Package
com.psycholight

com.psycholight
Class PSLightConnection

```
java.lang.Object
   |
   +-com.psycholight.PSLightConnection
```

All Implemented Interfaces:
com.jpbc.PConnection

---

public class **PSLightConnection**
extends java.lang.Object
implements com.jpbc.PConnection

Constructor Summary

public	PSLightConnection(java.lang.String url, java.lang.String user, java.lang.String pwd)

Method Summary

void	close()
com.jpbc.PBStatement	createPBStatement(java.lang.String command)
com.jpbc.PdlStatement	createPDLStatement(java.lang.String command)
com.jpbc.PmlStatement	createPMLStatement(java.lang.String command)
com.jpbc.PqlStatement	createPQLStatement(java.lang.String command)
java.sql.Connection	getPBConnection()

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PConnection

close, createPBStatement, createPDLStatement, createPMLStatement, createPQLStatement

---

Constructors

PSLightConnection

```
public PSLightConnection(java.lang.String url,
                         java.lang.String user,
                         java.lang.String pwd)
```

(continued on next page)

(continued from last page)

Methods

createPBStatement

```
public com.jpbc.PBStatement createPBStatement(java.lang.String command)
  throws com.jpbc.PException
```

createPDLStatement

```
public com.jpbc.PdlStatement createPDLStatement(java.lang.String command)
  throws com.jpbc.PException
```

createPMLStatement

```
public com.jpbc.PmlStatement createPMLStatement(java.lang.String command)
  throws com.jpbc.PException
```

createPQLStatement

```
public com.jpbc.PqlStatement createPQLStatement(java.lang.String command)
  throws com.jpbc.PException
```

close

```
public void close()
  throws com.jpbc.PException
```

getPBConnection

```
public java.sql.Connection getPBConnection()
```

com.psycholight
Class PSLightDriver

```
java.lang.Object
   |
   +-com.psycholight.PSLightDriver
```

All Implemented Interfaces:
com.jpbc.PDriver

---

public class **PSLightDriver**
extends java.lang.Object
implements com.jpbc.PDriver

Constructor Summary

public	PSLightDriver()

Method Summary

com.jpbc.PConnection	connect(java.lang.String url, java.lang.String user, java.lang.String pwd)
int	PBMSCompliant()

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `com.jpbc.PDriver`

`connect, PBMSCompliant`

Constructors

PSLightDriver

```
public PSLightDriver()
```

Methods

connect

```
public com.jpbc.PConnection connect(java.lang.String url,
         java.lang.String user,
         java.lang.String pwd)
  throws com.jpbc.PException
```

(continued from last page)

PBMSCompliant

```
public int PBMSCompliant()
```

com.psycholight
Class PSLightResultSet

```
java.lang.Object
   |
   +-com.psycholight.PSLightResultSet
```

All Implemented Interfaces:
com.jpbc.PResultSet

---

public class **PSLightResultSet**
extends java.lang.Object
implements com.jpbc.PResultSet

Constructor Summary

public	PSLightResultSet(java.sql.ResultSet rs)

Method Summary

void	close()
java.io.InputStream	getAsciiStream(int columnIndex)
java.io.InputStream	getAsciiStream(java.lang.String columnName)
java.io.InputStream	getBinaryStream(int columnIndex)
java.io.InputStream	getBinaryStream(java.lang.String columnName)
boolean	getBoolean(int columnIndex)
boolean	getBoolean(java.lang.String columnName)
byte	getByte(int columnIndex)
byte	getByte(java.lang.String columnName)
byte[]	getBytes(int columnIndex)
byte[]	getBytes(java.lang.String columnName)
java.sql.Date	getDate(int columnIndex)
java.sql.Date	getDate(java.lang.String columnName)
double	getDouble(int columnIndex)

double	getDouble(java.lang.String columnName)
float	getFloat(int columnIndex)
float	getFloat(java.lang.String columnName)
int	getInt(int columnIndex)
int	getInt(java.lang.String columnName)
long	getLong(int columnIndex)
long	getLong(java.lang.String columnName)
java.lang.Object	getObject(int columnIndex)
short	getShort(int columnIndex)
short	getShort(java.lang.String columnName)
java.lang.String	getString(int columnIndex)
java.lang.String	getString(java.lang.String columnName)
java.sql.Time	getTime(int columnIndex)
java.sql.Time	getTime(java.lang.String columnName)
java.sql.Timestamp	getTimestamp(int columnIndex)
java.sql.Timestamp	getTimestamp(java.lang.String columnName)
boolean	next()

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `com.jpbc.PResultSet`

close, getAsciiStream, getAsciiStream, getBinaryStream, getBinaryStream, getBoolean, getBoolean, getByte, getByte, getBytes, getBytes, getDate, getDate, getDouble, getDouble, getFloat, getFloat, getInt, getInt, getLong, getLong, getObject, getShort, getShort, getString, getString, getTime, getTime, getTimestamp, getTimestamp, next

Constructors

(continued from last page)

PSLightResultSet

```
public PSLightResultSet(java.sql.ResultSet rs)
```

Methods

next

```
public boolean next()
  throws com.jpbc.PException
```

---

close

```
public void close()
  throws com.jpbc.PException
```

---

getString

```
public java.lang.String getString(int columnIndex)
  throws com.jpbc.PException
```

---

getBoolean

```
public boolean getBoolean(int columnIndex)
  throws com.jpbc.PException
```

---

getByte

```
public byte getByte(int columnIndex)
  throws com.jpbc.PException
```

---

getShort

```
public short getShort(int columnIndex)
  throws com.jpbc.PException
```

---

getInt

```
public int getInt(int columnIndex)
  throws com.jpbc.PException
```

(continued from last page)

getLong

```
public long getLong(int columnIndex)
  throws com.jpbc.PException
```

getFloat

```
public float getFloat(int columnIndex)
  throws com.jpbc.PException
```

getDouble

```
public double getDouble(int columnIndex)
  throws com.jpbc.PException
```

getBytes

```
public byte[] getBytes(int columnIndex)
  throws com.jpbc.PException
```

getDate

```
public java.sql.Date getDate(int columnIndex)
  throws com.jpbc.PException
```

getTime

```
public java.sql.Time getTime(int columnIndex)
  throws com.jpbc.PException
```

getTimestamp

```
public java.sql.Timestamp getTimestamp(int columnIndex)
  throws com.jpbc.PException
```

getAsciiStream

```
public java.io.InputStream getAsciiStream(int columnIndex)
  throws com.jpbc.PException
```

(continued from last page)

getBinaryStream

```
public java.io.InputStream getBinaryStream(int columnIndex)
  throws com.jpbc.PException
```

getString

```
public java.lang.String getString(java.lang.String columnName)
  throws com.jpbc.PException
```

getBoolean

```
public boolean getBoolean(java.lang.String columnName)
  throws com.jpbc.PException
```

getByte

```
public byte getByte(java.lang.String columnName)
  throws com.jpbc.PException
```

getShort

```
public short getShort(java.lang.String columnName)
  throws com.jpbc.PException
```

getInt

```
public int getInt(java.lang.String columnName)
  throws com.jpbc.PException
```

getLong

```
public long getLong(java.lang.String columnName)
  throws com.jpbc.PException
```

getFloat

```
public float getFloat(java.lang.String columnName)
  throws com.jpbc.PException
```

(continued from last page)

getDouble

```
public double getDouble(java.lang.String columnName)
  throws com.jpbc.PException
```

getBytes

```
public byte[] getBytes(java.lang.String columnName)
  throws com.jpbc.PException
```

getDate

```
public java.sql.Date getDate(java.lang.String columnName)
  throws com.jpbc.PException
```

getTime

```
public java.sql.Time getTime(java.lang.String columnName)
  throws com.jpbc.PException
```

getTimestamp

```
public java.sql.Timestamp getTimestamp(java.lang.String columnName)
  throws com.jpbc.PException
```

getAsciiStream

```
public java.io.InputStream getAsciiStream(java.lang.String columnName)
  throws com.jpbc.PException
```

getBinaryStream

```
public java.io.InputStream getBinaryStream(java.lang.String columnName)
  throws com.jpbc.PException
```

getObject

```
public java.lang.Object getObject(int columnIndex)
  throws com.jpbc.PException
```

com.psycholight
Class PSLightStruct

```
java.lang.Object
   |
   +-com.psycholight.PSLightStruct
```

All Implemented Interfaces:
com.jpbc.PStruct

---

public class **PSLightStruct**
extends java.lang.Object
implements com.jpbc.PStruct

Constructor Summary

public	PSLightStruct(java.sql.Struct value)
public	PSLightStruct(java.sql.Struct value, java.lang.String typeName)

Method Summary

java.lang.Object[]	getAttributes()
java.lang.String	getTypeName()

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PStruct

getAttributes, getTypeName

Constructors

PSLightStruct

```
public PSLightStruct(java.sql.Struct value)
```

PSLightStruct

```
public PSLightStruct(java.sql.Struct value,
                     java.lang.String typeName)
```

(continued from last page)

Methods

getTypeName

```
public java.lang.String getTypeName()
```

---

getAttributes

```
public java.lang.Object[] getAttributes()
  throws com.jpbc.PException
```

Package
com.psycholight.mining

com.psycholight.mining
Class Apriori

```
java.lang.Object
   |
   +-com.psycholight.mining.Apriori
```

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

Apriori.java

This class implements the Apriori algorithm for finding large itemsets. (see "Fast Algorithms for Mining Association Rules" by Rakesh Agrawal and Ramakrishnan Srikant from IBM Almaden Research Center 1994)

Constructor Summary

public	Apriori()

Method Summary

java.util.Vector	findLargeItemsets(java.sql.ResultSet resultset, float minSupport) Find the frequent itemsets in a database
int	getPass_num()
void	setPass_num(int pass_num)

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

Apriori

```
public Apriori()
```

Methods

findLargeItemsets

```
public java.util.Vector findLargeItemsets(java.sql.ResultSet resultset,
        float minSupport)
```

Find the frequent itemsets in a database

Parameters:
dbReader - the object used to read from the database

(continued from last page)

`cacheWriter` - the object used to write to the cache if this is null, then nothing will be saved, this is useful for benchmarking
`minSupport` - the minimum support

Returns:

the number of passes executed over the database

---

getPass_num

```
public int getPass_num()
```

---

setPass_num

```
public void setPass_num(int pass_num)
```

com.psycholight.mining
Class AprioriRules

```
java.lang.Object
   |-com.psycholight.mining.MiningFunction
      |-com.psycholight.mining.AprioriRules
```

public class **AprioriRules**
extends MiningFunction

Fields inherited from class com.psycholight.mining.MiningFunction
JAVA_EXIT, NO_DEBUG_INFORMATON_INSERTED

Constructor Summary

public	AprioriRules()

Method Summary

java.util.Vector	findAssociations(java.util.Vector generatedItemsets, float minSupport, float minConfidence) Find association rules in a database, given the set of frequent itemsets.
static void	main(java.lang.String[] args)
static java.lang.String	mineAssociationRules(float minSupport, float minConfidence, java.lang.String dataSource, java.lang.String patternTypeName) Start the process of rule extraction and store the result in the Pattern Base.

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

AprioriRules

```
public AprioriRules()
```

Methods

(continued from last page)

mineAssociationRules

```
public static java.lang.String mineAssociationRules(float minSupport,
         float minConfidence,
         java.lang.String dataSource,
         java.lang.String patternTypeName)
```

Start the process of rule extraction and store the result in the Pattern Base.

Parameters:

`String` - The name of the datasource to use for the generation of the rules.

findAssociations

```
public java.util.Vector findAssociations(java.util.Vector generatedItemsets,
         float minSupport,
         float minConfidence)
```

Find association rules in a database, given the set of frequent itemsets.

Parameters:

`cacheReader` - the object used to read from the cache
`minSupport` - the minimum support
`minConfidence` - the minimum confidence

Returns:

a Vector containing all association rules found

main

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

com.psycholight.mining
Class AssociationRule

```
java.lang.Object
   |
   +-com.psycholight.mining.AssociationRule
```

All Implemented Interfaces:
java.io.Serializable

---

public class **AssociationRule**
extends java.lang.Object
implements java.io.Serializable

Field Summary

public static final	ANTECEDENT_SIZE Value: **1**
public static final	CONFIDENCE Value: **4**
public static final	CONSEQUENT_SIZE Value: **2**
public static final	SUPPORT Value: **3**

Constructor Summary

public	AssociationRule(Itemset antecedent, Itemset consequent, float support, float confidence) Creates a new association rule.

Method Summary

int	antecedentSize() Return size of antecedent.
int	compareTo(AssociationRule ar, int criteria) Compare two AssociationRule objects on one of several criteria.
int	consequentSize() Return size of consequent.
boolean	equals(java.lang.Object obj) Compare two AssociationRule objects on one of several criteria.
int	getAntecedentItem(int i) Return i-th item in antecedent.

float	getConfidence() Return confidence of association rule.
int	getConsequentItem(int i) Return i-th item in consequent.
float	getSupport() Return support of association rule.
java.lang.String	toString() Return a java.lang.String representation of the AssociationRule.

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

ANTECEDENT_SIZE

```
public static final int ANTECEDENT_SIZE
```

Constant value: **1**

CONSEQUENT_SIZE

```
public static final int CONSEQUENT_SIZE
```

Constant value: **2**

SUPPORT

```
public static final int SUPPORT
```

Constant value: **3**

CONFIDENCE

```
public static final int CONFIDENCE
```

Constant value: **4**

Constructors

AssociationRule

```
public AssociationRule(Itemset antecedent,
                       Itemset consequent,
                       float support,
                       float confidence)
```

Creates a new association rule.

(continued from last page)

Parameters:

`antecedent` - the antecedent of the association rule
`consequent` - the consequent of the association rule
`support` - the support of the association rule
`confidence` - the confidence of the association rule

Throws:

`IllegalArgumentException` - `antecedent` or `consequent` are null or `support` or `confidence` are not between 0 and 1

Methods

antecedentSize

`public int antecedentSize()`

Return size of antecedent.

Returns:

size of antecedent

---

consequentSize

`public int consequentSize()`

Return size of consequent.

Returns:

size of consequent

---

getSupport

`public float getSupport()`

Return support of association rule.

---

getConfidence

`public float getConfidence()`

Return confidence of association rule.

---

getAntecedentItem

`public int getAntecedentItem(int i)`

Return i-th item in antecedent.

Parameters:

`i` - the index of the item to get

Returns:

the `i`-th item in antecedent

Throws:

`IndexOutOfBoundsException` - `i` is an invalid index

(continued from last page)

getConsequentItem

```
public int getConsequentItem(int i)
```

Return i-th item in consequent.

Parameters:
`i` - the index of the item to get

Returns:
the `i`-th item in consequent

Throws:
`IndexOutOfBoundsException` - `i` is an invalid index

---

compareTo

```
public int compareTo(AssociationRule ar,
         int criteria)
```

Compare two AssociationRule objects on one of several criteria.

Parameters:
`ar` - the AssociationRule object with which we want to compare this object
`criteria` - the criteria on which we want to compare, can be one of ANTECEDENT_SIZE, CONSEQUENT_SIZE, SUPPORT or CONFIDENCE.

Returns:
a negative value if this object is smaller than `ar`, 0 if they are equal, and a positive value if this object is greater.

Throws:
`IllegalArgumentException` - `ar` is null or criteria is invalid

---

equals

```
public boolean equals(java.lang.Object obj)
```

Compare two AssociationRule objects on one of several criteria.

Parameters:
`ar` - the AssociationRule object with which we want to compare this object
`criteria` - the criteria on which we want to compare, can be one of ANTECEDENT_SIZE, CONSEQUENT_SIZE, SUPPORT or CONFIDENCE.

Returns:
true if the objects are equal in terms of antecedent and consequent items; false otherwise.

---

toString

```
public java.lang.String toString()
```

Return a java.lang.String representation of the AssociationRule.

Returns:
java.lang.String representation of AssociationRule

com.psycholight.mining
Class HashTree

```
java.lang.Object
   |
   +-com.psycholight.mining.HashTree
```

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

HashTree.java

A HashTree is a special data structure that is used to index a Vector of Itemset objects for more efficient processing.

Constructor Summary

public	`HashTree(int listSize, int hashSize, java.util.Vector itemsets)` Create a new HashTree.
public	`HashTree(java.util.Vector itemsets)` Create a new HashTree.

Method Summary

void	`add(int index)` This method indexes in the HashTree the Itemset at index `index` from Vector `itemsets` which was passed to the constructor of this HashTree.
void	`checkLargeness(Itemset itemset)` Verifies if any of the indexed Itemsets is not large by checking whether they're included in the frequent itemset `itemset`.
long	`countFrequentSubsets(Itemset itemset, long minWeight)` Count how many frequent Itemsets (frequent = having weight greater than a specified minimum weight) are included in `itemset`
long	`countSubsets(Itemset itemset)` Count how many Itemsets are included in `itemset`
void	`prepareForDescent()` This method should be called before calling update() to gather all leaves of the HashTree for more efficient processing.
void	`update(Itemset row)` Update the weights of all indexed Itemsets that are included in `row`
void	`update(Itemset row, long[][] counts)` Update the weights of all indexed Itemsets that are included in `row` and also update the matrix `counts`

Methods inherited from class `java.lang.Object`

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

(continued from last page)

Constructors

HashTree

```
public HashTree(int listSize,
                int hashSize,
                java.util.Vector itemsets)
```

Create a new HashTree. The `listSize` parameter determines after how many inserts in a ListNode we have to change it to a HashNode (i.e. perform a split). The `hashSize` parameter can be specified to improve the efficiency of the structure.

Parameters:

`listSize` - the size of the internal lists in the list nodes
`hashSize` - the size of the internal hashtables in the hash nodes
`itemsets` - the Vector of Itemsets that we should index

Throws:

`IllegalArgumentException` - `itemsets` is null or `listSize <= 0` or `hashSize <= 0`

HashTree

```
public HashTree(java.util.Vector itemsets)
```

Create a new HashTree. This initializes the HashTree with default parameters.

Parameters:

`itemsets` - the Vector of Itemsets that we should index

Throws:

`IllegalArgumentException` - `itemsets` is null

Methods

prepareForDescent

```
public void prepareForDescent()
```

This method should be called before calling update() to gather all leaves of the HashTree for more efficient processing.

---

add

```
public void add(int index)
```

This method indexes in the HashTree the Itemset at index `index` from Vector `itemsets` which was passed to the constructor of this HashTree.

Parameters:

`index` - the index of the Itemset that we need to index in this HashTree.

---

update

```
public void update(Itemset row)
```

Update the weights of all indexed Itemsets that are included in `row`

Parameters:

`row` - the Itemset (normally a database row) against which we test for inclusion

update

```
public void update(Itemset row,
                   long[][] counts)
```

Update the weights of all indexed Itemsets that are included in `row` and also update the matrix `counts`

Parameters:

`row` - the Itemset (normally a database row) against which we test for inclusion

`counts` - a matrix used by some algorithms to speed up computations; its rows correspond to Itemsets and its columns correspond to items; each value in the matrix tells for how many times had an item appeared together with an itemset in the rows of the database.

countFrequentSubsets

```
public long countFrequentSubsets(Itemset itemset,
                                 long minWeight)
```

Count how many frequent Itemsets (frequent = having weight greater than a specified minimum weight) are included in `itemset`

Parameters:

`itemset` - the Itemset for which we count the subsets

`minWeight` - the minimum weight

countSubsets

```
public long countSubsets(Itemset itemset)
```

Count how many Itemsets are included in `itemset`

Parameters:

`itemset` - the Itemset for which we count the subsets

checkLargeness

```
public void checkLargeness(Itemset itemset)
```

Verifies if any of the indexed Itemsets is not large by checking whether they're included in the frequent itemset `itemset`. If an Itemset is not large then it will be marked.

Parameters:

`itemset` - the Itemset we check

com.psycholight.mining
Class Itemset

```
java.lang.Object
   |
   +-com.psycholight.mining.Itemset
```

All Implemented Interfaces:
java.io.Serializable

---

public class **Itemset**
extends java.lang.Object
implements java.io.Serializable

Itemset.java

An itemset is an ordered list of integers that identify items coupled with a float value representing the support of the itemset as a percentage.

Constructor Summary

public	`Itemset()` Creates a new empty itemset.
public	`Itemset(int c)` Create a new empty itemset of specified capacity.
public	`Itemset(Itemset itemset)` Create a new itemset by copying a given one.

Method Summary

`Itemset`	`add(Itemset itemset)` Return a new Itemset that contains all those items that appear in this Itemset and in `itemset`.
boolean	`addItem(int item)` Add a new item to the itemset.
boolean	`canCombineWith(Itemset itemset)` Check whether two itemsets can be combined.
`Itemset`	`combineWith(Itemset itemset)` Combine two itemsets into a new one that will contain all the items in the first itemset plus the last item in the second itemset.
boolean	`doesIntersect(Itemset itemset)` Return true if this itemset has items in common with `itemset`.
int	`getFirstItem()` Return first item in set.
int	`getItem(int i)` Return i-th item in set.

int	getNextItem() Return next item in set.
float	getSupport() Return support of itemset.
long	getWeight() Return weight of itemset.
boolean	hasMoreItems() Return true if there are more items in the itemset.
void	incrementWeight() Increment the weight of the itemset.
boolean	isEqualTo(Itemset itemset) Checks equality with a given itemset.
boolean	isIncludedIn(Itemset itemset) Checks inclusion in a given itemset.
boolean	isMarked() Return itemset mark.
boolean	mark() Mark the itemset.
static void	pruneDuplicates(java.util.Vector v) Remove all duplicate itemsets from the vector v
static void	pruneNonMaximal(java.util.Vector v) Remove all non-maximal itemsets from the vector v
boolean	removeItem(int item) Removes a given item from the itemset.
boolean	removeLastItem() Removes last item (which has the greatest value) from the itemset.
void	setSupport(float newSupport) Set the support of the itemset.
void	setWeight(long newWeight) Set the weight of the itemset.
int	size() Return size of itemset.
Itemset	subtract(Itemset itemset) Return a new Itemset that contains only those items that do not appear in itemset.
java.lang.String	toString() Return a String representation of the Itemset.
boolean	unmark() Unmark the itemset.

Methods inherited from class java.lang.Object

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

Constructors

Itemset

`public Itemset()`

Creates a new empty itemset.

Itemset

`public Itemset(int c)`

Create a new empty itemset of specified capacity.

Parameters:
`c` - the capacity of the itemset

Throws:
`IllegalArgumentException` - `c` is negative or zero

Itemset

`public Itemset(Itemset itemset)`

Create a new itemset by copying a given one.

Parameters:
`itemset` - the itemset to be copied

Throws:
`IllegalArgumentException` - `itemset` is null

Methods

getSupport

`public float getSupport()`

Return support of itemset.

getWeight

`public long getWeight()`

Return weight of itemset.

getItem

`public int getItem(int i)`

Return i-th item in set.

Parameters:
`i` - the index of the item to get

(continued from last page)

Returns:
the `i`-th item

Throws:
`IndexOutOfBoundsException` - `i` is an invalid index

---

getFirstItem

`public int getFirstItem()`

Return first item in set.

Returns:
first item

Throws:
`IndexOutOfBoundsException` - there is no first item

---

getNextItem

`public int getNextItem()`

Return next item in set.

Returns:
next item

Throws:
`IndexOutOfBoundsException` - there is no next item

---

hasMoreItems

`public boolean hasMoreItems()`

Return true if there are more items in the itemset. You can call this method to find out whether you can call getNext without raising an exception.

Returns:
true if there are more items, false if not

---

size

`public int size()`

Return size of itemset.

Returns:
size of itemset

---

doesIntersect

`public boolean doesIntersect(Itemset itemset)`

Return true if this itemset has items in common with `itemset`.

Parameters:
`itemset` - the itemset with which we compare

Returns:

(continued from last page)

true if `itemset` contains items of this itemset, false otherwise.

Throws:
`IllegalArgumentException` - `itemset` is null

---

subtract

`public Itemset subtract(Itemset itemset)`

Return a new Itemset that contains only those items that do not appear in `itemset`.

Parameters:
`itemset` - the itemset whose items we want to subtract

Returns:
an Itemset containing only those items of this Itemset that do not appear in `itemset`.

Throws:
`IllegalArgumentException` - `itemset` is null

---

add

`public Itemset add(Itemset itemset)`

Return a new Itemset that contains all those items that appear in this Itemset and in `itemset`.

Parameters:
`itemset` - the itemset whose items we want to add

Returns:
an Itemset containing all those items that appear in this Itemset and in `itemset`.

Throws:
`IllegalArgumentException` - `itemset` is null

---

addItem

`public boolean addItem(int item)`

Add a new item to the itemset.

Parameters:
`item` - the item to be added

Returns:
true if item was added, false if it wasn't added (was already there!)

Throws:
`IllegalArgumentException` - `item` is <= 0

---

removeItem

`public boolean removeItem(int item)`

Removes a given item from the itemset.

Parameters:
`item` - the item to remove

(continued from last page)

Returns:
true if item was removed, false if it wasn't removed (was not found in itemset!)

Throws:
`IllegalArgumentException` - `item` is <= 0

---

removeLastItem

```
public boolean removeLastItem()
```

Removes last item (which has the greatest value) from the itemset.

Returns:
true if item was removed, false if it wasn't removed (the itemset was empty)

---

setSupport

```
public void setSupport(float newSupport)
```

Set the support of the itemset.

Parameters:
`newSupport` - the support of the itemset

Throws:
`IllegalArgumentException` - `newSupport` is < 0 or > 100

---

setWeight

```
public void setWeight(long newWeight)
```

Set the weight of the itemset.

Parameters:
`newWeight` - the weight of the itemset

Throws:
`IllegalArgumentException` - `newWeight` is < 0

---

incrementWeight

```
public void incrementWeight()
```

Increment the weight of the itemset.

---

isEqualTo

```
public boolean isEqualTo(Itemset itemset)
```

Checks equality with a given itemset.

Parameters:
`itemset` - the itemset against which we test for equality

Throws:
`IllegalArgumentException` - `itemset` is null

(continued from last page)

isIncludedIn

`public boolean isIncludedIn(Itemset itemset)`

Checks inclusion in a given itemset.

Parameters:
`itemset` - the itemset against which we test for inclusion

Throws:
`IllegalArgumentException` - `itemset` is null

---

mark

`public boolean mark()`

Mark the itemset.

Returns:
true if itemset was already marked, false otherwise

---

unmark

`public boolean unmark()`

Unmark the itemset.

Returns:
true if itemset was marked, false otherwise

---

isMarked

`public boolean isMarked()`

Return itemset mark.

Returns:
true if itemset is marked, false otherwise

---

toString

`public java.lang.String toString()`

Return a String representation of the Itemset.

Returns:
String representation of Itemset

---

canCombineWith

`public boolean canCombineWith(Itemset itemset)`

Check whether two itemsets can be combined. Two itemsets can be combined if they differ only in the last item.

Parameters:
`itemset` - itemset with which to combine

Returns:

(continued from last page)

true if the itemsets can be combined, false otherwise

Throws:

`IllegalArgumentException` - `itemset` is null

---

combineWith

`public Itemset combineWith(Itemset itemset)`

Combine two itemsets into a new one that will contain all the items in the first itemset plus the last item in the second itemset.

Parameters:

`itemset` - itemset with which to combine

Returns:

an itemset that combines the two itemsets as described above

Throws:

`IllegalArgumentException` - `itemset` is null

---

pruneNonMaximal

`public static void pruneNonMaximal(java.util.Vector v)`

Remove all non-maximal itemsets from the vector v

Parameters:

`v` - the collection of itemsets

---

pruneDuplicates

`public static void pruneDuplicates(java.util.Vector v)`

Remove all duplicate itemsets from the vector v

Parameters:

`v` - the collection of itemsets

com.psycholight.mining
Class MiningFunction

```
java.lang.Object
   |
   +-com.psycholight.mining.MiningFunction
```

Direct Known Subclasses:
AprioriRules, UsoApriori, UsoKMeans

---

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

Field Summary

public static	JAVA_EXIT
public static	NO_DEBUG_INFORMATON_INSERTED

Constructor Summary

public	MiningFunction()

Methods inherited from class `java.lang.Object`

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

Fields

NO_DEBUG_INFORMATON_INSERTED

`public static int` **`NO_DEBUG_INFORMATON_INSERTED`**

JAVA_EXIT

`public static int` **`JAVA_EXIT`**

Constructors

MiningFunction

`public` **`MiningFunction`**`()`

com.psycholight.mining
Class SET

```
java.lang.Object
   |
   +-com.psycholight.mining.SET
```

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

SET.java

Implements a Set Enumeration Tree, which is a prefix tree used for storing and retrieving itemset information.

Constructor Summary

public	SET() Create a new empty SET.

Method Summary

java.util.Vector	getItemsets() Return the itemsets of the SET.
java.util.Vector	getLargeItemsets() Return the maximal itemsets of the SET.
float	getSupport(Itemset itemset) Return the support for a given itemset.
void	insert(Itemset itemset) Insert a new itemset in the SET.

Methods inherited from class `java.lang.Object`

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

Constructors

SET

`public SET()`

Create a new empty SET.

Methods

insert

`public void insert(Itemset itemset)`

Insert a new itemset in the SET.

(continued from last page)

Parameters:
`itemset` - the itemset to be inserted

Throws:
`IllegalArgumentException` - `itemset` is null or is empty

---

getSupport

```
public float getSupport(Itemset itemset)
  throws java.util.NoSuchElementException
```

Return the support for a given itemset.

Parameters:
`itemset` - the itemset for which we want to obtain the support

Returns:
support

Throws:
`IllegalArgumentException` - `itemset` is null or is empty
`NoSuchElementException` - `itemset` not found in SET

---

getLargeItemsets

```
public java.util.Vector getLargeItemsets()
```

Return the maximal itemsets of the SET.

Returns:
a vector containing the maximal itemsets from the SET

---

getItemsets

```
public java.util.Vector getItemsets()
```

Return the itemsets of the SET.

Returns:
a vector containing the itemsets from the SET

com.psycholight.mining
Class UsoApriori

```
java.lang.Object
   |-com.psycholight.mining.MiningFunction
      |-com.psycholight.mining.UsoApriori
```

public class **UsoApriori**
extends MiningFunction

Field Summary

public static	dataSource
public static final	TIMEOUT_SECONDS Value: **10**

Fields inherited from class com.psycholight.mining.MiningFunction

JAVA_EXIT, NO_DEBUG_INFORMATON_INSERTED

Constructor Summary

public	UsoApriori()

Method Summary

static void	buildARModel(oracle.dmt.odm.Connection m_dmsConn, float minSupport, float minConfidence) Create a new Association Rule model and store it in the Data Mining Server.
static void	closeConnection(oracle.dmt.odm.Connection m_dmsConn) Close the current DM connection.
static boolean	executeTask(oracle.dmt.odm.Connection m_dmsConn, oracle.dmt.odm.task.MiningTask taskObj, java.lang.String taskName) Execute a task in order to perform the chosen operation.
static oracle.dmt.odm.LocationCellAccessData	exportModel(oracle.dmt.odm.Connection m_dmsConn, java.lang.String modelName) Export an Association Rule model previously stored as a PMML Model stored in a column of type XMLTYPE of a new table.
static java.lang.String	fillPatternBase(oracle.dmt.odm.Connection m_dmsConn, java.lang.String modelName, java.lang.String patternTypeName) Retrieve rules from a stored Association Rule Model and fills the PBMS in.
static java.lang.String	importModel(oracle.dmt.odm.Connection m_dmsConn) Import a PMML Model from an XML document stored in a table as an Association Rule model in the Data Mining Server.

static void	main(java.lang.String[] args)
static java.lang.String	mineRules(float minSupport, float minConfidence, java.lang.String dsName, java.lang.String patternTypeName) Start the process of rule extraction and store the result in the Pattern Base.
static oracle.dmt.odm.Connection	openConnection() Create an instance of the DMS server from the existing JDBC connection and get a DM connection.
static void	print(oracle.dmt.odm.Connection m_dmsConn, java.lang.String modelName)

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

TIMEOUT_SECONDS

```
public static final int TIMEOUT_SECONDS
```

Constant value: **10**

dataSource

```
public static java.lang.String dataSource
```

Constructors

UsoApriori

```
public UsoApriori()
```

Methods

openConnection

```
public static oracle.dmt.odm.Connection openConnection()
  throws oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.ODMException,
         java.sql.SQLException
```

Create an instance of the DMS server from the existing JDBC connection and get a DM connection.

Returns:
Connection

Throws:
InvalidArgumentException
ODMException

(continued from last page)

SQLException

closeConnection

```
public static void closeConnection(oracle.dmt.odm.Connection m_dmsConn)
  throws oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.ODMException
```

Close the current DM connection.

Parameters:

m_dmsConn - Connection

Throws:

InvalidArgumentException

ODMException

buildARModel

```
public static void buildARModel(oracle.dmt.odm.Connection m_dmsConn,
          float minSupport,
          float minConfidence)
  throws oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.ODMException,
         java.sql.SQLException,
         java.io.IOException
```

Create a new Association Rule model and store it in the Data Mining Server.

Parameters:

m_dmsConn - Connection

Throws:

InvalidArgumentException

ODMException

SQLException

IOException

print

```
public static void print(oracle.dmt.odm.Connection m_dmsConn,
          java.lang.String modelName)
  throws oracle.dmt.odm.MiningObjectException,
         oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.ODMException,
         java.sql.SQLException
```

exportModel

```
public static oracle.dmt.odm.LocationCellAccessData
exportModel(oracle.dmt.odm.Connection m_dmsConn,
          java.lang.String modelName)
  throws oracle.dmt.odm.ODMException,
         java.sql.SQLException,
         java.io.IOException
```

Export an Association Rule model previously stored as a PMML Model stored in a column of type XMLTYPE of a new table.

Parameters:

(continued from last page)

`m_dmsConn` - Connection The connection to the DM Server
`modelName` - String The name of the model to export

Returns:
The location where the model has been exported

Throws:
`ODMException`
`SQLException`
`IOException`

---

importModel

```
public static java.lang.String importModel(oracle.dmt.odm.Connection m_dmsConn)
  throws oracle.dmt.odm.ODMException,
         java.sql.SQLException,
         java.io.IOException
```

Import a PMML Model from an XML document stored in a table as an Association Rule model in the Data Mining Server.

Parameters:
`m_dmsConn` - Connection The connection to the DM Server
`modelName` - String The name of the model to export

Returns:
The location where the model has been exported

Throws:
`ODMException`
`SQLException`
`IOException`

---

executeTask

```
public static boolean executeTask(oracle.dmt.odm.Connection m_dmsConn,
          oracle.dmt.odm.task.MiningTask taskObj,
          java.lang.String taskName)
  throws oracle.dmt.odm.ODMException,
         java.sql.SQLException
```

Execute a task in order to perform the chosen operation. This can be the generation of a model, its import or export.

Parameters:
`m_dmsConn` - Connection The connection to the DM Server
`modelName` - String The name of the task to execute
`taskName` - String

Returns:
boolean

Throws:
`ODMException`
`SQLException`

(continued from last page)

fillPatternBase

```
public static java.lang.String fillPatternBase(oracle.dmt.odm.Connection m_dmsConn,
         java.lang.String modelName,
         java.lang.String patternTypeName)
  throws java.sql.SQLException,
         oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.MiningObjectException,
         oracle.dmt.odm.ODMException
```

Retrieve rules from a stored Association Rule Model and fills the PBMS in.

Parameters:

`m_dmsConn` - Connection The connection to the DM Server
`modelName` - String The name of the task to execute

Returns:

String

Throws:

`SQLException`
`InvalidArgumentException`
`MiningObjectException`
`ODMException`

---

mineRules

```
public static java.lang.String mineRules(float minSupport,
         float minConfidence,
         java.lang.String dsName,
         java.lang.String patternTypeName)
```

Start the process of rule extraction and store the result in the Pattern Base.

Parameters:

`String` - The name of the datasource to use for the generation of the rules.

---

main

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

com.psycholight.mining
Class UsoKMeans

```
java.lang.Object
   |-com.psycholight.mining.MiningFunction
      |-com.psycholight.mining.UsoKMeans
```

public class **UsoKMeans**
extends MiningFunction

Field Summary

public	dataSource

Fields inherited from class com.psycholight.mining.MiningFunction

JAVA_EXIT, NO_DEBUG_INFORMATON_INSERTED

Constructor Summary

public	UsoKMeans()

Method Summary

static void	buildModel(oracle.dmt.odm.Connection m_dmsConn, java.lang.String dataSource, int kValue)
static void	cleanup(oracle.dmt.odm.Connection m_dmsConn) Cleanup the previously created objects if any.
static void	closeConnection(oracle.dmt.odm.Connection m_dmsConn) Close the current DM connection.
static boolean	executeTask(oracle.dmt.odm.Connection m_dmsConn, oracle.dmt.odm.task.MiningTask taskObj, java.lang.String taskName) This method stores the given task with the specified name in the DMS and submits the task for asynchronous execution in the DMS.
static java.lang.String	fillPatternBaseConvex(oracle.dmt.odm.Connection m_dmsConn, java.lang.String dataSource, java.lang.String patternTypeName)
static java.lang.String	fillPatternBaseExtensional(oracle.dmt.odm.Connection m_dmsConn, java.lang.String dataSource, java.lang.String patternTypeName)
static java.lang.String	kMeansConvex(int kValue, java.lang.String dataSource, java.lang.String patternTypeName)
static java.lang.String	kMeansExtensional(int kValue, java.lang.String dataSource, java.lang.String patternTypeName)

static void	main(java.lang.String[] args)
static oracle.dmt.odm.Connection	openConnection() Create an instance of the DMS server from the existing JDBC connection and get a DM connection.

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

dataSource

```
public java.lang.String dataSource
```

Constructors

UsoKMeans

```
public UsoKMeans()
```

Methods

openConnection

```
public static oracle.dmt.odm.Connection openConnection()
  throws oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.ODMException,
         java.sql.SQLException
```

Create an instance of the DMS server from the existing JDBC connection and get a DM connection.

Returns:

Connection

Throws:

InvalidArgumentException
ODMException
SQLException

closeConnection

```
public static void closeConnection(oracle.dmt.odm.Connection m_dmsConn)
  throws oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.ODMException
```

Close the current DM connection.

Parameters:

m_dmsConn - Connection

(continued from last page)

Throws:

`InvalidArgumentException`

`ODMException`

buildModel

```
public static void buildModel(oracle.dmt.odm.Connection m_dmsConn,
         java.lang.String dataSource,
         int kValue)
  throws oracle.dmt.odm.ODMException,
         java.sql.SQLException,
         java.io.IOException
```

executeTask

```
public static boolean executeTask(oracle.dmt.odm.Connection m_dmsConn,
         oracle.dmt.odm.task.MiningTask taskObj,
         java.lang.String taskName)
  throws oracle.dmt.odm.ODMException,
         java.sql.SQLException
```

This method stores the given task with the specified name in the DMS and submits the task for asynchronous execution in the DMS. After completing the task successfully it returns true. If there is a task failure, then it prints error description and returns false.

Parameters:

`taskObj` - task object

`taskName` - name of the task

Returns:

boolean returns true when the task is successful

fillPatternBaseExtensional

```
public static java.lang.String fillPatternBaseExtensional(oracle.dmt.odm.Connection
m_dmsConn,
         java.lang.String dataSource,
         java.lang.String patternTypeName)
  throws java.sql.SQLException,
         oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.MiningObjectException,
         oracle.dmt.odm.ODMException
```

fillPatternBaseConvex

```
public static java.lang.String fillPatternBaseConvex(oracle.dmt.odm.Connection
m_dmsConn,
         java.lang.String dataSource,
         java.lang.String patternTypeName)
  throws java.sql.SQLException,
         oracle.dmt.odm.InvalidArgumentException,
         oracle.dmt.odm.MiningObjectException,
         oracle.dmt.odm.ODMException
```

(continued from last page)

cleanup

```
public static void cleanup(oracle.dmt.odm.Connection m_dmsConn)
  throws oracle.dmt.odm.ODMException,
         java.sql.SQLException
```

Cleanup the previously created objects if any.

---

kMeansConvex

```
public static java.lang.String kMeansConvex(int kValue,
         java.lang.String dataSource,
         java.lang.String patternTypeName)
```

---

kMeansExtensional

```
public static java.lang.String kMeansExtensional(int kValue,
         java.lang.String dataSource,
         java.lang.String patternTypeName)
```

---

main

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

Package
com.psycholight.statements

com.psycholight.statements
Class CreateClass

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.CreateClass
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateClass**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateClass(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateClass
public	CreateClass(java.lang.String ClassName, java.lang.String PatternTypeName)

Method Summary

java.lang.String	getClassName()
java.lang.String	getPatternTypeName()
java.lang.String	interpretStatement()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

CreateClass

```
public CreateClass(java.sql.Connection conn,
                   java.lang.String cmd)
```

Creates a new instance of CreateClass

CreateClass

```
public CreateClass(java.lang.String ClassName,
                   java.lang.String PatternTypeName)
```

Methods

getClassName

```
public java.lang.String getClassName()
```

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

com.psycholight.statements
Class CreateCompositionFunction

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreateCompositionFunction
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateCompositionFunction**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateCompositionFunction(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateCompositionFunction

Method Summary

java.lang.String	getCode()
java.lang.String	getFunctionName()
CreatePatternType	getNewType()
java.util.ArrayList	getParamFields()
java.lang.String	getPatternName()
java.lang.String	getPatternTypeName()
java.lang.String	getPNameInput1()
java.lang.String	getPNameInput2()
java.lang.String	getPTNameInput1()
java.lang.String	getPTNameInput2()

java.lang.String	interpretStatement() A composition function is translated in a stored function of PL/SQL that takes in input two patterns and returns a new one combining them as described in the body of the function.
boolean	isDefinedNew()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement

executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement

executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Constructors

CreateCompositionFunction

```
public CreateCompositionFunction(java.sql.Connection conn,
                                 java.lang.String cmd)
```

Creates a new instance of CreateCompositionFunction

Methods

(continued from last page)

getFunctionName

```
public java.lang.String getFunctionName()
```

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getPatternName

```
public java.lang.String getPatternName()
```

getPTNameInput1

```
public java.lang.String getPTNameInput1()
```

getPNameInput1

```
public java.lang.String getPNameInput1()
```

getPTNameInput2

```
public java.lang.String getPTNameInput2()
```

getPNameInput2

```
public java.lang.String getPNameInput2()
```

getNewType

```
public CreatePatternType getNewType()
```

isDefinedNew

```
public boolean isDefinedNew()
```

getParamFields

```
public java.util.ArrayList getParamFields()
```

(continued from last page)

getCode

```
public java.lang.String getCode()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

A composition function is translated in a stored function of PL/SQL that takes in input two patterns and returns a new one combining them as described in the body of the function.

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class CreateCondition

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreateCondition
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateCondition**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateCondition(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateClass

Method Summary

java.lang.String	getCode()
java.lang.String	getConditionName()
java.util.ArrayList	getParamFields()
java.lang.String	getPatternName()
java.lang.String	getPatternTypeName()
java.lang.String	interpretStatement()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

CreateCondition

```
public CreateCondition(java.sql.Connection conn,
                       java.lang.String cmd)
```

Creates a new instance of CreateClass

Methods

getConditionName

```
public java.lang.String getConditionName()
```

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getPatternName

```
public java.lang.String getPatternName()
```

(continued from last page)

getCode

```
public java.lang.String getCode()
```

getParamFields

```
public java.util.ArrayList getParamFields()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

com.psycholight.statements
Class CreateFormula

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreateFormula
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateFormula**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateFormula(java.sql.Connection conn, java.lang.String cmd)

Method Summary

java.lang.String	getCode()
java.util.ArrayList	getParamFields()
java.lang.String	getPatternName()
java.lang.String	getPatternTypeName()
java.lang.String	interpretStatement()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

CreateFormula

```
public CreateFormula(java.sql.Connection conn,
                     java.lang.String cmd)
```

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getPatternName

```
public java.lang.String getPatternName()
```

getParamFields

```
public java.util.ArrayList getParamFields()
```

getCode

```
public java.lang.String getCode()
```

(continued from last page)

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class CreateJoinCondition

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreateJoinCondition
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateJoinCondition**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateJoinCondition(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateClass

Method Summary

java.lang.String	getCode()
java.lang.String	getConditionName()
java.util.ArrayList	getParamFields()
java.lang.String	getPatternName1()
java.lang.String	getPatternName2()
java.lang.String	getPatternTypeName1()
java.lang.String	getPatternTypeName2()
java.lang.String	interpretStatement() A join condition is translated in a stored function of PL/SQL that takes in input two patterns and returns an integer.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement

executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement

executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Constructors

CreateJoinCondition

```
public CreateJoinCondition(java.sql.Connection conn,
                           java.lang.String cmd)
```

Creates a new instance of CreateClass

Methods

getConditionName

```
public java.lang.String getConditionName()
```

getPatternTypeName1

```
public java.lang.String getPatternTypeName1()
```

(continued from last page)

getPatternName1

```
public java.lang.String getPatternName1()
```

getCode

```
public java.lang.String getCode()
```

getParamFields

```
public java.util.ArrayList getParamFields()
```

getPatternTypeName2

```
public java.lang.String getPatternTypeName2()
```

getPatternName2

```
public java.lang.String getPatternName2()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

A join condition is translated in a stored function of PL/SQL that takes in input two patterns and returns an integer.

Parameters:

`CreateJoinCondition` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class CreateMeasureFunction

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreateMeasureFunction
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateMeasureFunction**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateMeasureFunction(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateMiningFunction

Method Summary

java.lang.String	getCode()
java.lang.String	getDataSourceName()
java.lang.String	getFunctionName()
java.util.ArrayList	getMeasureReturn()
java.util.ArrayList	getParamFields()
java.lang.String	getPatternName()
java.lang.String	getPatternTypeName()
java.lang.String	interpretStatement() A measure function is translated in a stored function of PL/SQL with exaclty the behaviour defined in the body and that returns a measure object.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement

executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement

executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Constructors

CreateMeasureFunction

```
public CreateMeasureFunction(java.sql.Connection conn,
                             java.lang.String cmd)
```

Creates a new instance of CreateMiningFunction

Methods

getFunctionName

```
public java.lang.String getFunctionName()
```

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

(continued from last page)

getPatternName

```
public java.lang.String getPatternName()
```

getDataSourceName

```
public java.lang.String getDataSourceName()
```

getParamFields

```
public java.util.ArrayList getParamFields()
```

getCode

```
public java.lang.String getCode()
```

getMeasureReturn

```
public java.util.ArrayList getMeasureReturn()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

A measure function is translated in a stored function of PL/SQL with exaclty the behaviour defined in the body and that returns a measure object.

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class CreateMiningFunction

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreateMiningFunction
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreateMiningFunction**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreateMiningFunction(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateMiningFunction

Method Summary

java.lang.String	getCode()
java.lang.String	getDataSourceName()
java.lang.String	getFunctionName()
java.util.ArrayList	getMiningFields()
java.util.ArrayList	getParamFields()
java.lang.String	getPatternTypeName()
java.lang.String	getTableName()
java.lang.String	interpretStatement() A mining function is translated in a stored function of PL/SQL with exaclty the behaviour defined in the body.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement

executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement

executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Constructors

CreateMiningFunction

```
public CreateMiningFunction(java.sql.Connection conn,
                            java.lang.String cmd)
```

Creates a new instance of CreateMiningFunction

Methods

getCode

```
public java.lang.String getCode()
```

getFunctionName

```
public java.lang.String getFunctionName()
```

(continued from last page)

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getDataSourceName

```
public java.lang.String getDataSourceName()
```

getParamFields

```
public java.util.ArrayList getParamFields()
```

getMiningFields

```
public java.util.ArrayList getMiningFields()
```

getTableName

```
public java.lang.String getTableName()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

A mining function is translated in a stored function of PL/SQL with exaclty the behaviour defined in the body.

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class CreatePatternType

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
            |-com.psycholight.statements.CreatePatternType
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **CreatePatternType**
extends StatementWithCode

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	CreatePatternType(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreatePatternType
public	CreatePatternType(java.sql.Connection conn, java.lang.String patternName, java.io.StreamTokenizer st)

Method Summary

java.lang.String	getEqualsCode()
java.util.List	getEqualsFields()
java.lang.String	getEqualsParameter()
java.lang.String	getFormulaCode()
java.util.List	getFormulaFields()
java.lang.String	getFormulaName()
java.lang.String	getFormulaParam()
java.util.List	getMeasureFields()
java.lang.String	getName()

java.lang.String	getReturnName() Getter & Setter
java.util.List	getStructureFields()
java.lang.String	getThetaCode()
java.util.List	getThetaFields()
java.lang.String	getThetaParameter()
boolean	hasCode()
java.lang.String	interpretStatement() The creation of a new pattern type is translated into several PL/SQL instructions, that are the creation of the types (also those corresponding to the structure and the measure), the declaration of the bodies of their methods, and the creation of the table typed on that pattern type.
void	parseStatement()
void	setEqualsCode(java.lang.String EqualsCode)
void	setEqualsFields(java.util.List EqualsFields)
void	setEqualsParameter(java.lang.String EqualsParameter)
void	setFormulaCode(java.lang.String FormulaCode)
void	setFormulaFields(java.util.List FormulaFields)
void	setFormulaName(java.lang.String FormulaName)
void	setFormulaParam(java.lang.String FormulaParam)
void	setHasCode(boolean hasFormulaCode)
void	setMeasureFields(java.util.List MeasureFields)
void	setName(java.lang.String Name)
void	setReturnName(java.lang.String ReturnName)
void	setStructureFields(java.util.List StructureFields)
void	setThetaCode(java.lang.String ThetaCode)
void	setThetaFields(java.util.List ThetaFields)
void	setThetaParameter(java.lang.String ThetaParameter)

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

CreatePatternType

```
public CreatePatternType(java.sql.Connection conn,
                         java.lang.String cmd)
```

Creates a new instance of CreatePatternType

CreatePatternType

```
public CreatePatternType(java.sql.Connection conn,
                         java.lang.String patternName,
                         java.io.StreamTokenizer st)
```

Methods

getReturnName

```
public java.lang.String getReturnName()
```

(continued from last page)

Getter & Setter

setReturnName

```
public void setReturnName(java.lang.String ReturnName)
```

getName

```
public java.lang.String getName()
```

setName

```
public void setName(java.lang.String Name)
```

getStructureFields

```
public java.util.List getStructureFields()
```

setStructureFields

```
public void setStructureFields(java.util.List StructureFields)
```

getMeasureFields

```
public java.util.List getMeasureFields()
```

setMeasureFields

```
public void setMeasureFields(java.util.List MeasureFields)
```

getFormulaFields

```
public java.util.List getFormulaFields()
```

setFormulaFields

```
public void setFormulaFields(java.util.List FormulaFields)
```

(continued from last page)

getEqualsFields

public java.util.List **getEqualsFields**()

setEqualsFields

public void **setEqualsFields**(java.util.List EqualsFields)

getEqualsCode

public java.lang.String **getEqualsCode**()

setEqualsCode

public void **setEqualsCode**(java.lang.String EqualsCode)

getThetaFields

public java.util.List **getThetaFields**()

setThetaFields

public void **setThetaFields**(java.util.List ThetaFields)

getThetaCode

public java.lang.String **getThetaCode**()

setThetaCode

public void **setThetaCode**(java.lang.String ThetaCode)

getFormulaCode

public java.lang.String **getFormulaCode**()

setFormulaCode

public void **setFormulaCode**(java.lang.String FormulaCode)

(continued from last page)

setThetaParameter

public void **setThetaParameter**(java.lang.String ThetaParameter)

setEqualsParameter

public void **setEqualsParameter**(java.lang.String EqualsParameter)

getThetaParameter

public java.lang.String **getThetaParameter**()

getEqualsParameter

public java.lang.String **getEqualsParameter**()

getFormulaName

public java.lang.String **getFormulaName**()

setFormulaName

public void **setFormulaName**(java.lang.String FormulaName)

hasCode

public boolean **hasCode**()

setHasCode

public void **setHasCode**(boolean hasFormulaCode)

getFormulaParam

public java.lang.String **getFormulaParam**()

(continued from last page)

setFormulaParam

```
public void setFormulaParam(java.lang.String FormulaParam)
```

---

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

---

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

The creation of a new pattern type is translated into several PL/SQL instructions, that are the creation of the types (also those corresponding to the structure and the measure), the declaration of the bodies of their methods, and the creation of the table typed on that pattern type.

Parameters:

`CreatePatternType` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class DataCovering

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PQLStatement
         |-com.psycholight.statements.DataCovering
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PqlStatement

---

public class **DataCovering**
extends PQLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DataCovering(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of DataCovering

Method Summary

java.lang.String	getClassName()
java.lang.String	getCondition()
java.util.Vector	getCondParams()
java.lang.String	getDatasource()
Select	getInnerSelect()
java.lang.String	getStoreAs()
com.jpbc.PResultSet	interpretStatement() It is translated in a call to a PL/SQL proedure that performs the query.
boolean	isSelect()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PQLStatement
executePQLcommand, executePQLcommand, getStatementType, interpretStatement

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PqlStatement
executePQLcommand, executePQLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DataCovering

```
public DataCovering(java.sql.Connection conn,
                    java.lang.String cmd)
```

Creates a new instance of DataCovering

Methods

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

getClassName

```
public java.lang.String getClassName()
```

getCondition

```
public java.lang.String getCondition()
```

getDatasource

```
public java.lang.String getDatasource()
```

getStoreAs

```
public java.lang.String getStoreAs()
```

isSelect

```
public boolean isSelect()
```

getInnerSelect

```
public Select getInnerSelect()
```

getCondParams

```
public java.util.Vector getCondParams()
```

interpretStatement

```
public com.jpbc.PResultSet interpretStatement()
  throws com.jpbc.PException
```

It is translated in a call to a PL/SQL proedure that performs the query.

Parameters:

`DataCovering` - The statement to translate

Returns:

The result of the query. Null in case of errors.

com.psycholight.statements
Class DeleteFromClass

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.DeleteFromClass
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **DeleteFromClass**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DeleteFromClass(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateClass
public	DeleteFromClass(java.lang.String ClassName, java.lang.String Condition)

Method Summary

java.lang.String	getAlias()
java.lang.String	getClassName()
java.lang.String	getCondition()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL proedure that performs the deletion.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DeleteFromClass

```
public DeleteFromClass(java.sql.Connection conn,
                       java.lang.String cmd)
```

Creates a new instance of CreateClass

DeleteFromClass

```
public DeleteFromClass(java.lang.String ClassName,
                       java.lang.String Condition)
```

Methods

getClassName

```
public java.lang.String getClassName()
```

getCondition

```
public java.lang.String getCondition()
```

getAlias

```
public java.lang.String getAlias()
```

(continued from last page)

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL proedure that performs the deletion.

Parameters:

`DeleteFromClass` - The statement to translate

Returns:

The number of updated patterns

com.psycholight.statements
Class DeletePatterns

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.DeletePatterns
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **DeletePatterns**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DeletePatterns(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of NewClass

Method Summary

java.lang.String	getAlias()
java.lang.String	getCondition()
java.lang.String	getPatternTypeName()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL proedure that performs the deletion.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DeletePatterns

```
public DeletePatterns(java.sql.Connection conn,
                      java.lang.String cmd)
```

Creates a new instance of NewClass

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getCondition

```
public java.lang.String getCondition()
```

getAlias

```
public java.lang.String getAlias()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

(continued from last page)

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL proedure that performs the deletion.

Parameters:

`DeletePatterns` - The statement to translate

Returns:

The number of deleted patterns

com.psycholight.statements
Class DirectInsertPattern

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.DirectInsertPattern
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **DirectInsertPattern**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DirectInsertPattern(java.sql.Connection conn, java.lang.String cmd)

Method Summary

java.lang.String	getDataSourceName()
java.lang.String	getIntoClass()
java.lang.String	getPatternTypeName()
java.lang.String	getStructureValue()
java.lang.String	getThresoldDefinition()
java.lang.String	getTsFinal()
java.lang.String	getTsInit()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL proedure that performs the insertion.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DirectInsertPattern

```
public DirectInsertPattern(java.sql.Connection conn,
                           java.lang.String cmd)
```

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getDataSourceName

```
public java.lang.String getDataSourceName()
```

getTsInit

```
public java.lang.String getTsInit()
```

(continued from last page)

getTsFinal

```
public java.lang.String getTsFinal()
```

getThresoldDefinition

```
public java.lang.String getThresoldDefinition()
```

getStructureValue

```
public java.lang.String getStructureValue()
```

getIntoClass

```
public java.lang.String getIntoClass()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL proedure that performs the insertion.

Parameters:

`DirectInsertPattern` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class DrillThrough

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PQLStatement
         |-com.psycholight.statements.DrillThrough
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PqlStatement

---

public class **DrillThrough**
extends PQLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DrillThrough(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of DrillThrough

Method Summary

java.lang.String	getClassName()
java.lang.String	getCondition()
java.util.Vector	getCondParams()
Select	getInnerSelect()
java.lang.String	getTableName()
com.jpbc.PResultSet	interpretStatement() It is translated in a call to a PL/SQL proedure that performs the query.
boolean	isSelect()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PQLStatement
executePQLcommand, executePQLcommand, getStatementType, interpretStatement

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PqlStatement
executePQLcommand, executePQLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DrillThrough

```
public DrillThrough(java.sql.Connection conn,
                    java.lang.String cmd)
```

Creates a new instance of DrillThrough

Methods

getCondition

```
public java.lang.String getCondition()
```

getTableName

```
public java.lang.String getTableName()
```

getClassName

```
public java.lang.String getClassName()
```

getInnerSelect

```
public Select getInnerSelect()
```

(continued from last page)

isSelect

```
public boolean isSelect()
```

getCondParams

```
public java.util.Vector getCondParams()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public com.jpbc.PResultSet interpretStatement()
  throws com.jpbc.PException
```

It is translated in a call to a PL/SQL proedure that performs the query.

Parameters:

`DrillThrough` - The statement to translate

Returns:

The result of the query. Null in case of errors.

com.psycholight.statements
Class DropClass

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.DropClass
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **DropClass**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DropClass(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of DropClass

Method Summary

java.lang.String	getName()
java.lang.String	interpretStatement() This is translated in a call to a stored function of PL/SQL that performs the deletion.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DropClass

```
public DropClass(java.sql.Connection conn,
                 java.lang.String cmd)
```

Creates a new instance of DropClass

Methods

getName

```
public java.lang.String getName()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a stored function of PL/SQL that performs the deletion.

Parameters:

`DropClass` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class DropMeasureFunction

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.DropMeasureFunction
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **DropMeasureFunction**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DropMeasureFunction(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of DropMiningFunction

Method Summary

java.lang.String	getName()
java.lang.String	interpretStatement() This is translated in the PL/SQL statement to delete the stored function.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DropMeasureFunction

```
public DropMeasureFunction(java.sql.Connection conn,
                           java.lang.String cmd)
```

Creates a new instance of DropMiningFunction

Methods

getName

```
public java.lang.String getName()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

This is translated in the PL/SQL statement to delete the stored function.

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class DropPatternType

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.DropPatternType
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **DropPatternType**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	DropPatternType(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreatePatternType

Method Summary

java.lang.String	getName() Getter & Setter
java.lang.String	interpretStatement() The deletion of a pattern type is translated in the deletion of the three base types defined and the typed class created.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

DropPatternType

```
public DropPatternType(java.sql.Connection conn,
                       java.lang.String cmd)
```

Creates a new instance of CreatePatternType

Methods

getName

```
public java.lang.String getName()
```

Getter & Setter

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.String interpretStatement()
  throws com.jpbc.PException
```

The deletion of a pattern type is translated in the deletion of the three base types defined and the typed class created.

Parameters:

`DropPatternType` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class ExecSQL

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.ExecSQL
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement

---

public class **ExecSQL**
extends PBMSStatement

This class wraps an EXEC SQL statement, it executes pure SQL or PL/pgSQL code

---

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	ExecSQL(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of ExecSQL

Method Summary

java.lang.String	getSQLCode()
int	getStatementType()
java.lang.Object	interpretStatement() Execute the SQL or PL/pgSQL statement
void	parseStatement()

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

(continued from last page)

ExecSQL

```
public ExecSQL(java.sql.Connection conn,
               java.lang.String cmd)
```

Creates a new instance of ExecSQL

Methods

getSQLCode

```
public java.lang.String getSQLCode()
```

---

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

---

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

Execute the SQL or PL/pgSQL statement

Returns:

An object containing the query result

Throws:

`PException` - if there was any error in the statement execution

---

getStatementType

```
public int getStatementType()
```

com.psycholight.statements
Class ExtractPattern

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.ExtractPattern
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **ExtractPattern**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	ExtractPattern(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of ExtractPatternType

Method Summary

java.lang.String	getAlias()
java.lang.String	getCondition()
java.lang.String	getDataSourceName()
java.lang.String	getExtParams()
java.lang.String	getIntoClass()
java.lang.String	getMeasureFunction()
java.util.ArrayList	getMeasureValues()
java.lang.String	getMiningFunctionName()
java.lang.String	getPatternTypeName()
java.lang.String	getThresoldDefinition()
java.lang.String	getTsFinal()

java.lang.String	getTsInit()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL proedure that performs the extraction.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement

executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement

executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Constructors

ExtractPattern

```
public ExtractPattern(java.sql.Connection conn,
                      java.lang.String cmd)
```

Creates a new instance of ExtractPatternType

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

(continued from last page)

getDataSourceName

```
public java.lang.String getDataSourceName()
```

getMiningFunctionName

```
public java.lang.String getMiningFunctionName()
```

getCondition

```
public java.lang.String getCondition()
```

getTsInit

```
public java.lang.String getTsInit()
```

getTsFinal

```
public java.lang.String getTsFinal()
```

getThresoldDefinition

```
public java.lang.String getThresoldDefinition()
```

getMeasureValues

```
public java.util.ArrayList getMeasureValues()
```

getMeasureFunction

```
public java.lang.String getMeasureFunction()
```

getIntoClass

```
public java.lang.String getIntoClass()
```

(continued from last page)

getExtParams

```
public java.lang.String getExtParams()
```

getAlias

```
public java.lang.String getAlias()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL proedure that performs the extraction.

Parameters:

`ExtractPattern` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class InsertIntoClass

```
java.lang.Object
   |
   +-com.psycholight.statements.PBMSStatement
      |
      +-com.psycholight.statements.PDL_PMLStatement
         |
         +-com.psycholight.statements.InsertIntoClass
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **InsertIntoClass**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	InsertIntoClass(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateClass
public	InsertIntoClass(java.lang.String ClassName, java.lang.String Condition)

Method Summary

java.lang.String	getAlias()
java.lang.String	getClassName()
java.lang.String	getCondition()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL procedure that performs the insertion.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

InsertIntoClass

```
public InsertIntoClass(java.sql.Connection conn,
                       java.lang.String cmd)
```

Creates a new instance of CreateClass

InsertIntoClass

```
public InsertIntoClass(java.lang.String ClassName,
                       java.lang.String Condition)
```

Methods

getClassName

```
public java.lang.String getClassName()
```

getCondition

```
public java.lang.String getCondition()
```

getAlias

```
public java.lang.String getAlias()
```

(continued from last page)

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL procedure that performs the insertion.

Parameters:

`InsertIntoClass` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class ParserUtilities

```
java.lang.Object
   |
   +-com.psycholight.statements.ParserUtilities
```

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

Title: ParserUtilities.java

Description: Contains static utilities methods used for parsing input streams

Constructor Summary

public	ParserUtilities()

Method Summary

static void	consumeToken(java.io.StreamTokenizer st, java.lang.String token) Consumes a given token of the input stream.
static void	consumeToken(java.io.StreamTokenizer st, java.lang.String[] token) Consumes tokens of the input stream.
static PDL_PMLStatement	getPdlStatementTypeFrom(java.sql.Connection conn, java.lang.String cmd)
static PDL_PMLStatement	getPmlStatementTypeFrom(java.sql.Connection conn, java.lang.String cmd)
static PQLStatement	getPqlStatementTypeFrom(java.sql.Connection conn, java.lang.String cmd)
static PBMSStatement	getStatementTypeFrom(java.sql.Connection conn, java.lang.String cmd)
static java.lang.String	nextStringToken(java.io.StreamTokenizer st) retreive the next string token in the stream.
static java.lang.String	nextToken(java.io.StreamTokenizer st) Retreive the next token in the stream.
static TypedField	parseFieldDeclaration(java.io.StreamTokenizer st, java.lang.String delim) Parse a field declaration part of a command from the input stream starting with a given keyword.
static TypedField	parseFieldDeclaration(java.io.StreamTokenizer st, java.lang.String[] delim) Parse a field declaration part of a command from the input stream bounded by the given keywords.
static java.lang.String	parseWhere(java.io.StreamTokenizer st, java.lang.String[] delim, java.lang.String alias, java.lang.String patternTypeName)

Methods inherited from class `java.lang.Object`
`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

ParserUtilities

```
public ParserUtilities()
```

Methods

nextStringToken

```
public static java.lang.String nextStringToken(java.io.StreamTokenizer st)
```

retreive the next string token in the stream.

Parameters:
`StreamTokenizer` - The stream to parse.

Returns:
The next token

nextToken

```
public static java.lang.String nextToken(java.io.StreamTokenizer st)
```

Retreive the next token in the stream.

Parameters:
`StreamTokenizer` - The stream to parse.

Returns:
The next token

consumeToken

```
public static void consumeToken(java.io.StreamTokenizer st,
                                java.lang.String token)
  throws com.jpbc.PException
```

Consumes a given token of the input stream.

Parameters:
`StreamTokenizer` - The stream to parse.
`String` - The starting keyword

Throws:
`WrongKeywordException` - If the current token in the stream is differnt from the given keyword

(continued from last page)

parseFieldDeclaration

```
public static TypedField parseFieldDeclaration(java.io.StreamTokenizer st,
         java.lang.String[] delim)
```

Parse a field declaration part of a command from the input stream bounded by the given keywords.

Parameters:

`StreamTokenizer` - The stream to parse.
`String` - The starting keyword

Returns:

The parsed field.

---

parseFieldDeclaration

```
public static TypedField parseFieldDeclaration(java.io.StreamTokenizer st,
         java.lang.String delim)
```

Parse a field declaration part of a command from the input stream starting with a given keyword.

Parameters:

`StreamTokenizer` - The stream to parse.
`String` - The starting keyword

Returns:

The parsed field.

---

parseWhere

```
public static java.lang.String parseWhere(java.io.StreamTokenizer st,
         java.lang.String[] delim,
         java.lang.String alias,
         java.lang.String patternTypeName)
```

---

consumeToken

```
public static void consumeToken(java.io.StreamTokenizer st,
         java.lang.String[] token)
  throws com.jpbc.PException
```

Consumes tokens of the input stream.

Parameters:

`StreamTokenizer` - The stream to parse.
`String` - The kewords to consume

Throws:

`WrongKeywordException` - If the current token in the stream is differnt from the given keywords

---

getStatementTypeFrom

```
public static PBMSStatement getStatementTypeFrom(java.sql.Connection conn,
         java.lang.String cmd)
  throws com.jpbc.PException
```

getPdlStatementTypeFrom

```
public static PDL_PMLStatement getPdlStatementTypeFrom(java.sql.Connection conn,
         java.lang.String cmd)
  throws com.jpbc.PException
```

getPmlStatementTypeFrom

```
public static PDL_PMLStatement getPmlStatementTypeFrom(java.sql.Connection conn,
         java.lang.String cmd)
  throws com.jpbc.PException
```

getPqlStatementTypeFrom

```
public static PQLStatement getPqlStatementTypeFrom(java.sql.Connection conn,
         java.lang.String cmd)
  throws com.jpbc.PException
```

com.psycholight.statements
Class PatternCovering

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PQLStatement
         |-com.psycholight.statements.PatternCovering
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PqlStatement

---

public class **PatternCovering**
extends PQLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	PatternCovering(java.sql.Connection conn, java.lang.String cmd)

Method Summary

java.lang.String	getClassName()
java.lang.String	getCondition()
java.util.Vector	getCondParams()
java.lang.String	getDatasource()
Select	getInnerSelect()
java.lang.String	getStoreAs()
com.jpbc.PResultSet	interpretStatement() It is translated in a call to a PL/SQL proedure that performs the query.
boolean	isSelect()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PQLStatement
executePQLcommand, executePQLcommand, getStatementType, interpretStatement

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PqlStatement
executePQLcommand, executePQLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

PatternCovering

```
public PatternCovering(java.sql.Connection conn,
                       java.lang.String cmd)
```

Methods

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public com.jpbc.PResultSet interpretStatement()
  throws com.jpbc.PException
```

It is translated in a call to a PL/SQL proedure that performs the query.

Parameters:

`PatternCovering` - The statement to translate

Returns:

The result of the query. Null in case of errors.

getClassName

```
public java.lang.String getClassName()
```

getCondition

```
public java.lang.String getCondition()
```

getDatasource

```
public java.lang.String getDatasource()
```

getStoreAs

```
public java.lang.String getStoreAs()
```

isSelect

```
public boolean isSelect()
```

getInnerSelect

```
public Select getInnerSelect()
```

getCondParams

```
public java.util.Vector getCondParams()
```

com.psycholight.statements
Class PBMSStatement

```
java.lang.Object
   |
   +-com.psycholight.statements.PBMSStatement
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement

Direct Known Subclasses:
ExecSQL, PDL_PMLStatement, PQLStatement

---

public abstract class **PBMSStatement**
extends java.lang.Object
implements com.jpbc.PBStatement, java.io.Serializable

The superclass of all PSYCHOfree statements

Field Summary

public static	OTHER_STATEMENT
public static	PDL_OR_PML_STATEMENT
public static	QUERY_STATEMENT

Constructor Summary

public	PBMSStatement()
public	PBMSStatement(java.sql.Connection conn, java.lang.String cmd)

Method Summary

void	close()
java.lang.Object	execute()
java.lang.Object	execute(java.lang.String str)
abstract int	getStatementType()
abstract java.lang.Object	interpretStatement()
abstract void	parseStatement() This method parse a PBMSStatement from input

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Fields

PDL_OR_PML_STATEMENT

```
public static int PDL_OR_PML_STATEMENT
```

QUERY_STATEMENT

```
public static int QUERY_STATEMENT
```

OTHER_STATEMENT

```
public static int OTHER_STATEMENT
```

Constructors

PBMSStatement

```
public PBMSStatement()
```

PBMSStatement

```
public PBMSStatement(java.sql.Connection conn,
                     java.lang.String cmd)
```

Methods

parseStatement

```
public abstract void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

Throws:

PException - If this statement's syntax is not correct

(continued from last page)

interpretStatement

```
public abstract java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

---

getStatementType

```
public abstract int getStatementType()
```

---

close

```
public void close()
```

---

execute

```
public java.lang.Object execute()
  throws com.jpbc.PException
```

---

execute

```
public java.lang.Object execute(java.lang.String str)
  throws com.jpbc.PException
```

com.psycholight.statements
Class PDL_PMLStatement

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
```

All Implemented Interfaces:
com.jpbc.PmlStatement, com.jpbc.PdlStatement, java.io.Serializable, com.jpbc.PBStatement

Direct Known Subclasses:
CreateClass, DeleteFromClass, DeletePatterns, DirectInsertPattern, DropClass, DropMeasureFunction, DropPatternType, ExtractPattern, InsertIntoClass, RecomputePattern, StatementWithCode, Update, UpdatePattern

---

public abstract class **PDL_PMLStatement**
extends PBMSStatement
implements com.jpbc.PBStatement, java.io.Serializable, com.jpbc.PdlStatement, com.jpbc.PmlStatement

General superclass for all PDL and PML statement whose execution will return a message (generally success or failure)

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	PDL_PMLStatement()
public	PDL_PMLStatement(java.sql.Connection conn, java.lang.String cmd)

Method Summary

java.lang.String	executePDLcommand()
java.lang.String	executePDLcommand(java.lang.String str)
int	executePMLcommand()
int	executePMLcommand(java.lang.String str)
int	getStatementType()

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

PDL_PMLStatement

```
public PDL_PMLStatement()
```

PDL_PMLStatement

```
public PDL_PMLStatement(java.sql.Connection conn,
                        java.lang.String cmd)
```

Methods

getStatementType

```
public int getStatementType()
```

executePDLcommand

```
public java.lang.String executePDLcommand()
  throws com.jpbc.PException
```

executePDLcommand

```
public java.lang.String executePDLcommand(java.lang.String str)
  throws com.jpbc.PException
```

executePMLcommand

```
public int executePMLcommand()
  throws com.jpbc.PException
```

executePMLcommand

```
public int executePMLcommand(java.lang.String str)
  throws com.jpbc.PException
```

com.psycholight.statements
Class PQLStatement

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PQLStatement
```

All Implemented Interfaces:
com.jpbc.PqlStatement, java.io.Serializable, com.jpbc.PBStatement

Direct Known Subclasses:
DataCovering, DrillThrough, PatternCovering, Select, Show

---

public abstract class **PQLStatement**
extends PBMSStatement
implements com.jpbc.PBStatement, java.io.Serializable, com.jpbc.PqlStatement

Title: PQLStatement

Description: General superclass for all querying statement whose execution will return a ResultSet object

---

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	PQLStatement()
public	PQLStatement(java.sql.Connection conn, java.lang.String cmd)

Method Summary

com.jpbc.PResultSet	executePQLcommand()
com.jpbc.PResultSet	executePQLcommand(java.lang.String str)
int	getStatementType()
abstract com.jpbc.PResultSet	interpretStatement()

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PqlStatement
executePQLcommand, executePQLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

PQLStatement

```
public PQLStatement()
```

PQLStatement

```
public PQLStatement(java.sql.Connection conn,
                    java.lang.String cmd)
```

Methods

getStatementType

```
public int getStatementType()
```

interpretStatement

```
public abstract com.jpbc.PResultSet interpretStatement()
  throws com.jpbc.PException
```

executePQLcommand

```
public com.jpbc.PResultSet executePQLcommand()
  throws com.jpbc.PException
```

executePQLcommand

```
public com.jpbc.PResultSet executePQLcommand(java.lang.String str)
  throws com.jpbc.PException
```

com.psycholight.statements
Class RecomputePattern

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.RecomputePattern
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **RecomputePattern**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	RecomputePattern(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of ExtractPatternType

Method Summary

java.lang.String	getAlias()
java.lang.String	getCondition()
java.lang.String	getDataSourceName()
java.lang.String	getIntoClass()
java.lang.String	getMeasureFunctionName()
java.lang.String	getPatternTypeName()
java.lang.String	getTsFinal()
java.lang.String	getTsInit()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL proedure that performs the operation.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement

executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement

executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement

executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Constructors

RecomputePattern

```
public RecomputePattern(java.sql.Connection conn,
                        java.lang.String cmd)
```

Creates a new instance of ExtractPatternType

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getDataSourceName

```
public java.lang.String getDataSourceName()
```

(continued from last page)

getCondition

```
public java.lang.String getCondition()
```

getTsInit

```
public java.lang.String getTsInit()
```

getTsFinal

```
public java.lang.String getTsFinal()
```

getAlias

```
public java.lang.String getAlias()
```

getMeasureFunctionName

```
public java.lang.String getMeasureFunctionName()
```

getIntoClass

```
public java.lang.String getIntoClass()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL proedure that performs the operation.

Parameters:

`RecomputePattern` - The statement to translate

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class Select

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PQLStatement
         |-com.psycholight.statements.Select
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PqlStatement

---

public class **Select**
extends PQLStatement

Field Summary

public static final	CJOIN Value: **3**
public static final	INTERSECT Value: **2**
public static final	NATURAL Value: **0**
public static final	UNION Value: **1**

Fields inherited from class com.psycholight.statements.PBMSStatement

OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	Select(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of CreateClass
public	Select(java.sql.Connection conn)

Method Summary

java.util.ArrayList	getClassLists()
java.lang.String	getClassName()
java.lang.String	getCompFuntionName()

java.lang.String	getCondition()
java.util.ArrayList	getFields()
java.lang.String	getFirstAlias()
java.lang.String	getJoinedAlias()
java.lang.String	getJoinedClass()
int	getJoinType()
boolean	getOpType()
java.lang.String	getTableName()
java.lang.String	getTypeName()
com.jpbc.PResultSet	interpretStatement() The selection operation is translated in a call to a PL/SQL proedure that performs the query.
boolean	isJoin()
void	parseStatement()
void	parseStatement(java.io.StreamTokenizer st)

Methods inherited from class com.psycholight.statements.PQLStatement

executePQLcommand, executePQLcommand, getStatementType, interpretStatement

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

Methods inherited from interface com.jpbc.PqlStatement

executePQLcommand, executePQLcommand

Methods inherited from interface com.jpbc.PBStatement

close, execute, execute

(continued from last page)

Fields

NATURAL

```
public static final int NATURAL
```

Constant value: **0**

UNION

```
public static final int UNION
```

Constant value: **1**

INTERSECT

```
public static final int INTERSECT
```

Constant value: **2**

CJOIN

```
public static final int CJOIN
```

Constant value: **3**

Constructors

Select

```
public Select(java.sql.Connection conn,
              java.lang.String cmd)
```

Creates a new instance of CreateClass

Select

```
public Select(java.sql.Connection conn)
```

Methods

getFields

```
public java.util.ArrayList getFields()
```

getClassLists

```
public java.util.ArrayList getClassLists()
```

getCondition

public java.lang.String **getCondition**()

getTableName

public java.lang.String **getTableName**()

getClassName

public java.lang.String **getClassName**()

getJoinedClass

public java.lang.String **getJoinedClass**()

isJoin

public boolean **isJoin**()

getTypeName

public java.lang.String **getTypeName**()

getJoinType

public int **getJoinType**()

getCompFuntionName

public java.lang.String **getCompFuntionName**()

getOpType

public boolean **getOpType**()

(continued from last page)

parseStatement

```
public void parseStatement(java.io.StreamTokenizer st)
  throws com.jpbc.PException
```

---

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

---

getJoinedAlias

```
public java.lang.String getJoinedAlias()
```

---

getFirstAlias

```
public java.lang.String getFirstAlias()
```

---

interpretStatement

```
public com.jpbc.PResultSet interpretStatement()
  throws com.jpbc.PException
```

The selection operation is translated in a call to a PL/SQL proedure that performs the query.

Parameters:

`Select` - The statement to translate

Returns:

The result of the query. Null in case of errors.

com.psycholight.statements
Class Show

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PQLStatement
         |-com.psycholight.statements.Show
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PqlStatement

---

public class **Show**
extends PQLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	Show(java.sql.Connection conn, java.lang.String cmd) Creates a new instance of Show

Method Summary

com.jpbc.PResultSet	interpretStatement()
boolean	isIsClasses()
boolean	isIsPatternTypes()
void	parseStatement()

Methods inherited from class com.psycholight.statements.PQLStatement
executePQLcommand, executePQLcommand, getStatementType, interpretStatement

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PqlStatement
executePQLcommand, executePQLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

Show

```
public Show(java.sql.Connection conn,
            java.lang.String cmd)
```

Creates a new instance of Show

Methods

isIsClasses

```
public boolean isIsClasses()
```

isIsPatternTypes

```
public boolean isIsPatternTypes()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public com.jpbc.PResultSet interpretStatement()
  throws com.jpbc.PException
```

com.psycholight.statements
Class StatementWithCode

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.StatementWithCode
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

Direct Known Subclasses:
CreateCompositionFunction, CreateCondition, CreateFormula, CreateJoinCondition, CreateMeasureFunction, CreateMiningFunction, CreatePatternType

---

public abstract class **StatementWithCode**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	StatementWithCode(java.sql.Connection conn, java.lang.String cmd)

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

StatementWithCode

```
public StatementWithCode(java.sql.Connection conn,
                         java.lang.String cmd)
```

com.psycholight.statements
Class Update

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.Update
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **Update**
extends PDL_PMLStatement

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	Update(java.sql.Connection conn, java.lang.String cmd)

Method Summary

java.lang.String	getAlias()
java.lang.String	getPatternTypeName()
java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL proedure that performs the update of the validity period.
void	parseStatement()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

Update

```
public Update(java.sql.Connection conn,
              java.lang.String cmd)
```

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getAlias

```
public java.lang.String getAlias()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL proedure that performs the update of the validity period.

Parameters:

`DeletePatterns` - The statement to translate

(continued from last page)

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class UpdatePattern

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.UpdatePattern
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

Direct Known Subclasses:
UpdatePatternWithMeasureFunction, UpdateValidity

---

public abstract class **UpdatePattern**
extends PDL_PMLStatement

Title: UpdatePattern.java

Description: Updating patterns may follow three distinct ways: SYNCHRONIZE patterns, VALIDATE patterns, SET patterns VALIDITY; this is the superclass of these three different statements. An object belonging this class is owned by Update statement which should instanciate it correctly reading what user typed in input.

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	UpdatePattern(java.sql.Connection conn, java.lang.String patternTypeName, java.lang.String alias)

Method Summary

java.lang.String	getAlias()
java.lang.String	getCondition()
java.lang.String	getPatternTypeName()

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

UpdatePattern

```
public UpdatePattern(java.sql.Connection conn,
                     java.lang.String patternTypeName,
                     java.lang.String alias)
```

Methods

getPatternTypeName

```
public java.lang.String getPatternTypeName()
```

getAlias

```
public java.lang.String getAlias()
```

getCondition

```
public java.lang.String getCondition()
```

com.psycholight.statements
Class UpdatePatternWithMeasureFunction

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.UpdatePattern
            |-com.psycholight.statements.UpdatePatternWithMeasureFunction
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

Direct Known Subclasses:
UpdateSynchronize, UpdateValidate

---

public abstract class **UpdatePatternWithMeasureFunction**
extends UpdatePattern

Title: UpdatePatternWithMeasureFunction.java

Description: This is the superclass of UpdateValidate and UpdateValidity as they both refers to a measure function which is commonly handled here

---

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	UpdatePatternWithMeasureFunction(java.sql.Connection conn, java.lang.String patternTypeName, java.lang.String alias)

Method Summary

java.lang.String	getFunctionName()

Methods inherited from class com.psycholight.statements.UpdatePattern
getAlias, getCondition, getPatternTypeName

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object

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

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

UpdatePatternWithMeasureFunction

```
public UpdatePatternWithMeasureFunction(java.sql.Connection conn,
                                        java.lang.String patternTypeName,
                                        java.lang.String alias)
```

Methods

getFunctionName

```
public java.lang.String getFunctionName()
```

com.psycholight.statements
Class UpdateSynchronize

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.UpdatePattern
            |-com.psycholight.statements.UpdatePatternWithMeasureFunction
               |-com.psycholight.statements.UpdateSynchronize
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **UpdateSynchronize**
extends UpdatePatternWithMeasureFunction

Title: UpdateSynchronize.java

Description: This class wraps and Update statement dealing with synchronization of patterns

---

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	UpdateSynchronize(java.sql.Connection conn, java.lang.String patternTypeName, java.lang.String alias, java.io.StreamTokenizer st)

Method Summary

java.lang.Object	interpretStatement() This is translated in a call to a PL/SQL procedure that performs patterns sync.
void	parseStatement()

Methods inherited from class com.psycholight.statements.UpdatePatternWithMeasureFunction
getFunctionName

Methods inherited from class com.psycholight.statements.UpdatePattern
getAlias, getCondition, getPatternTypeName

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

UpdateSynchronize

```
public UpdateSynchronize(java.sql.Connection conn,
                         java.lang.String patternTypeName,
                         java.lang.String alias,
                         java.io.StreamTokenizer st)
```

Methods

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

This is translated in a call to a PL/SQL procedure that performs patterns sync.

Returns:

A message containing the outcome of the execution.

com.psycholight.statements
Class UpdateValidate

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.UpdatePattern
            |-com.psycholight.statements.UpdatePatternWithMeasureFunction
               |-com.psycholight.statements.UpdateValidate
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **UpdateValidate**
extends UpdatePatternWithMeasureFunction

Title: UpdateValidate.java

Description: This class wraps and Update statement dealing with validation of patterns

---

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	UpdateValidate(java.sql.Connection conn, java.lang.String patternTypeName, java.lang.String alias, java.io.StreamTokenizer st)

Method Summary

java.lang.String	getIntoClass()
java.lang.Object	interpretStatement()
void	parseStatement()

Methods inherited from class com.psycholight.statements.UpdatePatternWithMeasureFunction
getFunctionName

Methods inherited from class com.psycholight.statements.UpdatePattern
getAlias, getCondition, getPatternTypeName

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement
close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

UpdateValidate

```
public UpdateValidate(java.sql.Connection conn,
                      java.lang.String patternTypeName,
                      java.lang.String alias,
                      java.io.StreamTokenizer st)
```

Methods

getIntoClass

```
public java.lang.String getIntoClass()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

This method parse a PBMSStatement from input

(continued from last page)

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```

com.psycholight.statements
Class UpdateValidity

```
java.lang.Object
   |-com.psycholight.statements.PBMSStatement
      |-com.psycholight.statements.PDL_PMLStatement
         |-com.psycholight.statements.UpdatePattern
            |-com.psycholight.statements.UpdateValidity
```

All Implemented Interfaces:
java.io.Serializable, com.jpbc.PBStatement, com.jpbc.PmlStatement, com.jpbc.PdlStatement

---

public class **UpdateValidity**
extends UpdatePattern

Title: UpdateValidity.java

Description: This class wraps an Update statement dealing with pattern validity period

---

Fields inherited from class com.psycholight.statements.PBMSStatement
OTHER_STATEMENT, PDL_OR_PML_STATEMENT, QUERY_STATEMENT

Constructor Summary

public	UpdateValidity(java.sql.Connection conn, java.lang.String patternTypeName, java.lang.String alias, java.io.StreamTokenizer st)

Method Summary

java.lang.String	getTsFinal()
java.lang.String	getTsInit()
java.lang.Object	interpretStatement()
void	parseStatement()

Methods inherited from class com.psycholight.statements.UpdatePattern
getAlias, getCondition, getPatternTypeName

Methods inherited from class com.psycholight.statements.PDL_PMLStatement
executePDLcommand, executePDLcommand, executePMLcommand, executePMLcommand, getStatementType

Methods inherited from class com.psycholight.statements.PBMSStatement

close, execute, execute, getStatementType, interpretStatement, parseStatement

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PdlStatement
executePDLcommand, executePDLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Methods inherited from interface com.jpbc.PmlStatement
executePMLcommand, executePMLcommand

Methods inherited from interface com.jpbc.PBStatement
close, execute, execute

Constructors

UpdateValidity

```
public UpdateValidity(java.sql.Connection conn,
                      java.lang.String patternTypeName,
                      java.lang.String alias,
                      java.io.StreamTokenizer st)
```

Methods

getTsInit

```
public java.lang.String getTsInit()
```

getTsFinal

```
public java.lang.String getTsFinal()
```

parseStatement

```
public void parseStatement()
  throws com.jpbc.PException
```

(continued from last page)

This method parse a PBMSStatement from input

interpretStatement

```
public java.lang.Object interpretStatement()
  throws com.jpbc.PException
```