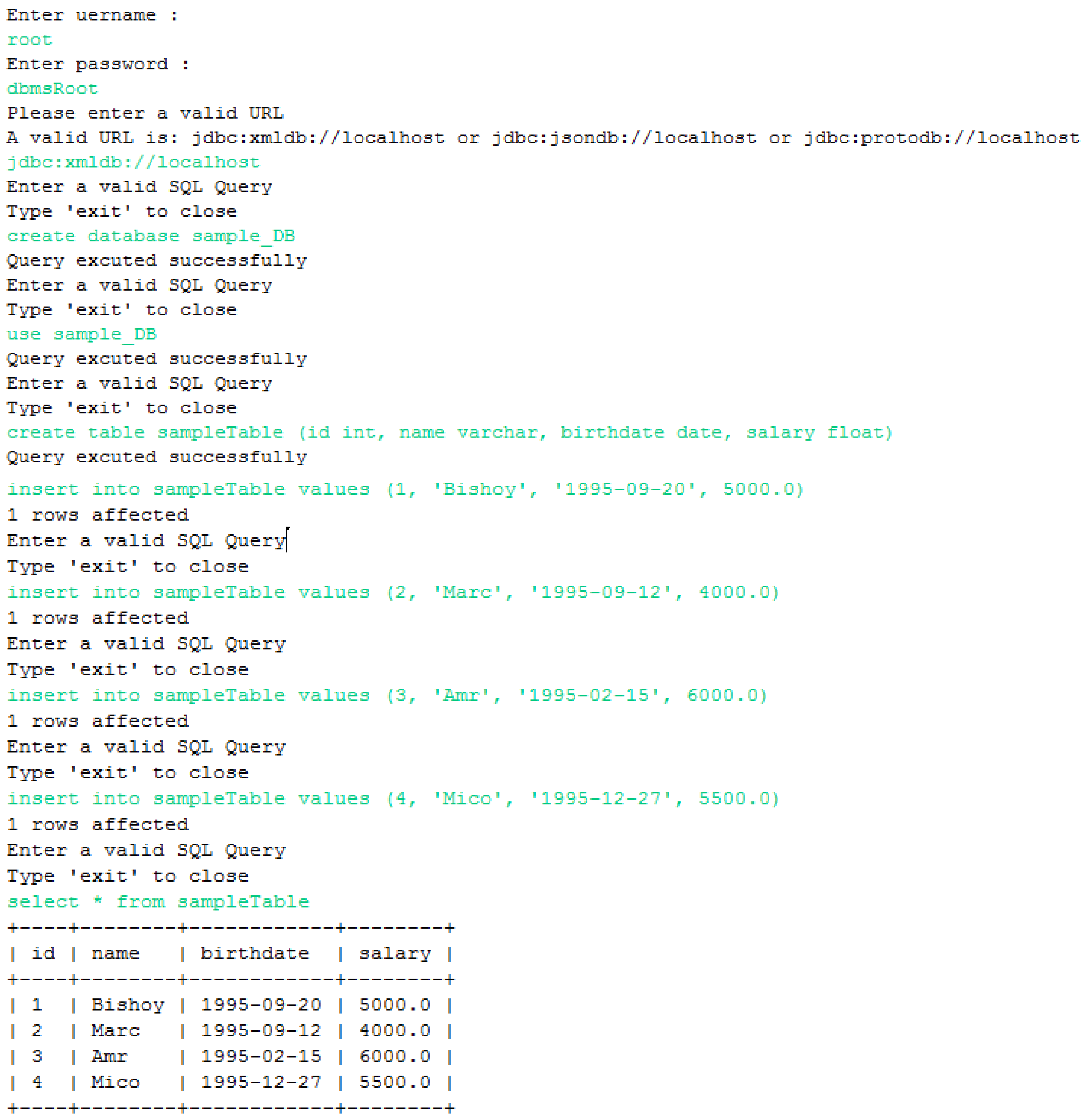


<<Java Interface>>
FileManager
fileManager
writeDB(Database):void
writeTable(Table):void
readDB(String):Database
readTable(String,String):Table
deleteDB(String):void
deleteTable(String,String):void
checkTableExist(String,String):Boolean
checkDatabaseExist(String):Boolean
setDefaultSaveExtension(SaveExtension):void
setDefaultSavePath(String):void
<<Java Interface>>
ConfigurationFileManager
fileManager
getDBMSMeta():DBMSMeta
saveDBMSMeta(DBMSMeta):void
<<Java Class>>
FileManagerSingleton
fileManager
FileManagerSingleton(String,SaveExtension)
getInstance():FileManagerSingleton
getInstance(String,SaveExtension):FileManagerSingleton
setInstance(String,SaveExtension):void
writeDB(Database):void
createRootDirectory():void
writeTable(Table):void
readDB(String):Database
readTable(String,String):Table
getTableFileNameWithDatabase(String,String):String
deleteDB(String):void
deleteTable(String,String):void
checkTableExist(String,String):Boolean
checkDatabaseExist(String):Boolean
getDBMSMeta():DBMSMeta
saveDBMSMeta(DBMSMeta):void
setDefaultSavePath(String):void
setDefaultSaveExtension(SaveExtension):void
-fileManager
0..1
<<Java Class>>
XMLObjectAdapter
fileManager
XMLObjectAdapter()
getSavableDatabase(Database):SavableDatabase
getSavableTable(Table):SavableTable
getTableFromSavable(SavableTable):Table
<<Java Class>>
XMLAdapterHelper
fileManager
XMLAdapterHelper()
getTablesMeta(Database):List<TableMetaForDatabase>
getSavableRows(Table):List<SavableRow>
getSavableCells(Row,Table):List<SavableCell>
getSavableHeader(Table):SavableHeader
getRowsFromSavable(List<SavableRow>,Table):List<Row>
-helper
0..1
-objectAdapter
-serializer
0..1
-objectAdapter
0..1
-saveLoadHelper
0..1
<<Java Class>>
JsonFileWriter
fileManager
JsonFileWriter()
saveTable(Table,File):void
loadTable(File):Table
saveObject(Object,File):void
loadSavTable(File):SavableTable
getExtension():String
<<Java Class>>
SaveLoadManager
fileManager
defaultFilesPath: String
setDefaultFilesPath(String):void
SaveLoadManager(FileWriter,String)
saveTable(Table,String):void
createDTDForTable(SavableTable,File):void
createDirectory(File):void
getFile(String,String):File
getDirectory(String):File
loadTable(String):Table
directoryExist(File):Boolean
deleteFile(String):void
deleteFile(File):void
deleteFolder(File):void
fileExist(File):Boolean
getDBMSMeta():DBMSMeta
setFileWriter(FileWriter):void
getSaveExtension():String
<<Java Class>>
XMLFileWriter
fileManager
XMLFileWriter()
saveTable(Table,File):void
loadTable(File):Table
saveObject(Object,File):void
loadSavTable(File):SavableTable
loadMeta(File):DBMSMeta
getExtension():String
-jsonFileWriter
0..1
-xmlHelper
0..1
-xmlFileWriter
0..1
<<Java Class>>
FileWriterFactoryImp
fileManager
FileWriterFactoryImp()
getSaveHelper(SaveExtension):FileWriter
getXmlWriter():FileWriter
getJsonWriter():FileWriter
getProtoWriter():FileWriter
-factory
0..1
<<Java Interface>>
FileWriterFactory
fileManager
getSaveHelper(SaveExtension):FileWriter
<<Java Interface>>
FileWriter
fileManager
saveTable(Table,File):void
loadTable(File):Table
getExtension():String
-fileWriter
0..1
-protoFileWriter
0..1
-dtdHelper
0..1
<<Java Class>>
ProtoFileWriter
fileManager
ProtoFileWriter()
saveTable(Table,File):void
loadTable(File):Table
getExtension():String
<<Java Class>>
DTDHelper
fileManager
DTDHelper()
getFirstStaticPart():String
getElementsForColumns(SavableTable):String
getAttributes(SavableTable):String
-objectAdapter
0..1
<<Java Class>>
ProtocolBufferObjectAdapter
fileManager
ProtocolBufferObjectAdapter()
getTableFromProto(TableProto):Table
getHeaderFromProto(HeaderProto):Header
getColsTypeFromProto(List<Datatype>):List<Datatype>
getRowsFromProto(Table,List<RowProto>):List<Row>
getCellsFromProto(List<String>):List<String>
getProtoFromTable(Table):TableProto
getColsType(List<Datatype>):List<Datatype>
getRows(List<Row>):List<RowProto>


<<Java Class>>
DriverImp
jdbc
jdbcAdapter: IjdbcAdaptor
DriverImp()
acceptsURL(String):boolean
connect(String,Properties):Connection
getPropertyInfo(String,Properties):DriverPropertyInfo[]
getMajorVersion():int
getMinorVersion():int
getParentLogger():Logger
jdbcCompliant():boolean
<<Java Class>>
ConnectionImp
jdbc
jdbcAdaptor: IjdbcAdaptor
connectionInfo: Properties
isClosed: boolean
statements: ArrayList<Statement>
ConnectionImp(Properties,IjdbcAdaptor)
createStatement():Statement
close():void
isWrapperFor(Class<?>):boolean
unwrap(Class<T>)
abort(Executor):void
clearWarnings():void
commit():void
createArrayOf(String,Object[]):Array
createBlob():Blob
createClob():Clob
createNClob():NClob
createSQLXML():SQLXML
createStatement(int,int):Statement
createStatement(int,int,int):Statement
createStruct(String,Object[]):Struct
getAutoCommit():boolean
getCatalog():String
getClientInfo():Properties
getClientInfo(String):String
getHoldability():int
getMetaData():DatabaseMetaData
getNetworkTimeout():int
getSchema():String
getTransactionIsolation():int
getTypeMap():Map<String,Class<?>>
getWarnings():SQLWarning
isClosed():boolean
isReadOnly():boolean
isValid(int):boolean
nativeSQL(String):String
prepareCall(String):CallableStatement
prepareCall(String,int,int):CallableStatement
prepareCall(String,int,int,int):CallableStatement
prepareStatement(String):PreparedStatement
prepareStatement(String,int):PreparedStatement
prepareStatement(String,int[]):PreparedStatement
prepareStatement(String,String[]):PreparedStatement
prepareStatement(String,int,int):PreparedStatement
prepareStatement(String,int,int,int):PreparedStatement
releaseSavepoint(Savepoint):void
rollback():void
rollback(Savepoint):void
setAutoCommit(boolean):void
setCatalog(String):void
setClientInfo(Properties):void
setClientInfo(String,String):void
setHoldability(int):void
setNetworkTimeout(Executor,int):void
setReadOnly(boolean):void
setSavepoint():Savepoint
setSavepoint(String):Savepoint
setSchema(String):void
setTransactionIsolation(int):void
setTypeMap(Map<String,Class<?>>):void
<<Java Class>>
StatementImp
jdbc
connection: Connection
commands: List<String>
isClosed: boolean
jdbcAdapter: IjdbcAdaptor
resultSet: ResultSet
updateCount: int
StatementImp(Connection,Properties,IjdbcAdaptor)
throw SQlExceptionIfClosed():void
addBatch(String):void
clearBatch():void
close():void
execute(String):boolean
executeBatch():int[]
executeQuery(String):ResultSet
executeUpdate(String):int
getConnection():Connection
getResultSet():ResultSet
getUpdateCount():int
getQueryTimeout():int
setQueryTimeout(int):void
clearWarnings():void
isWrapperFor(Class<?>):boolean
unwrap(Class<T>)
cancel():void
closeOnCompletion():void
execute(String,int):boolean
execute(String,int[]):boolean
execute(String,String[]):boolean
executeUpdate(String,int):int
executeUpdate(String,int[]):int
executeUpdate(String,String[]):int
getFetchDirection():int
getFetchSize():int
getGeneratedKeys():ResultSet
getMaxFieldSize():int
getMaxRows():int
getMoreResults():boolean
getMoreResults(int):boolean
getResultSetConcurrency():int
getResultSetHoldability():int
getResultSetType():int
getWarnings():SQLWarning
isCloseOnCompletion():boolean
isClosed():boolean
isPoolable():boolean
setCursorName(String):void
setEscapeProcessing(boolean):void
setFetchDirection(int):void
setFetchSize(int):void
setMaxFieldSize(int):void
setMaxRows(int):void
setPoolable(boolean):void
<<Java Class>>
ResultSetImp
jdbc
table: Table
metaData: ResultSetMetaData
statement: Statement
currentRowIndex: int
isClosed: boolean
ResultSetImp(Table,Statement)
throw SQlExceptionIfClosed():void
throw SQLExceptionIfColumnIndexOutOfBounds(int):void
throw SQLExceptionIfTypesNotMatch(Datatype,int):void
absolute(int):boolean
afterLast():void
beforeFirst():void
close():void
findColumn(String):int
first():boolean
getDate(int):Date
getCurrentRow():Row
getDate(String):Date
getFloat(int):float
getFloat(String):float
getInt(int):int
getInt(String):int
getMetaData():ResultSetMetaData
getObject(int):Object
getStatement():Statement
getString(int):String
getString(String):String
isAfterLast():boolean
isBeforeFirst():boolean
isClosed():boolean
isFirst():boolean
isLast():boolean
last():boolean
next():boolean
previous():boolean
isWrapperFor(Class<?>):boolean
unwrap(Class<T>)
cancelRowUpdates():void
clearWarnings():void
deleteRow():void
getArray(int):Array
getArray(String):Array
getAsciiStream(int):InputStream
getAsciiStream(String):InputStream
getBigDecimal(int):BigDecimal
getBigDecimal(String):BigDecimal
getBigDecimal(int,int):BigDecimal
getBigDecimal(String,int):BigDecimal
getBinaryStream(int):InputStream
getBinaryStream(String):InputStream
getBlob(int):Blob
getBlob(String):Blob
getBoolean(int):boolean
getBoolean(String):boolean
getByte(int):byte
getByte(String):byte
getBytes(int):byte[]
getBytes(String):byte[]
getCharacterStream(int):Reader
getCharacterStream(String):Reader
getClob(int):Clob
getClob(String):Clob
getConcurrency():int
getCursorName():String
getDate(int,Calendar):Date
getDate(String,Calendar):Date
getDouble(int):double
getDouble(String):double
getFetchDirection():int
getFetchSize():int
getHoldability():int
getLong(int):long
getLong(String):long
getNCharacterStream(int):Reader
getNCharacterStream(String):Reader
getNClob(int):NClob
getNClob(String):NClob
getNString(int):String
getNString(String):String
getObject(String):Object
getObject(int,Map<String,Class<?>>):Object
getObject(String,Map<String,Class<?>>):Object
getObject(int,Class<T>)
getObject(String,Class<T>)
getRef(int):Ref
getRef(String):Ref
getRow():int
getRowId(int):RowId
getRowId(String):RowId
getSQLXML(int):SQLXML
getSQLXML(String):SQLXML
getShort(int):short
getShort(String):short
getTime(int):Time
getTime(String):Time
getTime(int,Calendar):Time
getTime(String,Calendar):Time
getTimestamp(int):Timestamp
getTimestamp(String):Timestamp
getTimestamp(int,Calendar):Timestamp
getTimestamp(String,Calendar):Timestamp
getType():int
getURL(int):URL
getURL(String):URL
getUnicodeStream(int):InputStream
getUnicodeStream(String):InputStream
getWarnings():SQLWarning
insertRow():void
moveToCurrentRow():void
moveToInsertRow():void
refreshRow():void
relative(int):boolean
rowDeleted():boolean
rowInserted():boolean
rowUpdated():boolean
setFetchDirection(int):void
setFetchSize(int):void
updateArray(int,Array):void
updateArray(String,Array):void
updateAsciiStream(int,InputStream):void
updateAsciiStream(String,InputStream):void
updateAsciiStream(int,InputStream,int):void
updateAsciiStream(String,InputStream,int):void
updateAsciiStream(int,InputStream,long):void
updateAsciiStream(String,InputStream,long):void
updateBigDecimal(int,BigDecimal):void
updateBigDecimal(String,BigDecimal):void
updateBinaryStream(int,InputStream):void
updateBinaryStream(String,InputStream):void
updateBinaryStream(int,InputStream,int):void
updateBinaryStream(String,InputStream,int):void
updateBinaryStream(int,InputStream,long):void
updateBinaryStream(String,InputStream,long):void
updateBlob(int,Blob):void
updateBlob(String,Blob):void
updateBlob(int,InputStream):void
updateBlob(String,InputStream):void
updateBlob(int,InputStream,long):void
updateBlob(String,InputStream,long):void
updateBoolean(int,boolean):void
updateBoolean(String,boolean):void
updateByte(int,byte):void
updateByte(String,byte):void
updateBytes(int,byte[]):void
updateBytes(String,byte[]):void
updateCharacterStream(int,Reader):void
updateCharacterStream(String,Reader):void
updateCharacterStream(int,Reader,int):void
updateCharacterStream(String,Reader,int):void
updateCharacterStream(int,Reader,long):void
updateCharacterStream(String,Reader,long):void
updateClob(int,Clob):void
updateClob(String,Clob):void
updateClob(int,Reader):void
updateClob(String,Reader):void
updateClob(int,Reader,long):void
updateClob(String,Reader,long):void
updateDate(int,Date):void
updateDate(String,Date):void
updateDouble(int,double):void
updateDouble(String,double):void
updateFloat(int,float):void
updateFloat(String,float):void
updateInt(int,int):void
updateInt(String,int):void
updateLong(int,long):void
updateLong(String,long):void
updateNCharacterStream(int,Reader):void
updateNCharacterStream(String,Reader):void
updateNCharacterStream(int,Reader,long):void
updateNCharacterStream(String,Reader,long):void
updateNClob(int,NClob):void
updateNClob(String,NClob):void
updateNClob(int,Reader):void
updateNClob(String,Reader):void
updateNClob(int,Reader,long):void
updateNClob(String,Reader,long):void
updateNString(int,String):void
updateNString(String,String):void
updateNull(int):void
updateNull(String):void
updateObject(int,Object):void
updateObject(String,Object):void
updateObject(int,Object,int):void
updateObject(String,Object,int):void
updateRef(int,Ref):void
updateRef(String,Ref):void
updateRow():void
updateRowId(int,RowId):void
updateRowId(String,RowId):void
updateSQLXML(int,SQLXML):void
updateSQLXML(String,SQLXML):void
updateShort(int,short):void
updateShort(String,short):void
updateString(int,String):void
updateString(String,String):void
updateTime(int,Time):void
updateTime(String,Time):void
updateTimestamp(int,Timestamp):void
updateTimestamp(String,Timestamp):void
wasNull():boolean
<<Java Class>>
ResultSetMetaDataImp
jdbc
table: Table
head: Header
ResultSetMetaDataImp(Table)
getColumnCount():int
getColumnLabel(int):String
getColumnName(int):String
getColumnType(int):int
getTableName(int):String
isWrapperFor(Class<?>):boolean
unwrap(Class<T>)
getCatalogName(int):String
getColumnClassName(int):String
getColumnDisplaySize(int):int
getColumnTypeName(int):String
getPrecision(int):int
getScale(int):int
getSchemaName(int):String
isAutoIncrement(int):boolean
isCaseSensitive(int):boolean
isCurrency(int):boolean
isDefinitelyWritable(int):boolean
isNullable(int):int
isReadOnly(int):boolean
isSearchable(int):boolean
isSigned(int):boolean
isWritable(int):boolean

III. Sample Run using Command Line Interface

- The user should login using the same username and password that are found in the configuration file.
- Notice that the path of the databases is a property inside the configuration file.
- The user is then asked to enter a valid URL which determines whether the program uses XML, JSON or Protocol Buffers for saving data.
- Finally, the user can now write SQL queries to the terminal and get the suitable response.



```
Enter uername :
root
Enter password :
dbmsRoot
Please enter a valid URL
A valid URL is: jdbc:xmldb://localhost or jdbc:jsondb://localhost or jdbc:protodb://localhost
jdbc:xmldb://localhost
Enter a valid SQL Query
Type 'exit' to close
create database sample_DB
Query excuted successfully
Enter a valid SQL Query
Type 'exit' to close
use sample_DB
Query excuted successfully
Enter a valid SQL Query
Type 'exit' to close
create table sampleTable (id int, name varchar, birthdate date, salary float)
Query excuted successfully
insert into sampleTable values (1, 'Bishoy', '1995-09-20', 5000.0)
1 rows affected
Enter a valid SQL Query
Type 'exit' to close
insert into sampleTable values (2, 'Marc', '1995-09-12', 4000.0)
1 rows affected
Enter a valid SQL Query
Type 'exit' to close
insert into sampleTable values (3, 'Amr', '1995-02-15', 6000.0)
1 rows affected
Enter a valid SQL Query
Type 'exit' to close
insert into sampleTable values (4, 'Mico', '1995-12-27', 5500.0)
1 rows affected
Enter a valid SQL Query
Type 'exit' to close
select * from sampleTable
+----+--------+------------+--------+
| id | name   | birthdate  | salary |
+----+--------+------------+--------+
| 1  | Bishoy | 1995-09-20 | 5000.0 |
| 2  | Marc   | 1995-09-12 | 4000.0 |
| 3  | Amr    | 1995-02-15 | 6000.0 |
| 4  | Mico   | 1995-12-27 | 5500.0 |
+----+--------+------------+--------+
```