<noscript> <div>JavaScript is disabled on your browser.</div> </noscript> <h2>Frame Alert</h2> <p>This document is designed to be viewed using the frames feature. If you see this message, you are using a non-frame-capable web client. Link to Non-frame version.</p>

<div>JavaScript is disabled on your browser.</div>

Class MySql

java.lang.Object
    MySql

```
public class MySql
extends java.lang.Object
```

Created by RonOS on 16-12-2016.

Field Summary

Fields

Modifier and Type	Field and Description
(package private) java.sql.Connection	**connection**

Constructor Summary

Constructors

Constructor and Description
MySql(java.lang.String address, int port, java.lang.String database, java.lang.String username, java.lang.String password)
The mysql object is responsible for facilitating a connection to the Mysql database and allows for some generic methods such as select queries and insertions.

Method Summary

| All Methods | Instance Methods | Concrete Methods |

Modifier and Type	Method and Description
void	**delete**(java.lang.String table, java.lang.String condition, java.lang.Boolean deleteAll) This method allows the deletion of items from a table
java.sql.ResultSet	**getSelectFromQuery**(java.lang.String query) This method allows the programmer to select entries from a database
void	**insertQuery**(java.lang.String query) This will execute a given insert query

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll,
toString, wait, wait, wait
```

Field Detail

connection

```
java.sql.Connection connection
```

Constructor Detail

MySql

```
public MySql(java.lang.String address,
             int port,
             java.lang.String database,
```

```
                java.lang.String username,
                java.lang.String password)
```

The mysql object is responsible for facilitating a connection to the Mysql database and allows for some generic methods such as select queries and insertions.

Parameters:

```
address - The adress of the database engine

port - The port of the database engine

database - The database to connect to

username - Database username

password - Database password
```

Method Detail

getSelectFromQuery

```
public java.sql.ResultSet getSelectFromQuery(java.lang.String query)
```

This method allows the programmer to select entries from a database

Parameters:

```
query - The select query to be executed.
```

Returns:

```
A resultset is returned containing the result(s) of the query
```

delete

```
public void delete(java.lang.String table,
                   java.lang.String condition,
                   java.lang.Boolean deleteAll)
```

This method allows the deletion of items from a table

Parameters:

```
table - The table from which items need to be deleted
```

condition - A condition allow the filtering of items such as a where clause

deleteAll - The option to delete everything froma a table

insertQuery

```
public void insertQuery(java.lang.String query)
```

This will execute a given insert query

Parameters:

query - The insert to be made

<div>JavaScript is disabled on your browser.</div>

PACKAGE

CLASS

PREV CLASS   FRAMES   ALL CLASSES

NEXT CLASS   NO FRAMES

SUMMARY:
NESTED |
FIELD |
CONSTR |
METHOD
DETAIL:
FIELD |
CONSTR |
METHOD

Class Receiver

java.lang.Object
    java.lang.Thread
        Receiver

All Implemented Interfaces:

```
java.lang.Runnable
```

```
public class Receiver
extends java.lang.Thread
```

Created by RonOS on 1/29/2017.

Nested Class Summary

Nested classes/interfaces inherited from class java.lang.Thread
java.lang.Thread.State, java.lang.Thread.UncaughtExceptionHandler

---

Field Summary

Fields	
Modifier and Type	**Field and Description**
private java.util.Vector<**clientThread**>	**clients**
private int	**counter**

private int	**port**	
private java.net.ServerSocket	**serverSocket**	

Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

Constructor Summary

Constructors

Constructor and Description

Receiver(int port, java.lang.String databaseIp, int databasePort, java.lang.String databaseUsername, java.lang.String databasePassword, java.lang.String databaseName)

The receiver object is responsible for handling all the incoming connections and logging the existing connections.

Method Summary

All Methods | **Instance Methods** | **Concrete Methods**

Modifier and Type	Method and Description
void	**listen**() This starts listening on the serversocket in order to handle incoming connections
void	**run**() This starts the thread and starts the listering proces

Methods inherited from class java.lang.Thread

activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup,

getUncaughtExceptionHandler, holdsLock, interrupt, interrupted,
isAlive, isDaemon, isInterrupted, join, join, join, resume,
setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler,
setName, setPriority, setUncaughtExceptionHandler, sleep, sleep,
start, stop, stop, suspend, toString, yield

Methods inherited from class java.lang.Object

equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait,
wait

Field Detail

serverSocket

private java.net.ServerSocket serverSocket

counter

private int counter

port

private int port

clients

private java.util.Vector<clientThread> clients

Constructor Detail

Receiver

```
public Receiver(int port,
                java.lang.String databaseIp,
                int databasePort,
                java.lang.String databaseUsername,
                java.lang.String databasePassword,
                java.lang.String databaseName)
```

The receiver object is responsible for handling all the incoming connections and logging the existing connections.

Parameters:

```
port - The portnumber to listen to

databaseIp - The IP-adres of the database to write to

databasePort - The portnumber of the database to write to

databaseUsername - The username for the databse

databasePassword - The password for the database

databaseName - The name of database to write to
```

Method Detail

run

```
public void run()
```

This starts the thread and starts the listering proces

Specified by:

```
run in interface java.lang.Runnable
```

Overrides:

```
run in class java.lang.Thread
```

listen

```
public void listen()
```

This starts listening on the serversocket in order to handle incoming connections

<div>JavaScript is disabled on your browser.</div>

Class Server

java.lang.Object
    Server

```
public class Server
extends java.lang.Object
```

Constructor Summary

Constructors

Constructor and Description
Server()

Method Summary

All Methods	Static Methods	Concrete Methods

Modifier and Type	Method and Description
static void	**main**(java.lang.String[] args) The main method for the program

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll,

```
toString, wait, wait, wait
```

Constructor Detail

Server

```
public Server()
```

Method Detail

main

```
public static void main(java.lang.String[] args)
                throws java.io.IOException
```

The main method for the program

Parameters:

```
args -
```

Throws:

```
java.io.IOException
```

<div>JavaScript is disabled on your browser.</div>

Class WeatherDatabaseHelper

java.lang.Object
    WeatherDatabaseHelper

---

public class **WeatherDatabaseHelper**
extends java.lang.Object

Created by RonOS on 16-12-2016.

Field Summary

Fields	
Modifier and Type	**Field and Description**
(package private) **MySql mySql**	

Constructor Summary

Constructors
Constructor and Description
WeatherDatabaseHelper(**MySql** mysql)   This class makes handling project specific database operations easier for the programmer

Method Summary

| All Methods | Instance Methods | Concrete Methods |

Modifier and Type	Method and Description
float	**getAverageCloudCoverage**(int stationId, java.lang.String tablename) This method returns the average cloudcoverage for a given station
float	**getAverageVisibilty**(int stationId, java.lang.String tablename) This method returns the average visibility for a given station
void	**insertArgentinaData**(int stationId, java.lang.Float cloudcoverage, java.lang.Float visibility, java.lang.String date) This query allows the insertion of Argentina's weatherdata
void	**insertAverageArgentinaData**() This method calculates the average value of the last 24 hours of measurements and inserts it into the average database for Argentina
void	**insertAverageOceaniaData**() This method calculates the average value of the last 24 hours of measurements and inserts it into the average database for Oceania
void	**insertOceaniaData**(int stationId, java.lang.Float cloudcoverage, java.lang.String date) This query allows the insertion of Oceania's weatherdata

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

mySql

```
MySql mySql
```

Constructor Detail

WeatherDatabaseHelper

```
public WeatherDatabaseHelper(MySql mysql)
```

This class makes handling project specific database operations easier for the programmer

Parameters:

mysql -

Method Detail

insertOceaniaData

```
public void insertOceaniaData(int stationId,
                              java.lang.Float cloudcoverage,
                              java.lang.String date)
```

This query allows the insertion of Oceania's weatherdata

Parameters:

stationId - The station id from which the measurement originated

cloudcoverage - The measured cloudcoverage

date - The date of the measurement

insertArgentinaData

```
public void insertArgentinaData(int stationId,
                                java.lang.Float cloudcoverage,
                                java.lang.Float visibility,
                                java.lang.String date)
```

This query allows the insertion of Argentina's weatherdata

Parameters:

```
stationId - The station id from which the measurement originated
```

```
cloudcoverage - The measured cloudcoverage
```

```
visibility - The measured visibility
```

```
date - The date of the measurement
```

insertAverageOceaniaData

```
public void insertAverageOceaniaData()
```

This method calculates the average value of the last 24 hours of measurements and inserts it into the average database for Oceania

insertAverageArgentinaData

```
public void insertAverageArgentinaData()
```

This method calculates the average value of the last 24 hours of measurements and inserts it into the average database for Argentina

getAverageVisibilty

```
public float getAverageVisibilty(int stationId,
                                 java.lang.String tablename)
```

This method returns the average visibility for a given station

Parameters:

```
stationId - The station from which the average visibility must be
```

```
calculated
```

tablename - The tablename from where the data must originate

Returns:

```
Returns the average in floating point format
```

getAverageCloudCoverage

```
public float getAverageCloudCoverage(int stationId,
                                     java.lang.String tablename)
```

This method returns the average cloudcoverage for a given station

Parameters:

```
stationId - The station from which the average cloudcoverage must be
calculated
```

tablename - The tablename from where the data must originate

Returns:

```
Returns the average in floating point format
```

<div>JavaScript is disabled on your browser.</div>

PACKAGE

CLASS

PREV CLASS   FRAMES   ALL CLASSES

NEXT CLASS   NO FRAMES

SUMMARY:
NESTED |
FIELD |
CONSTR |
METHOD
DETAIL:
FIELD |
CONSTR |
METHOD

Class averageThread

java.lang.Object
    java.lang.Thread
        averageThread

All Implemented Interfaces:

java.lang.Runnable

---

public class **averageThread**
extends java.lang.Thread

Created by RonOS on 2/1/2017.

Nested Class Summary

Nested classes/interfaces inherited from class java.lang.Thread
java.lang.Thread.State, java.lang.Thread.UncaughtExceptionHandler

Field Summary

Fields	
Modifier and Type	**Field and Description**
private **WeatherDatabaseHelper**	**database**
private **MySql**	**sql**

Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

Constructor Summary

Constructors

Constructor and Description
averageThread() The average thread is responsible for updating the average values in the database

Method Summary

All Methods	Instance Methods	Concrete Methods

Modifier and Type	Method and Description
void	**run**() This starts the thread and makes it calculate an updated average of all the measurements of the last 24 hours, The update interval is set to 1 second by default.

Methods inherited from class java.lang.Thread

activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

Field Detail

sql

private MySql sql

database

private WeatherDatabaseHelper database

Constructor Detail

averageThread

public averageThread()

The average thread is responsible for updating the average values in the database

Method Detail

run

public void run()

This starts the thread and makes it calculate an updated average of all the measurements of the last 24 hours, The update interval is set to 1 second by default.

Specified by:

`run in interface java.lang.Runnable`

Overrides:

`run in class java.lang.Thread`

PACKAGE

PREV CLASS   FRAMES   ALL CLASSES

NEXT CLASS   NO FRAMES

SUMMARY:
NESTED |
FIELD |
CONSTR |
METHOD
DETAIL:
FIELD |
CONSTR |
METHOD

<div>JavaScript is disabled on your browser.</div>

Class clientThread

java.lang.Object
    java.lang.Thread
        clientThread

All Implemented Interfaces:

java.lang.Runnable

---

public class **clientThread**
extends java.lang.Thread

Created by RonOS on 1/29/2017.

Nested Class Summary

Nested classes/interfaces inherited from class java.lang.Thread

java.lang.Thread.State, java.lang.Thread.UncaughtExceptionHandler

Field Summary

Fields

Modifier and Type	Field and Description
(package private) java.io.BufferedReader	**br**
private int	**currentStation**

private **WeatherDatabaseHelper**	**database**
private boolean	**receiving**
java.net.Socket	**socket**
private **MySql**	**sql**
private boolean	**writing**

Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

Constructor Summary

Constructors

Constructor and Description
clientThread(java.net.Socket socket) The clientThread class is a class that represents a connected client it facilitates the parsing and writing of received data.

Method Summary

All Methods	Static Methods	Instance Methods	Concrete Methods

Modifier and Type	Method and Description
static org.w3c.dom.Document	**loadXMLFromString**(java.lang.String xml) This method is responsible for parsing each measurement to a workable document object
void	**receive**() When this method is called the thread will begin to receive and parse the XML data
void	**run**() This method runs the thread and starts the receiving process

void	**write**() This method enables the clientThread to write its received data to the database

Methods inherited from class java.lang.Thread

activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

Methods inherited from class java.lang.Object

equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

socket

public java.net.Socket socket

sql

private MySql sql

database

private WeatherDatabaseHelper database

receiving

```
private boolean receiving
```

writing

```
private boolean writing
```

currentStation

```
private int currentStation
```

br

```
java.io.BufferedReader br
```

Constructor Detail

clientThread

```
public clientThread(java.net.Socket socket)
```

The clientThread class is a class that represents a connected client it facilitates the parsing and writing of received data.

Parameters:

```
socket -
```

Method Detail

run

```
public void run()
```

This method runs the thread and starts the receiving process

Specified by:

```
run in interface java.lang.Runnable
```

Overrides:

```
run in class java.lang.Thread
```

receive

```
public void receive()
```

When this method is called the thread will begin to receive and parse the XML data

write

```
public void write()
```

This method enables the clientThread to write its received data to the database

loadXMLFromString

```
public static org.w3c.dom.Document loadXMLFromString(java.lang.String xml)
                                              throws java.lang.Exception
```

This method is responsible for parsing each measurement to a workable document object

Parameters:

```
xml - A string containing the XML to be parsed
```

Returns:

```
returns a document that allows for parsing of data
```

Throws:

```
java.lang.Exception - when invalid XML is provided it throws the
```

```
exception
```