My Project

Generated by Doxygen 1.9.3

1 Class Documentation

1.1 AnswerCallback

Additional Inherited Members

1.1.1 Detailed Description

Answer callback - an answer callback.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.2 AppSocket

Public Member Functions

- **AppSocket** (String ip, int port) throws IOException
- **AppSocket** (Socket socket) throws IOException
- void **close** ()
- void **close** (**MyError** err)
- void **interruptListener** (**MyError** err)
- void **requestMessage** (**Message** requestMsg, **MessageCallback** onRes)
- **MessagesHandler** **getMessagesHandler** ()
- void **setMessagesHandler** (**MessagesHandler** messagesHandler)
- void **respond** (**Message** msg, **Message** respondingTo)
- synchronized void **writeMessage** (**Message** msg)
- boolean **isConnected** ()
- String **getLocalAddress** ()
- String **getRemoteAddress** ()
- boolean **isClosed** ()
- **Message** **requestMessage** (**Message** requestMsg)
- void **stopReading** ()

Protected Member Functions

- void **handledRun** ()

Protected Attributes

- final Socket **msgSocket**

Additional Inherited Members

1.2.1 Detailed Description

App socket - represents a communications socket able to send and receive messages from the client to the server and vice versa.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.2.2 Constructor & Destructor Documentation

1.2.2.1 AppSocket() [1/2] AppSocket (

```
String ip,
int port ) throws IOException
```

Instantiates a new App socket.

Parameters

ip	the ip
port	the port

Exceptions

IOException	the io exception

1.2.2.2 AppSocket() [2/2]

```
AppSocket (
        Socket socket ) throws IOException
```

Instantiates a new App socket.

Parameters

socket	the socket

Exceptions

IOException	the io exception

1.2.3 Member Function Documentation

1.2.3.1 close() [1/2]

```
void close ( )
```

Close.

1.2.3.2 close() [2/2]

```
void close (
        MyError err )
```

Close.

Parameters

err	the error

1.2.3.3 getLocalAddress()

```
String getLocalAddress ( )
```

Gets local address.

Returns

the local address

1.2.3.4 getMessagesHandler() `MessagesHandler getMessagesHandler ( )`

Gets messages handler.

Returns

the messages handler

1.2.3.5 getRemoteAddress() `String getRemoteAddress ( )`

Gets remote address.

Returns

the remote address

1.2.3.6 handledRun() `void handledRun ( ) [protected]`

Handled run.

Reimplemented from **MyThread** (p. 169).

1.2.3.7 interruptListener() `void interruptListener ( MyError err )`

Interrupt listener.

Parameters

err	the err to interrupt with

1.2.3.8 isClosed() `boolean isClosed ( )`

Is closed boolean.

Returns

the boolean

1.2.3.9 isConnected() `boolean isConnected ( )`

Is connected boolean.

Returns

the boolean

1.2.3.10 requestMessage() [1/2]

```
Message requestMessage (
        Message requestMsg )
```

sending request and blocking til res

Parameters

requestMsg	= "can i have x message?"

Returns

response

1.2.3.11 requestMessage() [2/2]

```
void requestMessage (
        Message requestMsg,
        MessageCallback onRes )
```

Request message.

Parameters

requestMsg	the request msg
onRes	the on res

1.2.3.12 respond()

```
void respond (
        Message msg,
        Message respondingTo )
```

Respond.

Parameters

msg	the msg
respondingTo	the responding to

1.2.3.13 setMessagesHandler() `void setMessagesHandler ( MessagesHandler messagesHandler )`

Sets messages handler.

Parameters

messagesHandler	the messages handler

1.2.3.14 stopReading() `void stopReading ( )`

Stop reading.

1.2.3.15 writeMessage() `synchronized void writeMessage ( Message msg )`

Write message.

Parameters

msg	the msg

1.2.4 Member Data Documentation

1.2.4.1 msgSocket `final Socket msgSocket [protected]`

The Msg socket.

1.3 ArchivedGameInfo

Public Member Functions

- **ArchivedGameInfo** (String **gameId**, String **creatorUsername**, String **opponentUsername**, GameSettings **gameSettings**, String winner, Stack< **Move** > moveStack)
- String **getWinner** ()
- String **toString** ()

Additional Inherited Members

1.3.1 Detailed Description

Archived game info.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.3.2 Constructor & Destructor Documentation

1.3.2.1 ArchivedGameInfo() **ArchivedGameInfo** (

```
String gameId,
String creatorUsername,
String opponentUsername,
GameSettings gameSettings,
String winner,
Stack< Move > moveStack )
```

Instantiates a new Archived game info.

Parameters

gameId	the game id
creatorUsername	the creator username
opponentUsername	the opponent username
gameSettings	the game settings
winner	the winner
moveStack	the move stack

1.3.3 Member Function Documentation

1.3.3.1 getWinner() `String getWinner ( )`

Gets winner.

Returns

the winner

1.3.3.2 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented from **EstablishedGameInfo** (p. 73).

1.4 Arg

Public Member Functions

- **Arg** (**ArgType argType**)
- **Arg** (**ArgType argType**, **Config**<?> **config**)
- **Arg** (**ArgType argType**, boolean **escape**, **Config**<?> **config**)
- boolean **isUserInput** ()
- void **setUserInput** (boolean userInput)
- boolean **equals** (Object o)
- String **toString** ()
- String **createVal** (Object val)

Public Attributes

- final String **replnStr**
- final boolean **escape**
- final **ArgType argType**
- final **Config**<?> **config**

1.4.1 Detailed Description

Arg (p. 8) - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.4.2 Constructor & Destructor Documentation

1.4.2.1 Arg() [1/3]

```
Arg (
    ArgType argType )
```

Instantiates a new **Arg** (p. 8).

Parameters

argType	the arg type

1.4.2.2 Arg() [2/3]

```
Arg (
    ArgType argType,
    Config<?> config )
```

Instantiates a new **Arg** (p. 8).

Parameters

argType	the arg type
config	the config

1.4.2.3 Arg() [3/3]

```
Arg (
    ArgType argType,
    boolean escape,
    Config<?> config )
```

Instantiates a new **Arg** (p. 8).

Parameters

argType	the arg type
escape	the escape
config	the config

1.4.3 Member Function Documentation

1.4.3.1 createVal()

```
String createVal (
    Object val )
```

Create val string.

Parameters

val	the val

Returns

the string

1.4.3.2 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

1.4.3.3 isUserInput() `boolean isUserInput ( )`

Is user input boolean.

Returns

the boolean

1.4.3.4 setUserInput() `void setUserInput ( boolean userInput )`

Sets user input.

Parameters

userInput	the user input

1.4.3.5 toString() `String toString ( )`

To string string.

Returns

the string

1.4.4 Member Data Documentation

1.4.4.1 argType `final ArgType argType`

The **Arg** (p. 8) type.

1.4.4.2 config `final Config<?> config`

The **Config** (p. 49).

1.4.4.3 escape `final boolean escape`

The Escape.

1.4.4.4 repInStr `final String repInStr`

The Rep in str.

1.5 ArgsUtil

Public Member Functions

- **OptionalArg equalsSign** (String preEqualStr)
- **OptionalArg plainTextIgnoreCase** (String str)
- record **OptionalArg** (String str)

Static Public Member Functions

- static **ArgsUtil create** (String[ ] args)

1.5.1 Detailed Description

The type Args utils.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.5.2 Member Function Documentation

1.5.2.1 create() `static ArgsUtil create ( String[ ] args ) [static]`

Create args util.

Parameters

args	the args

Returns

the args util

1.5.2.2 equalsSign() `OptionalArg equalsSign ( String preEqualStr )`

Equals sign optional arg. for any arg of this format: preEqualStr=argval%

Parameters

preEqualStr	the pre equal str

Returns

the optional arg value(assuming there is one) argval% in the example above

1.5.2.3 OptionalArg() `record OptionalArg ( String str )`

Optional arg.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.5.2.4 plainTextIgnoreCase() `OptionalArg plainTextIgnoreCase ( String str )`

Plain text ignore case optional arg.

Parameters

str	the str

Returns

the optional arg

1.6 ArgType Enum Reference

Public Member Functions

- **ArgType** ()
- **ArgType** (boolean **isUserInput**)

Public Attributes

- **Date**
- **DateRange**
- **Text**
- **ServerAddress**
- **Number**
- **Username**
- **Password**
- **Url**
- **PictureUrl**
- final boolean **isUserInput**

1.6.1 Detailed Description

Arg (p. 8) type - argument type.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.6.2 Constructor & Destructor Documentation

1.6.2.1 ArgType() `[1/2]` **`ArgType`** `( )`

Instantiates a new **Arg** (p. 8) type.

1.6.2.2 ArgType() `[2/2]` **`ArgType`** `(`
`boolean` *`isUserInput`* `)`

Instantiates a new **Arg** (p. 8) type.

Parameters

isUserInput	the is user input

1.6.3 Member Data Documentation

1.6.3.1 Date `Date`

Date arg type.

1.6.3.2 DateRange `DateRange`

Date range arg type.

1.6.3.3 isUserInput `final boolean isUserInput`

The Is user input.

1.6.3.4 Number `Number`

Number arg type.

1.6.3.5 Password `Password`

Password arg type.

1.6.3.6 PictureUrl `PictureUrl`

Picture url arg type.

1.6.3.7 ServerAddress `ServerAddress`

Server address arg type.

1.6.3.8 Text `Text`

Text arg type.

1.6.3.9 Url `Url`

Url arg type.

1.6.3.10 Username `Username`

Username arg type.

1.7 ArrUtils

Static Public Member Functions

- static< T > T[ ] **concat** (T[ ] array1, T... array2)
- static< T > ArrayList< T > **createList** (**ObjCallback**< T > objCreator, int size)
- static< T > T **exists** (T[ ] arr, int... index)

1.7.1 Detailed Description

Array Utility Class.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.7.2 Member Function Documentation

1.7.2.1 concat()

```
static< T > T[ ] concat (
        T[ ] array1,
        T...  array2 )  [static]
```

`credit`

Parameters

<T>	the type parameter
array1	the array 1
array2	the array 2

Returns

t [ ]

1.7.2.2 createList()

```
static< T > ArrayList< T > createList (
         ObjCallback< T > objCreator,
        int size )  [static]
```

Create list array list.

Parameters

<T>	the type parameter
objCreator	the obj creator
size	the size

Returns

the array list

1.7.2.3 exists()

```
static< T > T exists (
        T[] arr,
        int... index ) [static]
```

Exists t.

Parameters

<T>	the type parameter
arr	the arr
index	the index

Returns

the t

1.8 BasicMove

Public Member Functions

- **BasicMove** (**BasicMove** other)
- **BasicMove** (**Location** movingFrom, **Location** movingTo)
- **BasicMove** (String move)
- void **flip** ()
- **Location getMovingFrom** ()
- void **setMovingFrom** (**Location** movingFrom)
- **Location getMovingTo** ()
- void **setMovingTo** (**Location** movingTo)
- int **hashCode** ()
- boolean **equals** (Object o)
- String **toString** ()
- String **getBasicMoveAnnotation** ()
- **BasicMove cp** ()

Static Public Member Functions

- static **BasicMove**[ ] **createBatch** (Location... locs)
- static **BasicMove getFlipped** (**BasicMove** basicMove)

1.8.1 Detailed Description

Basic move - represents a basic move. with a source and a destination.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.8.2 Constructor & Destructor Documentation

1.8.2.1 BasicMove() [1/3] BasicMove (

```
BasicMove other )
```

Copy constructor.

Parameters

other	the other

1.8.2.2 BasicMove() [2/3] BasicMove (

```
Location movingFrom,
Location movingTo )
```

Instantiates a new Basic move.

Parameters

movingFrom	the moving from
movingTo	the moving to

1.8.2.3 BasicMove() [3/3] BasicMove (

```
String move )
```

Instantiates a new Basic move.

Parameters

move	the move

1.8.3 Member Function Documentation

1.8.3.1 cp() BasicMove cp ( )

copies this move.

Returns

the new copy

1.8.3.2 createBatch() `static BasicMove[] createBatch ( Location... locs ) [static]`

Create batch basic move [ ].

Parameters

locs	the locs

Returns

the basic move [ ]

1.8.3.3 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

Reimplemented in **Move** (p. 150).

1.8.3.4 flip() `void flip ( )`

Flips the source and destination.

1.8.3.5 getBasicMoveAnnotation() `String getBasicMoveAnnotation ( )`

Gets basic move annotation.

Returns

the basic move annotation

1.8.3.6 getFlipped() `static BasicMove getFlipped ( BasicMove basicMove ) [static]`

Gets a copy of the provided move with the source and destination flipped

Parameters

basicMove	the original move

Returns

the flipped move

1.8.3.7 getMovingFrom() `Location getMovingFrom ( )`

Gets moving from. aka source

Returns

the moving from

1.8.3.8 getMovingTo() `Location getMovingTo ( )`

Gets moving to.

Returns

the moving to

1.8.3.9 hashCode() `int hashCode ( )`

Hash code int.

Returns

the int

1.8.3.10 setMovingFrom() `void setMovingFrom ( Location movingFrom )`

Sets moving from.

Parameters

movingFrom	the moving from

1.8.3.11 setMovingTo() `void setMovingTo ( Location movingTo )`

Sets moving to.

Parameters

movingTo	the moving to

1.8.3.12 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented in **Move** (p. 157).

1.9 BitData

Static Public Attributes

- static final long **notAFile**
- static final long **notHFile**
- static final long **everything**

1.9.1 Detailed Description

Bit data - utility class meant for storing useful board constants. (in bitboard format of course)

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.9.2 Member Data Documentation

1.9.2.1 everything `final long everything [static]`

The constant everything. the whole board

1.9.2.2 notAFile `final long notAFile [static]`

The constant notAFile. the whole board but the A file

1.9.2.3 notHFile `final long notHFile [static]`

The constant notHFile. the whole board but the H file

1.10 Board

Public Member Functions

- **Board** (**Board** other)
- **Board** ()
- void **setPiece** (**Location** loc, **Piece** piece)
- **Square** **getSquare** (**Location** loc)
- **Board** (String fen)
- void **fenSetup** (String fen)
- **Square**[ ] **getRow** (int row, boolean flipLocs)
- **Piece** **getPiece** (**Location** loc, boolean notNull)
- **Piece** **getPiece** (**Location** loc)
- void **print** ()
- Iterator< **Square** > **iterator** ()
- void **setSquareEmpty** (**Location** loc)
- boolean **isSquareEmpty** (**Location** loc)
- String **toString** ()

Static Public Member Functions

- static **Board** **startingPos** ()

Static Public Attributes

- static final String **startingFen**
- static final **Board** **example**

1.10.1 Detailed Description

Board (p. 21) - represents the logic board of the model

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.10.2 Constructor & Destructor Documentation

1.10.2.1 Board() [1/3] `Board (`
`Board other )`

Instantiates a new **Board** (p. 21).

Parameters

other	the other

1.10.2.2 Board() [2/3] **Board** ( )

Instantiates a new **Board** (p. 21).

1.10.2.3 Board() [3/3] **Board** (
String *fen*)

Instantiates a new **Board** (p. 21).

Parameters

fen	the fen

1.10.3 Member Function Documentation

1.10.3.1 fenSetup() void fenSetup (
String *fen*)

Fen setup.

Parameters

fen	the fen

1.10.3.2 getPiece() [1/2] **Piece** getPiece (
Location *loc*)

Gets piece.

Parameters

loc	the loc

Returns

the piece

1.10.3.3 getPiece() [2/2] `Piece getPiece ( Location loc, boolean notNull )`

```
Piece getPiece (
     Location loc,
    boolean notNull )
```

Gets piece.

Parameters

loc	the loc
notNull	the not null

Returns

the piece

1.10.3.4 getRow()

```
Square[] getRow (
    int row,
    boolean flipLocs )
```

Get row square [ ].

Parameters

row	the row
flipLocs	the flip locs

Returns

the square [ ]

1.10.3.5 getSquare()

```
Square getSquare (
     Location loc )
```

Gets square.

Parameters

loc	the loc

Returns

the square

1.10.3.6 isSquareEmpty() `boolean isSquareEmpty ( Location loc )`

Is square empty boolean.

Parameters

loc	the loc

Returns

the boolean

1.10.3.7 iterator() `Iterator< Square > iterator ( )`

Iterator iterator.

Returns

the iterator

1.10.3.8 print() `void print ( )`

Print.

1.10.3.9 setPiece() `void setPiece ( Location loc, Piece piece )`

Sets piece.

Parameters

loc	the loc
piece	the piece

1.10.3.10 setSquareEmpty() `void setSquareEmpty ( Location loc )`

Sets square empty.

Parameters

loc	the loc

1.10.3.11 startingPos() `static Board startingPos ( ) [static]`

Starting pos board.

Returns

the board

1.10.3.12 toString() `String toString ( )`

To string string.

Returns

the string

1.10.4 Member Data Documentation

1.10.4.1 example `final Board example [static]`

The constant example.

1.10.4.2 startingFen `final String startingFen [static]`

The constant startingFen.

1.11 MyJFrame.BooleanClosing

Public Member Functions

- default Boolean **show** ()
- default boolean **checkClosingVal** (Boolean val)
- void **closing** ()

Additional Inherited Members

1.11.1 Detailed Description

The interface Boolean closing.

1.11.2 Member Function Documentation

1.11.2.1 checkClosingVal() `default boolean checkClosingVal ( Boolean val )`

Check closing val boolean.

Parameters

val	the val

Returns

the boolean

1.11.2.2 closing() `void closing ( )`

Closing (p. 31).

1.11.2.3 show() `default Boolean show ( )`

Show boolean.

Returns

the boolean

1.12 Callback$<$ T $>$

Public Member Functions

- void **callback** (T obj)

1.12.1 Detailed Description

Callback (p. 26) - an object callback.

Parameters

$<$*T*$>$	the object's type

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.12.2 Member Function Documentation

1.12.2.1 callback() `void callback ( T obj )`

Callback (p. 26).

Parameters

obj	the obj

1.13 CastlingRights

Classes

- enum **Side**

Public Member Functions

- **CastlingRights** (byte rights)
- **CastlingRights** ()
- **CastlingRights** (**CastlingRights** other)
- void **enableCastling** (**PlayerColor** playerColor, **Side** side)
- String **toString** ()
- boolean **isEnabled** (**PlayerColor** playerColor, **Side** side)
- byte **disableCastling** (**PlayerColor** playerColor, **Side** side)
- byte **getRights** ()
- void **enable** (byte b)
- boolean **hasAny** (**PlayerColor** playerColor)
- byte **getPlayersCastling** (**PlayerColor** playerColor)

Static Public Member Functions

- static **CastlingRights** **createFromStr** (String castlingAbilityStr)
- static void **main** (String[ ] args)
- static **PlayerColor** **whosCastling** (byte castlingRights)

Static Public Attributes

- static final String **NO_CASTLING_ABILITY**

1.13.1 Detailed Description

Castling rights - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.13.2 Constructor & Destructor Documentation

1.13.2.1 CastlingRights() `[1/3]` `CastlingRights (`

`byte rights )`

Instantiates a new Castling rights.

Parameters

rights	the rights

1.13.2.2 CastlingRights() `[2/3]` `CastlingRights ( )`

Instantiates a new Castling rights.

1.13.2.3 CastlingRights() `[3/3]` `CastlingRights ( CastlingRights other )`

Instantiates a new Castling rights.

Parameters

other	the other

1.13.3 Member Function Documentation

1.13.3.1 createFromStr() `static CastlingRights createFromStr ( String castlingAbilityStr ) [static]`

Create from str castling rights.

Parameters

castlingAbilityStr	the castling ability str

Returns

the castling rights

1.13.3.2 disableCastling() `byte disableCastling ( PlayerColor playerColor, Side side )`

Disable castling byte.

Parameters

playerColor	the player color
side	the side

Returns

the byte

1.13.3.3 enable()

```
void enable (
        byte b )
```

Enable.

Parameters

b	the b

1.13.3.4 enableCastling()

```
void enableCastling (
        PlayerColor playerColor,
        Side side )
```

Enable castling.

Parameters

playerColor	the player color
side	the side

1.13.3.5 getPlayersCastling()

```
byte getPlayersCastling (
        PlayerColor playerColor )
```

Gets players castling.

Parameters

playerColor	the player color

Returns

the players castling

1.13.3.6 getRights() `byte getRights ( )`

Gets rights.

Returns

the rights

1.13.3.7 hasAny() `boolean hasAny ( PlayerColor playerColor )`

Has any boolean.

Parameters

playerColor	the player color

Returns

the boolean

1.13.3.8 isEnabled() `boolean isEnabled ( PlayerColor playerColor, Side side )`

Is enabled boolean.

Parameters

playerColor	the player color
side	the side

Returns

the boolean

1.13.3.9 main() `static void main ( String[ ] args ) [static]`

The entry point of application.

Parameters

args	the input arguments

1.13.3.10 toString() `String toString ( )`

To string string.

Returns

the string

1.13.3.11 whosCastling() `static PlayerColor whosCastling ( byte castlingRights ) [static]`

Whos castling player color.

Parameters

castlingRights	the castling rights

Returns

the player color

1.13.4 Member Data Documentation

1.13.4.1 NO_CASTLING_ABILITY `final String NO_CASTLING_ABILITY [static]`

The constant NO_CASTLING_ABILITY.

1.14 MyJFrame.Closing< T >

Public Member Functions

- default void **tryClose** ()
- T **show** ()
- boolean **checkClosingVal** (T val)
- void **closing** (T val)

Public Attributes

- String **title**
- String **header**
- ImageIcon **icon**

1.14.1 Detailed Description

The interface **Closing** (p. 31).

Parameters

<T>	the type parameter

1.14.2 Member Function Documentation

1.14.2.1 checkClosingVal() `boolean checkClosingVal ( T val )`

Check closing val boolean.

Parameters

val	the val

Returns

the boolean

1.14.2.2 closing() `void closing ( T val )`

Closing (p. 31).

Parameters

val	the val

1.14.2.3 show() `T show ( )`

Show t.

Returns

the t

1.14.2.4 tryClose() `default void tryClose ( )`

Try close.

1.14.3 Member Data Documentation

1.14.3.1 header `String header`

The constant header.

1.14.3.2 icon `ImageIcon icon`

The constant icon.

1.14.3.3 title `String title`

The constant title.

1.15 Col

Public Member Functions

- **Col** (**Col** col)
- **Col** (String colName, String alias)
- **Col** (String colName)
- String **label** ()
- void **wrap** ()
- **Col** **time** ()
- **Col** **date** ()
- void **setWrapped** (boolean wrapped)
- **Col** **as** ()
- **Col** **as** (String alias)
- void **setColName** (String colName)
- String **colName** ()
- **Col** **of** (**Table** table)
- **Col** **of** (String ofWhom)
- **Col** **replace** (String replacing, String replaceWith)
- String **nested** ()
- **Col** **math** (**Math** operation, Object value)
- **Col** **math** (**Math** operation, Object value, boolean changeSelf)
- String **toString** ()

Static Public Member Functions

- static **Col** **count** (String **as**)
- static **Col** **count** (String **as**, Object countWhat)
- static **Col** **countIf** (String **as**, **Condition** condition)
- static CustomCol **sum** (String **as**, Col... colsToSum)
- static **Col** **switchCase** (String **as**, SwitchCase... cases)

Static Public Attributes

- static final **Col** **GameID**
- static final **Col** **SavedGame**
- static final **Col** **Password**
- static final **Col** **Player1**
- static final **Col** **Player2**
- static final **Col** **Username**
- static final **Col** **Winner**
- static final **Col** **PlayerToMove**
- static final **Col** **CreatedDateTime**
- static final **Col** **ProfilePic**

1.15.1 Detailed Description

Col (p. 33) - represents a column. either existing column in the db (the constant columns **GameID** (p. 42), **SavedGame** (p. 42) ...) or created columns.

Author

Bezalel Avrahami (bezalel3250@gmail.com)

1.15.2 Constructor & Destructor Documentation

1.15.2.1 Col() [1/3] Col (

```
Col col )
```

Instantiates a new **Col** (p. 33).

Parameters

col	the col

1.15.2.2 Col() [2/3] Col (

```
String colName,
String alias )
```

Instantiates a new **Col** (p. 33).

Parameters

colName	the col name
alias	the alias

1.15.2.3 Col() [3/3] Col (String *colName*)

Instantiates a new **Col** (p. 33).

Parameters

colName	the col name

1.15.3 Member Function Documentation

1.15.3.1 as() [1/2] Col as ( )

As own name. practically keeps the name's case.

Returns

the col

1.15.3.2 as() [2/2] Col as (String *alias*)

creates a new column with the given alias as its alias

Parameters

alias	the alias

Returns

the col

1.15.3.3 colName() `String colName ( )`

Col (p. 33) name string.

Returns

the string

1.15.3.4 count() [1/2] `static Col count ( String as ) [static]`

Count col. counts every row (∗)

Parameters

as	the alias

Returns

the col

1.15.3.5 count() [2/2] `static Col count ( String as, Object countWhat ) [static]`

Count col.

Parameters

as	the alias
countWhat	the count what

Returns

the col

1.15.3.6 countIf() `static Col countIf ( String as, Condition condition ) [static]`

Count if col. only counts if the given condition is true

Parameters

as	the as
condition	the condition

Returns

the col

1.15.3.7 date() `Col date ( )`

Date col.

Returns

a new col representing datetime

1.15.3.8 label() `String label ( )`

Label string.

Returns

the string

1.15.3.9 math() [1/2]

```
Col math (
    Math operation,
    Object value )
```

Math (p. 109) col.

Parameters

operation	the operation
value	the value

Returns

the col

1.15.3.10 math() [2/2]

```
Col math (
    Math operation,
    Object value,
    boolean changeSelf )
```

Math (p. 109) col.

Parameters

operation	the operation
value	the value
changeSelf	the change self

Returns

the col

1.15.3.11 nested()

```
String nested ( )
```

Nested string.

Returns

the string

1.15.3.12 of() [1/2]

```
Col of (
    String ofWhom )
```

new col like this that belongs to the given owner

Parameters

ofWhom	the of whom

Returns

the col

1.15.3.13 of() [2/2]

```
Col of (
    Table table )
```

new col like this that belongs to the given table

Parameters

table	the table

Returns

the col

1.15.3.14 replace()

```
Col replace (
        String replacing,
        String replaceWith )
```

Replace col.

Parameters

replacing	the replacing
replaceWith	the replace with

Returns

the col

1.15.3.15 setColName()

```
void setColName (
        String colName )
```

Sets col name.

Parameters

colName	the col name

1.15.3.16 setWrapped()

```
void setWrapped (
        boolean wrapped )
```

Sets wrapped.

Parameters

wrapped	the wrapped

1.15.3.17 sum()

```
static CustomCol sum (
            String as,
            Col... colsToSum ) [static]
```

Sum cols.

Parameters

as	the alias
colsToSum	the cols to sum

Returns

the custom col

1.15.3.18 switchCase()

```
static Col switchCase (
            String as,
            SwitchCase... cases ) [static]
```

Switch case col.

Parameters

as	the as
cases	the cases

Returns

the col

1.15.3.19 time()

```
Col time ( )
```

Time col.

Returns

a new col representing time

1.15.3.20 toString()

```
String toString ( )
```

To string string.

Returns

the string

1.15.3.21 wrap() `void wrap ( )`

Wrap.

1.15.4 Member Data Documentation

1.15.4.1 CreatedDateTime `final Col CreatedDateTime [static]`

The constant CreatedDateTime.

1.15.4.2 GameID `final Col GameID [static]`

The constant GameID.

1.15.4.3 Password `final Col Password [static]`

The constant Password.

1.15.4.4 Player1 `final Col Player1 [static]`

The constant Player1.

1.15.4.5 Player2 `final Col Player2 [static]`

The constant Player2.

1.15.4.6 PlayerToMove `final Col PlayerToMove [static]`

The constant PlayerToMove.

1.15.4.7 ProfilePic `final Col ProfilePic [static]`

The constant ProfilePic.

1.15.4.8 SavedGame `final Col SavedGame [static]`

The constant SavedGame.

1.15.4.9 Username `final Col Username [static]`

The constant Username.

1.15.4.10 Winner `final Col Winner [static]`

The constant Winner.

1.16 Condition

Classes

- enum **Relation**

Public Member Functions

- **Condition** (String str, Object... parms)
- **Condition noNulls** ()
- **Condition add** (**Condition** condition, **Relation** relation)
- **Condition add** (**Condition** condition, **Relation** relation, boolean **wrap**)
- void **wrap** ()
- **Condition and** (**Condition** condition)
- String **getStr** ()
- void **setStr** (String str)
- String **toString** ()

Static Public Member Functions

- static **Condition equals** (Object col, Object value)
- static **Condition math** (Object col, @MagicConstant(stringValues={">", ">=", "<", "<="}) String operation, Object value)
- static **Condition between** (Object col, Object start, Object end)
- static **Condition notEquals** (Object col, Object value)

1.16.1 Detailed Description

Condition (p. 43) - represents a condition.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.16.2 Class Documentation

1.16.2.1 enum ver14::SharedClasses::DBActions::Condition::Relation **Relation** (p. **??**) - relations between conditions.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Enumerator

AND	And relation.
OR	Or relation.

1.16.3 Constructor & Destructor Documentation

1.16.3.1 Condition()

```
Condition (
        String str,
        Object...  parms )
```

Instantiates a new **Condition** (p. 43).

Parameters

str	the str
parms	the parms

1.16.4 Member Function Documentation

1.16.4.1 add() [1/2]

```
Condition add (
        Condition condition,
        Relation relation )
```

Add condition.

Parameters

condition	the condition
relation	the relation

Returns

the condition

1.16.4.2 add() [2/2]

```
Condition add (
        Condition condition,
        Relation relation,
        boolean wrap )
```

Add condition.

Parameters

condition	the condition
relation	the relation
wrap	the wrap

Returns

THIS condition

1.16.4.3 and()

```
Condition and (
    Condition condition )
```

wraps

Parameters

condition	the condition

Returns

condition

1.16.4.4 between()

```
static Condition between (
    Object col,
    Object start,
    Object end ) [static]
```

Between condition.

Parameters

col	the col
start	the start
end	the end

Returns

the condition

1.16.4.5 equals()

```
static Condition equals (
    Object col,
    Object value ) [static]
```

Equals condition.

Parameters

col	the col
value	the value

Returns

the condition

1.16.4.6 getStr() `String getStr ( )`

Gets str.

Returns

the str

1.16.4.7 math()

```
static Condition math (
        Object col,
        @MagicConstant(stringValues={">", ">=", "<", "<="}) String operation,
        Object value )  [static]
```

Math condition.

Parameters

col	the col
operation	the operation
value	the value

Returns

the condition

1.16.4.8 noNulls() `Condition noNulls ( )`

No nulls condition.

Returns

the condition

1.16.4.9 notEquals() static **Condition** notEquals (Object *col*, Object *value*) [static]

Not equals condition.

Parameters

col	the col
value	the value

Returns

the condition

1.16.4.10 setStr() void setStr (String *str*)

Sets str.

Parameters

str	the str

1.16.4.11 toString() String toString ( )

To string string.

Returns

the string

1.16.4.12 wrap() void wrap ( )

Wrap.

1.17 Config< V >

Public Member Functions

- **Config** ()
- **Config** (String **description**)
- **Config** (String **description**, boolean **canUseDefault**, Described< V > defaultValue)
- **Config** (String **description**, V defVal)
- **Config** (String **description**, V defVal, String defDesc)
- **Config** (String **description**, Described< V > defaultValue)
- void **addSuggestion** (Described< V > suggestion)
- ArrayList< Described< V > > **getValuesSuggestion** ()
- Described< V > **getDescribedDefault** ()
- V **getDefault** ()
- String **toString** ()
- String **getDefaultDesc** ()

Public Attributes

- final boolean **canUseDefault**
- final String **description**

1.17.1 Detailed Description

Config (p. 49) - an argument's configuration.

Parameters

<V>	the type

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.17.2 Constructor & Destructor Documentation

1.17.2.1 Config() [1/6]

```
Config ( )
```

Instantiates a new **Config** (p. 49).

1.17.2.2 Config() [2/6]

```
Config (
        String description )
```

Instantiates a new **Config** (p. 49).

Parameters

description	the description

1.17.2.3 Config() [3/6]

```
Config (
        String description,
        boolean canUseDefault,
        Described< V > defaultValue )
```

Instantiates a new **Config** (p. 49).

Parameters

description	the description
canUseDefault	the can use default
defaultValue	the default value

1.17.2.4 Config() [4/6]

```
Config (
        String description,
        V defVal )
```

Instantiates a new **Config** (p. 49).

Parameters

description	the description
defVal	the def val

1.17.2.5 Config() [5/6]

```
Config (
        String description,
        V defVal,
        String defDesc )
```

Instantiates a new **Config** (p. 49).

Parameters

description	the description
defVal	the def val
defDesc	the def desc

1.17.2.6 Config() [6/6] Config (

```
String description,
Described< V > defaultValue )
```

Instantiates a new **Config** (p. 49).

Parameters

description	the description
defaultValue	the default value

1.17.3 Member Function Documentation

1.17.3.1 addSuggestion() void addSuggestion (

```
Described< V > suggestion )
```

Add suggestion.

Parameters

suggestion	the suggestion

1.17.3.2 getDefault() V getDefault ( )

Gets default.

Returns

the default

1.17.3.3 getDefaultDesc() String getDefaultDesc ( )

Gets default desc.

Returns

the default desc

1.17.3.4 getDescribedDefault() `Described< V > getDescribedDefault ( )`

Gets described default.

Returns

the described default

1.17.3.5 getValuesSuggestion() `ArrayList< Described< V > > getValuesSuggestion ( )`

Gets values suggestion.

Returns

the values suggestion

1.17.3.6 toString() `String toString ( )`

To string string.

Returns

the string

1.17.4 Member Data Documentation

1.17.4.1 canUseDefault `final boolean canUseDefault`

The Can use default.

1.17.4.2 description `final String description`

The Description.

1.18 ConfirmDialogs

Static Public Member Functions

- static boolean **confirm** (Component parent, String title, String message, ImageIcon icon)
- static void **main** (String[ ] args)
- static String **confirm** (Component parent, String title, String header, String message, ImageIcon icon, String initialValue)

1.18.1 Detailed Description

The utility class Confirm dialogs.

1.18.2 Member Function Documentation

1.18.2.1 confirm() [1/2]

```
static String confirm (
        Component parent,
        String title,
        String header,
        String message,
        ImageIcon icon,
        String initialValue )  [static]
```

Confirm string.

Parameters

parent	the parent
header	the header
message	the message
title	the title
icon	the icon
initialValue	the initial value

Returns

the string

1.18.2.2 confirm() [2/2]

```
static boolean confirm (
        Component parent,
        String title,
        String message,
        ImageIcon icon )  [static]
```

Confirm boolean.

Parameters

parent	the parent
title	the title
message	the message
icon	the icon

Returns

the boolean

1.18.2.3 main() `static void main ( String[] args ) [static]`

The entry point of application.

Parameters

args	the input arguments

1.19 CreatedGame

Public Member Functions

- **CreatedGame** (String **gameId**, String **creatorUsername**, GameSettings **gameSettings**)
- String **getGameDesc** ()

Additional Inherited Members

1.19.1 Detailed Description

Created game.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.19.2 Constructor & Destructor Documentation

1.19.2.1 CreatedGame()

```
CreatedGame (
        String gameId,
        String creatorUsername,
        GameSettings gameSettings )
```

Instantiates a new Created game.

Parameters

gameId	the game id
creatorUsername	the creator username
gameSettings	the game settings

1.19.3 Member Function Documentation

1.19.3.1 getGameDesc() `String getGameDesc ( )`

Gets game desc.

Returns

the game desc

Reimplemented from **GameInfo** (p. 82).

1.20 CustomStatement

Public Member Functions

- **CustomStatement** (DBRequest.Type **type**, @Language("SQL") String statement)

Protected Member Functions

- String **createStatement** ()

Additional Inherited Members

1.20.1 Detailed Description

Custom statement - a custom sql statement.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.20.2 Constructor & Destructor Documentation

1.20.2.1 CustomStatement()

```
CustomStatement (
        DBRequest.Type type,
        @Language("SQL") String statement )
```

Instantiates a new Custom statement.

Parameters

type	the type
statement	the statement

1.20.3 Member Function Documentation

1.20.3.1 createStatement() `String createStatement ( ) [protected]`

Create statement string.

Returns

the string

Reimplemented from **SQLStatement** (p. 211).

1.21 DBRequest

Classes

- enum **Type**

Public Member Functions

- **DBRequest** (**SQLStatement** sqlStatement)
- **DBRequest** (**Type type**, String request, **RequestBuilder** builder)
- **DBRequest** (**SQLStatement** sqlStatement, **RequestBuilder** builder)
- **RequestBuilder getBuilder** ()
- **DBRequest getSubRequest** ()
- void **setSubRequest** (**DBRequest** subRequest)
- String **toString** ()
- String **getRequest** ()

Public Attributes

- final **Type type**

1.21.1 Detailed Description

Db request - a database request.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.21.2 Class Documentation

1.21.2.1 enum ver14::SharedClasses::DBActions::DBRequest::DBRequest::Type Type (p. **??**) - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Enumerator

Query	Query type.
Update	Update type.

1.21.3 Constructor & Destructor Documentation

1.21.3.1 DBRequest() [1/3]

```
DBRequest (
    SQLStatement sqlStatement )
```

Instantiates a new Db request.

Parameters

sqlStatement	the sql statement

1.21.3.2 DBRequest() [2/3]

```
DBRequest (
    Type type,
    String request,
    RequestBuilder builder )
```

Instantiates a new Db request.

Parameters

type	the type
request	the request
builder	the builder

1.21.3.3 DBRequest() [3/3]

```
DBRequest (
    SQLStatement sqlStatement,
    RequestBuilder builder )
```

Instantiates a new Db request.

Parameters

sqlStatement	the sql statement
builder	the builder

1.21.4 Member Function Documentation

1.21.4.1 getBuilder() `RequestBuilder getBuilder ( )`

Gets builder.

Returns

the builder

1.21.4.2 getRequest() `String getRequest ( )`

Gets request.

Returns

the request

1.21.4.3 getSubRequest() `DBRequest getSubRequest ( )`

Gets sub request.

Returns

the sub request

1.21.4.4 setSubRequest()

```
void setSubRequest (
        DBRequest subRequest )
```

Sets sub request.

Parameters

subRequest	the sub request

1.21.4.5 toString() `String toString ( )`

To string string.

Returns

the string

1.21.5 Member Data Documentation

1.21.5.1 type `final` **`Type`** `type`

The **Type** (p. **??**).

1.22 DBResponse

Classes

- enum **Status**

Public Member Functions

- **DBRequest getRequest** ()
- **Status getStatus** ()
- boolean **isSuccess** ()
- abstract boolean **isAnyData** ()
- **DBResponse getAddedRes** ()
- void **setAddedRes** (**DBResponse addedRes**)
- abstract **DBResponse clean** ()
- void **print** ()
- String **toString** ()

Protected Member Functions

- **DBResponse** (**Status status**, **DBRequest request**)

Protected Attributes

- final **Status status**
- final **DBRequest request**
- **DBResponse addedRes**

1.22.1 Detailed Description

Db response - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.22.2 Class Documentation

1.22.2.1 enum ver14::SharedClasses::DBActions::DBResponse::DBResponse::Status Status (p. ??) - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Enumerator

ERROR	Error status.
SUCCESS	Success status.

1.22.3 Constructor & Destructor Documentation

1.22.3.1 DBResponse() `DBResponse ( Status status, DBRequest request ) [protected]`

Instantiates a new Db response.

Parameters

status	the status
request	the request

1.22.4 Member Function Documentation

1.22.4.1 clean() `abstract DBResponse clean ( ) [abstract]`

Clean db response.

Returns

the db response

Reimplemented in **StatusResponse** (p. 216), and **TableDBResponse** (p. 233).

1.22.4.2 getAddedRes() `DBResponse getAddedRes ( )`

Gets added res.

Returns

the added res

1.22.4.3 getRequest() `DBRequest getRequest ( )`

Gets request.

Returns

the request

1.22.4.4 getStatus() `Status getStatus ( )`

Gets status.

Returns

the status

1.22.4.5 isAnyData() `abstract boolean isAnyData ( ) [abstract]`

Is any data boolean.

Returns

the boolean

Reimplemented in **StatusResponse** (p. 216), and **TableDBResponse** (p. 235).

1.22.4.6 isSuccess() `boolean isSuccess ( )`

Is success boolean.

Returns

the boolean

1.22.4.7 print() `void print ( )`

Print.

1.22.4.8 setAddedRes() `void setAddedRes ( DBResponse addedRes )`

Sets added res.

Parameters

addedRes	the added res

1.22.4.9 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented in **TableDBResponse** (p. 236).

1.22.5 Member Data Documentation

1.22.5.1 addedRes `DBResponse addedRes [protected]`

The Added res.

1.22.5.2 request `final DBRequest request [protected]`

The Request.

1.22.5.3 status `final Status status [protected]`

The **Status** (p. **??**).

1.23 Delete

Public Member Functions

- **Delete** (**Table** deletingFrom, **Condition** condition)

Protected Member Functions

- String **createStatement** ()

Additional Inherited Members

1.23.1 Detailed Description

Delete (p. 62) - deletion statement.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.23.2 Constructor & Destructor Documentation

1.23.2.1 Delete()

```
Delete (
        Table deletingFrom,
        Condition condition )
```

Instantiates a new **Delete** (p. 62).

Parameters

deletingFrom	the deleting from
condition	the condition

1.23.3 Member Function Documentation

1.23.3.1 createStatement()

```
String createStatement ( )  [protected]
```

Create statement string.

Returns

the string

Reimplemented from **SQLStatement** (p. 211).

1.24 Direction Enum Reference

Public Member Functions

- **Direction**[ ] **getCombination** ()
- **Direction perspective** (**PlayerColor** playerColor)
- abstract **Direction opposite** ()

Static Public Member Functions

- static **Direction** **getRelative** (**Location** loc1, **Location** loc2)
- static **Direction** **getDirectionByOffset** (int **offset**)

Public Attributes

- **U**
- **D**
- **L**
- **R**
- **U_U**
- **D_D**
- **U_R**
- **U_L**
- **D_R**
- **D_L**
- **U_U_R**
- **U_U_L**
- **U_R_R**
- **U_L_L**
- **D_D_R**
- **D_D_L**
- **D_R_R**
- **D_L_L**
- final long **andWith**
- final int **offset**
- final int **asInt**
- final **Direction**[ ] **combination**

Static Public Attributes

- static final int **NUM_OF_DIRECTIONS**
- static final int **NUM_OF_KNIGHT_DIRECTIONS**
- static final int **NUM_OF_DIRECTIONS_WO_KNIGHT**
- static final **Direction**[ ] **ALL_DIRECTIONS**
- static final List< **Direction** > **ALL_USED_DIRECTIONS**
- static final **PlayerColor** **normalPerspective**

1.24.1 Detailed Description

Direction (p. 63) - represents a moving direction on a board. sort of like a vector. has an `offset (p. 68)` that is added to a certain location or bitboard, in order to achieve movement in that direction. the general direction map looks like this:

-9	-8	-7
-1	loc	1
7	8	9

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.24.2 Member Function Documentation

1.24.2.1 getCombination() `Direction[] getCombination ( )`

Get combination direction [ ].

Returns

the direction [ ]

1.24.2.2 getDirectionByOffset() `static Direction getDirectionByOffset ( int offset ) [static]`

Gets direction by offset.

Parameters

offset	the offset

Returns

the direction by offset

1.24.2.3 getRelative() `static Direction getRelative ( Location loc1, Location loc2 ) [static]`

Gets relative.

Parameters

loc1	the loc 1
loc2	the loc 2

Returns

the relative

1.24.2.4 opposite() `abstract` **`Direction`** `opposite ( ) [abstract]`

the Opposite direction to this one.

Returns

the direction

1.24.2.5 perspective() **`Direction`** `perspective (` **`PlayerColor`** *`playerColor`* `)`

gets the correct perspective for the provided player color.this is necessary because for example: a white pawn push(**`U`** (p. 68)) is the exact opposite of a black pawn push (**`D`** (p. 67)). so the perspective needs to be in relation to the moving color.

Parameters

playerColor	the player color

Returns

the direction

1.24.3 Member Data Documentation

1.24.3.1 ALL_DIRECTIONS `final` **`Direction`** `[] ALL_DIRECTIONS [static]`

The All directions.

1.24.3.2 ALL_USED_DIRECTIONS `final List<` **`Direction`**`> ALL_USED_DIRECTIONS [static]`

The All used directions.

1.24.3.3 andWith `final long andWith`

some directions need to filter false positives. for example: moving left one square from the left-most column, will overflow to the previous row. to fix this problem some directions have a andWith value they have to perform a bitwise and with, after every offset. to cancel the false positives. in the left direction example, the andWith is the whole board but the right-most column

1.24.3.4 asInt `final int asInt`

an int for quick access by index.

1.24.3.5 combination `final Direction [] combination`

some

1.24.3.6 D `D`

one square down the board.

1.24.3.7 D_D `D_D`

two squares down.

1.24.3.8 D_D_L `D_D_L`

two squares down and one square left.

1.24.3.9 D_D_R `D_D_R`

two squares down and one square right.

1.24.3.10 D_L `D_L`

one square down and one square left.

1.24.3.11 D_L_L `D_L_L`

two squares left and one square down.

1.24.3.12 D_R `D_R`

one square down and one square right.

1.24.3.13 D_R_R `D_R_R`

two squares right and one square down.

1.24.3.14 L `L`

one square left.

1.24.3.15 normalPerspective `final PlayerColor normalPerspective [static]`

the perspective the offset is correct for. if the moving piece's color is not this value the direction need to be flipped.

1.24.3.16 NUM_OF_DIRECTIONS `final int NUM_OF_DIRECTIONS [static]`

The constant NUM_OF_DIRECTIONS.

1.24.3.17 NUM_OF_DIRECTIONS_WO_KNIGHT `final int NUM_OF_DIRECTIONS_WO_KNIGHT [static]`

The constant NUM_OF_DIRECTIONS_WO_KNIGHT.

1.24.3.18 NUM_OF_KNIGHT_DIRECTIONS `final int NUM_OF_KNIGHT_DIRECTIONS [static]`

The constant NUM_OF_KNIGHT_DIRECTIONS.

1.24.3.19 offset `final int offset`

The actual offset.

1.24.3.20 R `R`

one square right.

1.24.3.21 U `U`

one square up the board.

1.24.3.22 U_L `U_L`

one square up and one square left.

1.24.3.23 U_L_L `U_L_L`

two squares left and one square up.

1.24.3.24 U_R `U_R`

one square up and one square right.

1.24.3.25 U_R_R `U_R_R`

two squares right and one square up.

1.24.3.26 U_U `U_U`

two squares up.

1.24.3.27 U_U_L `U_U_L`

two squares up and one square left.

1.24.3.28 U_U_R `U_U_R`

two squares up and one square right.

1.25 EnvManager

Public Member Functions

- void **handledErr** (**MyError** err)
- void **criticalErr** (**MyError** err)

1.25.1 Detailed Description

Env manager - an object that can handle errors as they occur.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.25.2 Member Function Documentation

1.25.2.1 criticalErr()

```
void criticalErr (
        MyError err )
```

notifies manager of an un-handleable error. the manager must shut down everything

Parameters

err	the error thrown

1.25.2.2 handledErr()

```
void handledErr (
        MyError err )
```

notifies manager of a managed error

Parameters

err	the error thrown

1.26 ErrorHandler< E extends MyError >

Public Member Functions

- void **handle** (**MyError** err)

Static Public Member Functions

- static boolean **ignore** (ThrowingRunnable runnable)

1.26.1 Detailed Description

The interface Error handler.

Parameters

<E>	the type parameter

1.26.2 Member Function Documentation

1.26.2.1 handle()

```
void handle (
          MyError err )
```

Handle.

Parameters

err	the err

1.26.2.2 ignore()

```
static boolean ignore (
          ThrowingRunnable runnable )  [static]
```

Ignore boolean.

Parameters

runnable	the runnable

Returns

true if the runnable threw, false otherwise

1.27 EstablishedGameInfo

Public Member Functions

- Date **getCreatedAt** ()
- void **setCreatedAt** (Date **createdAt**)
- Stack< **Move** > **getMoveStack** ()
- String **toString** ()
- String **getGameDesc** ()

Public Attributes

- final String **opponentUsername**

Protected Member Functions

- **EstablishedGameInfo** (String **gameId**, String **creatorUsername**, String **opponentUsername**, GameSettings **gameSettings**, Stack< **Move** > moveStack)

Protected Attributes

- Date **createdAt**

Additional Inherited Members

1.27.1 Detailed Description

Established game info.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.27.2 Constructor & Destructor Documentation

1.27.2.1 EstablishedGameInfo()

```
EstablishedGameInfo (
        String gameId,
        String creatorUsername,
        String opponentUsername,
        GameSettings gameSettings,
        Stack< Move > moveStack )  [protected]
```

Instantiates a new Established game info.

Parameters

gameId	the game id
creatorUsername	the creator username
opponentUsername	the opponent username
gameSettings	the game settings
moveStack	the move stack

1.27.3 Member Function Documentation

1.27.3.1 getCreatedAt() `Date getCreatedAt ( )`

Gets created at.

Returns

the created at

1.27.3.2 getGameDesc() `String getGameDesc ( )`

Gets game desc.

Returns

the game desc

Reimplemented from **GameInfo** (p. 82).

1.27.3.3 getMoveStack() `Stack< Move > getMoveStack ( )`

Gets move stack.

Returns

the move stack

1.27.3.4 setCreatedAt()

```
void setCreatedAt (
    Date createdAt )
```

Sets created at.

Parameters

created↩ At	the created at

1.27.3.5 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented from **GameInfo** (p. 83).

Reimplemented in **ArchivedGameInfo** (p. 7).

1.27.4 Member Data Documentation

1.27.4.1 createdAt `Date createdAt [protected]`

The Created at.

1.27.4.2 opponentUsername `final String opponentUsername`

The Opponent username.

1.28 Evaluation

Public Member Functions

- **Evaluation** (**GameStatus** gameStatus, **PlayerColor** evaluationFor)
- **Evaluation** (int eval, **GameStatus** gameStatus, **PlayerColor** evaluationFor)
- **Evaluation** (**PlayerColor** evaluationFor)
- **Evaluation** (**Evaluation** other)
- void **addDetail** (**EvaluationParameters** parm, int value)
- void **assertNotGameOver** ()
- Integer **getEvaluationDepth** ()
- void **setEvaluationDepth** (Integer evaluationDepth)
- boolean **isGameOver** ()
- boolean **isCheck** ()
- boolean **isGreaterThan** (**Evaluation** other)
- int **getEval** ()
- void **setEval** (int eval)
- **GameStatus** **getGameStatus** ()
- boolean **equals** (Object o)
- String **toString** ()
- **PlayerColor** **getEvaluationFor** ()
- **Evaluation** **setPerspective** (**PlayerColor** playerColor)
- void **flipEval** ()
- void **print** ()
- float **convertFromCentipawns** ()
- record **EvaluationDetail** (**EvaluationParameters** parm, double eval) implements Serializable

Static Public Member Functions

- static **Evaluation book** ()

Static Public Attributes

- static final int **TIE_EVAL**
- static final int **WIN_EVAL**
- static final int **LOSS_EVAL**

1.28.1 Detailed Description

Evaluation (p. 73).

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.28.2 Constructor & Destructor Documentation

1.28.2.1 Evaluation() [1/4] Evaluation

```
Evaluation (
        GameStatus gameStatus,
        PlayerColor evaluationFor )
```

Instantiates a new **Evaluation** (p. 73).

Parameters

gameStatus	the game status
evaluationFor	the evaluation for

1.28.2.2 Evaluation() [2/4] Evaluation

```
Evaluation (
       int eval,
        GameStatus gameStatus,
        PlayerColor evaluationFor )
```

Instantiates a new **Evaluation** (p. 73).

Parameters

eval	the eval
gameStatus	the game status
evaluationFor	the evaluation for

1.28.2.3 Evaluation() [3/4] `Evaluation ( PlayerColor evaluationFor )`

Instantiates a new **Evaluation** (p. 73).

Parameters

evaluationFor	the evaluation for

1.28.2.4 Evaluation() [4/4] `Evaluation ( Evaluation other )`

Instantiates a new **Evaluation** (p. 73).

Parameters

other	the other

1.28.3 Member Function Documentation

1.28.3.1 addDetail() `void addDetail ( EvaluationParameters parm, int value )`

Add detail.

Parameters

parm	the parm
value	the value

1.28.3.2 assertNotGameOver() `void assertNotGameOver ( )`

Assert not game over.

1.28.3.3 book() `static Evaluation book ( ) [static]`

Book evaluation.

Returns

the evaluation

1.28.3.4 convertFromCentipawns() `float convertFromCentipawns ( )`

Convert from centipawns float.

Returns

the float

1.28.3.5 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

1.28.3.6 EvaluationDetail() `record EvaluationDetail ( EvaluationParameters parm, double eval )`

Evaluation (p. 73) detail.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.28.3.7 flipEval() `void flipEval ( )`

Flip eval.

1.28.3.8 getEval() `int getEval ( )`

Gets eval.

Returns

the eval

1.28.3.9 getEvaluationDepth() `Integer getEvaluationDepth ( )`

Gets evaluation depth.

Returns

the evaluation depth

1.28.3.10 getEvaluationFor() `PlayerColor getEvaluationFor ( )`

Gets evaluation for.

Returns

the evaluation for

1.28.3.11 getGameStatus() `GameStatus getGameStatus ( )`

Gets game status.

Returns

the game status

1.28.3.12 isCheck() `boolean isCheck ( )`

Is check boolean.

Returns

the boolean

1.28.3.13 isGameOver() `boolean isGameOver ( )`

Is game over boolean.

Returns

the boolean

1.28.3.14 isGreaterThan() `boolean isGreaterThan (`

```
        Evaluation other )
```

Is greater than boolean.

Parameters

other	the other

Returns

the boolean

1.28.3.15 print() `void print ( )`

Print.

1.28.3.16 setEval() `void setEval ( int eval )`

Sets eval.

Parameters

eval	the eval

1.28.3.17 setEvaluationDepth() `void setEvaluationDepth ( Integer evaluationDepth )`

Sets evaluation depth.

Parameters

evaluationDepth	the evaluation depth

1.28.3.18 setPerspective() `Evaluation setPerspective ( PlayerColor playerColor )`

Sets perspective.

Parameters

playerColor	the player color

Returns

the perspective

1.28.3.19 toString() `String toString ( )`

To string string.

Returns

the string

1.28.4 Member Data Documentation

1.28.4.1 LOSS_EVAL `final int LOSS_EVAL [static]`

The constant LOSS_EVAL.

1.28.4.2 TIE_EVAL `final int TIE_EVAL [static]`

The constant TIE_EVAL.

1.28.4.3 WIN_EVAL `final int WIN_EVAL [static]`

The constant WIN_EVAL.

1.29 EvaluationParameters Enum Reference

Public Attributes

- **MATERIAL**
- **PIECE_TABLES**
- **KING_SAFETY**
- **HANGING_PIECES**
- **SQUARE_CONTROL**
- **MOVEMENT_ABILITY**
- **FORCE_KING_TO_CORNER**
- **EG_WEIGHT**
- **STOCKFISH_SAYS**
- final double **weight**

1.29.1 Detailed Description

Evaluation (p. 73) parameters - all evaluation parameters. (some are unused).

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.29.2 Member Data Documentation

1.29.2.1 EG_WEIGHT `EG_WEIGHT`

Eg weight evaluation parameters.

1.29.2.2 FORCE_KING_TO_CORNER `FORCE_KING_TO_CORNER`

Force king to corner evaluation parameters.

1.29.2.3 HANGING_PIECES `HANGING_PIECES`

Hanging pieces evaluation parameters.

1.29.2.4 KING_SAFETY `KING_SAFETY`

King safety evaluation parameters.

1.29.2.5 MATERIAL `MATERIAL`

Material evaluation parameters.

1.29.2.6 MOVEMENT_ABILITY `MOVEMENT_ABILITY`

Movement ability evaluation parameters.

1.29.2.7 PIECE_TABLES `PIECE_TABLES`

Piece tables evaluation parameters.

1.29.2.8 SQUARE_CONTROL `SQUARE_CONTROL`

Square control evaluation parameters.

1.29.2.9 STOCKFISH_SAYS `STOCKFISH_SAYS`

Stockfish says evaluation parameters.

1.29.2.10 weight `final double weight`

The parameter's weight

1.30 GameInfo

Public Member Functions

- boolean **isCreator** (String username)
- String **ID** ()
- boolean **equals** (Object o)
- String **toString** ()
- abstract String **getGameDesc** ()
- **PlayerColor getJoiningPlayerColor** ()
- **PlayerColor getStartingColor** ()

Static Public Member Functions

- static **GameInfo example** ()

Public Attributes

- final String **gameId**
- final String **creatorUsername**
- final GameSettings **gameSettings**

Protected Member Functions

- **GameInfo** (String **gameId**, String **creatorUsername**, GameSettings **gameSettings**)

1.30.1 Detailed Description

Game info.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.30.2 Constructor & Destructor Documentation

1.30.2.1 GameInfo()

```
GameInfo (
        String gameId,
        String creatorUsername,
        GameSettings gameSettings )  [protected]
```

Instantiates a new Game info.

Parameters

gameId	the game id
creatorUsername	the creator username
gameSettings	the game settings

1.30.3 Member Function Documentation

1.30.3.1 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

1.30.3.2 example() `static GameInfo example ( ) [static]`

Example game info.

Returns

the game info

1.30.3.3 getGameDesc() `abstract String getGameDesc ( ) [abstract]`

Gets game desc.

Returns

the game desc

Reimplemented in **CreatedGame** (p. 55), and **EstablishedGameInfo** (p. 72).

1.30.3.4 getJoiningPlayerColor() `PlayerColor getJoiningPlayerColor ( )`

Gets joining player color.

Returns

the joining player color

1.30.3.5 getStartingColor() `PlayerColor getStartingColor ( )`

Gets starting color.

Returns

the starting color

1.30.3.6 ID() `String ID ( )`

Id string.

Returns

the string

1.30.3.7 isCreator()

```
boolean isCreator (
        String username )
```

Is creator boolean.

Parameters

username	the username

Returns

the boolean

1.30.3.8 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented in **ArchivedGameInfo** (p. 7), and **EstablishedGameInfo** (p. 73).

1.30.4 Member Data Documentation

1.30.4.1 creatorUsername `final String creatorUsername`

The Creator username.

1.30.4.2 gameId `final String gameId`

The Game id.

1.30.4.3 gameSettings `final GameSettings gameSettings`

The Game settings.

1.31 GameStatus

Classes

- enum **GameStatusType**
- enum **SpecificStatus**

Public Member Functions

- void **setCustomStr** (String customStr)
- boolean **isDisconnected** ()
- **Location getCheckedKingLoc** ()
- **PlayerColor getWinningColor** ()
- **GameStatusType getGameStatusType** ()
- void **setInCheck** (**Location** checkedKingLoc)
- boolean **isCheck** ()
- String **getDetailedStr** ()
- String **getDetailedStr** (Map< **PlayerColor**, String > playerUsernamesMap)
- boolean **isGameOver** ()
- String **toString** ()
- void **setDepth** (int depth)
- **SpecificStatus getSpecificStatus** ()

Static Public Member Functions

- static **GameStatus** **checkmate** (**PlayerColor** winningPlayerColor, **Location** matedKing)
- static **GameStatus** **gameGoesOn** ()
- static **GameStatus** **tieByAgreement** ()
- static **GameStatus** **stalemate** ()
- static **GameStatus** **fiftyMoveRule** ()
- static **GameStatus** **serverStoppedGame** (String cause)
- static **GameStatus** **threeFoldRepetition** ()
- static **GameStatus** **insufficientMaterial** ()
- static **GameStatus** **playerDisconnected** (**PlayerColor** disconnectedPlayer, boolean isVsAi)
- static **GameStatus** **timedOut** (**PlayerColor** timedOutPlayer, boolean isSufficientMaterial)
- static **GameStatus** **playerResigned** (**PlayerColor** resignedPlayer)

1.31.1 Detailed Description

Game status - represents a game status .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.31.2 Member Function Documentation

1.31.2.1 checkmate()

```
static GameStatus checkmate (
        PlayerColor winningPlayerColor,
        Location matedKing ) [static]
```

Checkmate game status.

Parameters

winningPlayerColor	the winning player color
matedKing	the mated king

Returns

the game status

1.31.2.2 fiftyMoveRule()

```
static GameStatus fiftyMoveRule ( ) [static]
```

Fifty move rule game status.

Returns

the game status

1.31.2.3 gameGoesOn() `static GameStatus gameGoesOn ( ) [static]`

Game goes on game status.

Returns

the game status

1.31.2.4 getCheckedKingLoc() `Location getCheckedKingLoc ( )`

Gets checked king loc.

Returns

the checked king loc

1.31.2.5 getDetailedStr() [1/2] `String getDetailedStr ( )`

Gets detailed str.

Returns

the detailed str

1.31.2.6 getDetailedStr() [2/2]

```
String getDetailedStr (
        Map< PlayerColor, String > playerUsernamesMap )
```

Gets detailed str.

Parameters

playerUsernamesMap	the player usernames map

Returns

the detailed str

1.31.2.7 getGameStatusType() `GameStatusType getGameStatusType ( )`

Gets game status type.

Returns

the game status type

1.31.2.8 getSpecificStatus() `SpecificStatus getSpecificStatus ( )`

Gets specific status.

Returns

the specific status

1.31.2.9 getWinningColor() `PlayerColor getWinningColor ( )`

Gets winning color.

Returns

the winning color

1.31.2.10 insufficientMaterial() `static GameStatus insufficientMaterial ( ) [static]`

Insufficient material game status.

Returns

the game status

1.31.2.11 isCheck() `boolean isCheck ( )`

Is check boolean.

Returns

the boolean

1.31.2.12 isDisconnected() `boolean isDisconnected ( )`

Is disconnected boolean.

Returns

the boolean

1.31.2.13 isGameOver() `boolean isGameOver ( )`

Is game over boolean.

Returns

the boolean

1.31.2.14 playerDisconnected() `static GameStatus playerDisconnected ( PlayerColor disconnectedPlayer, boolean isVsAi ) [static]`

Player disconnected game status.

Parameters

disconnectedPlayer	the disconnected player
isVsAi	the is vs ai

Returns

the game status

1.31.2.15 playerResigned() `static GameStatus playerResigned ( PlayerColor resignedPlayer ) [static]`

Player resigned game status.

Parameters

resignedPlayer	the resigned player

Returns

the game status

1.31.2.16 serverStoppedGame()

```
static GameStatus serverStoppedGame (
        String cause ) [static]
```

Server stopped game game status.

Parameters

cause	the cause

Returns

the game status

1.31.2.17 setCustomStr()

```
void setCustomStr (
        String customStr )
```

Sets custom str.

Parameters

customStr	the custom str

1.31.2.18 setDepth()

```
void setDepth (
        int depth )
```

Sets depth.

Parameters

depth	the depth

1.31.2.19 setInCheck()

```
void setInCheck (
        Location checkedKingLoc )
```

Sets in check.

Parameters

checkedKingLoc	the checked king loc

1.31.2.20 stalemate() `static GameStatus stalemate ( ) [static]`

Stalemate game status.

Returns

the game status

1.31.2.21 threeFoldRepetition() `static GameStatus threeFoldRepetition ( ) [static]`

Three fold repetition game status.

Returns

the game status

1.31.2.22 tieByAgreement() `static GameStatus tieByAgreement ( ) [static]`

Tie by agreement game status.

Returns

the game status

1.31.2.23 timedOut()

```
static GameStatus timedOut (
    PlayerColor timedOutPlayer,
   boolean isSufficientMaterial ) [static]
```

Timed out game status.

Parameters

timedOutPlayer	the timed out player
isSufficientMaterial	the is sufficient material

Returns

the game status

1.31.2.24 toString() `String toString ( )`

To string string.

Returns

the string

1.32 GameStatus.GameStatusType Enum Reference

Public Member Functions

- **GameStatusType** (String **annotation**)
- **GameStatusType** (String **annotation**, String **gameOverStr**)
- boolean **isGameOver** ()

Public Attributes

- **TIE**
- **CHECK**
- **GAME_GOES_ON**
- **WIN_OR_LOSS**
- **UNFINISHED**
- final String **annotation**
- final String **gameOverStr**

1.32.1 Detailed Description

Game status type .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.32.2 Constructor & Destructor Documentation

1.32.2.1 GameStatusType() [1/2] `GameStatusType ( String annotation )`

Instantiates a new Game status type.

Parameters

annotation	the annotation

1.32.2.2 GameStatusType() [2/2] `GameStatusType ( String annotation, String gameOverStr )`

Instantiates a new Game status type.

Parameters

annotation	the annotation
gameOverStr	the game over str

1.32.3 Member Function Documentation

1.32.3.1 isGameOver() `boolean isGameOver ( )`

Is game over.

Returns

true if is game over. false otherwise

1.32.4 Member Data Documentation

1.32.4.1 annotation `final String annotation`

The game status annotation.

1.32.4.2 CHECK `CHECK`

Check game status type.

1.32.4.3 GAME_GOES_ON `GAME_GOES_ON`

Game goes on game status type.

1.32.4.4 gameOverStr `final String gameOverStr`

game over str

1.32.4.5 TIE `TIE`

Tie game status type.

1.32.4.6 UNFINISHED `UNFINISHED`

Unfinished game status type.

1.32.4.7 WIN_OR_LOSS `WIN_OR_LOSS`

Win or loss game status type.

1.33 HandledThread

Public Member Functions

- **HandledThread** ()
- **HandledThread** (ThrowingRunnable runnable)
- void **setRunnable** (ThrowingRunnable runnable)

Static Public Member Functions

- static **HandledThread** **runInHandledThread** (ThrowingRunnable runnable)

Protected Member Functions

- void **handledRun** () throws Throwable

1.33.1 Detailed Description

The type Handled thread.

1.33.2 Constructor & Destructor Documentation

1.33.2.1 HandledThread() `[1/2]` `HandledThread ( )`

Instantiates a new Handled thread.

1.33.2.2 HandledThread() `[2/2]` `HandledThread (`

`ThrowingRunnable` *`runnable`* `)`

Instantiates a new Handled thread.

Parameters

runnable	the runnable

1.33.3 Member Function Documentation

1.33.3.1 handledRun() `void handledRun ( ) throws Throwable [protected]`

Handled run.

Exceptions

Throwable	the throwable

Reimplemented from **MyThread** (p. 169).

1.33.3.2 runInHandledThread() `static HandledThread runInHandledThread ( ThrowingRunnable runnable ) [static]`

Run in handled thread handled thread.

Parameters

runnable	the runnable

Returns

the handled thread

1.33.3.3 setRunnable() `void setRunnable ( ThrowingRunnable runnable )`

Sets runnable.

Parameters

runnable	the runnable

1.34 MyJFrame.MyAdapter.HeldDown

Public Member Functions

- void **startPress** ()
- void **endPress** ()
- int **key** ()

1.34.1 Detailed Description

The interface Held down.

1.34.2 Member Function Documentation

1.34.2.1 endPress() `void endPress ( )`

End press.

1.34.2.2 key() `int key ( )`

Key int.

Returns

the int

1.34.2.3 startPress() `void startPress ( )`

Start press.

1.35 IDsGenerator

Public Member Functions

- synchronized String **generate** ()
- boolean **canUseId** (String id)

1.35.1 Detailed Description

IDs generator.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.35.2 Member Function Documentation

1.35.2.1 canUseId() `boolean canUseId ( String id )`

Can use id boolean.

Parameters

id	the id

Returns

the boolean

1.35.2.2 generate() `synchronized String generate ( )`

Generate id.

Returns

the string

1.36 StrUtils.IterationThingy< T >

Public Member Functions

- T **iteration** (int i, boolean isLast)

1.36.1 Detailed Description

Iteration thingy.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.36.2 Member Function Documentation

1.36.2.1 iteration()

```
T iteration (
        int i,
        boolean isLast )
```

Iteration t.

Parameters

i	the
isLast	the is last

Returns

the t

1.37 Location Enum Reference

Public Member Functions

- **Location** ()
- String **matrixStr** ()
- String **getColString** ()
- **Location flip** ()
- boolean **isBlackSquare** ()
- boolean **isWhiteSquare** ()
- String **toString** ()
- int **getMaxDistance** (**Location** other)
- String **getRowString** ()
- int **hash** (**PieceType** pieceType)

Static Public Member Functions

- static String **locsMatStr** ()
- static **Location**[ ][ ] **locsMat** ()
- static String **createMatIndicesStr** ()
- static **Location getLoc** (**Location** loc, **Direction** direction)
- static **Location getLoc** (**Location** loc, int numOfMult, **Direction** direction)
- static **Location getLoc** (**Location** loc, int add)
- static **Location getLoc** (int locIndex)
- static **Location valueOf** (int locIndex)
- static **Location getLoc** (String str)
- static **Location getLoc** (int **row**, int **col**)
- static **Location getLoc** (int **row**, int **col**, boolean **flip**)
- static int **flip** (int num)
- static boolean **isInBounds** (**Location** loc)

Public Attributes

- final long **asLong**
- final int **asInt**
- final int **row**
- final int **col**

1.37.1 Detailed Description

Location (p. 97) - an enum consisting of 64 values representing all 64 squares on the board. used to access squares on the board
an enum is used for performance reasons.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.37.2 Constructor & Destructor Documentation

1.37.2.1 Location() `Location ( )`

Instantiates a new **Location** (p. 97).

1.37.3 Member Function Documentation

1.37.3.1 createMatIndicesStr() `static String createMatIndicesStr ( ) [static]`

Create mat indices str string.

Returns

the string

1.37.3.2 flip() `[1/2]` `Location flip ( )`

Flip location.

Returns

the location

1.37.3.3 flip() `[2/2]`

```
static int flip (
        int num ) [static]
```

Flip int.

Parameters

num	the num

Returns

the int

1.37.3.4 getColString() `String getColString ( )`

Gets col string.

Returns

the col string

1.37.3.5 getLoc() [1/7]

```
static Location getLoc (
            int locIndex )  [static]
```

Gets location corresponding to the locIndex provided (0..63)

Parameters

locIndex	the locIndex

Returns

the location if the provided index is inside the bounds(0...63). null otherwise

1.37.3.6 getLoc() [2/7]

```
static Location getLoc (
            int row,
            int col )  [static]
```

Gets loc.

Parameters

row	the row
col	the col

Returns

the loc

1.37.3.7 getLoc() [3/7]

```
static Location getLoc (
            int row,
            int col,
            boolean flip )  [static]
```

Gets loc.

Parameters

row	the row
col	the col
flip	the flip

Returns

the loc

1.37.3.8 getLoc() [4/7] static Location getLoc (Location loc, Direction direction) [static]

```
static Location getLoc (
        Location loc,
        Direction direction )  [static]
```

Gets the location relative to loc in the direction

Parameters

loc	the loc
direction	the direction

Returns

the location if the calculated index is inside the bounds(0...63). null otherwise

1.37.3.9 getLoc() [5/7]

```
static Location getLoc (
        Location loc,
       int add )  [static]
```

Gets the location that is exactly add squares from loc NOTE: add should be in bitboard format

Parameters

loc	the loc
add	the number of squares to add

Returns

the location if the calculated index is inside the bounds(0...63). null otherwise

See also

ver14.Model.Bitboard

1.37.3.10 getLoc() [6/7]

```
static Location getLoc (
        Location loc,
        int numOfMult,
        Direction direction ) [static]
```

Gets the location relative to loc in the direction given and the distance is determined by the numOfMult

Parameters

loc	the loc
numOfMult	the num of mult
direction	the direction

Returns

the location if the calculated index is inside the bounds(0...63). null otherwise

1.37.3.11 getLoc() [7/7]

```
static Location getLoc (
        String str ) [static]
```

gets the location corresponding to the provided location string.

Parameters

str	the square's coordinate according to the Algebraic notation

Returns

the loc if given str is valid. null otherwise

See also

...

1.37.3.12 getMaxDistance()

```
int getMaxDistance (
        Location other )
```

Gets max distance.

Parameters

other	the other

Returns

the max distance

1.37.3.13 getRowString() `String getRowString ( )`

Gets row string.

Returns

the row string

1.37.3.14 hash() `int hash ( PieceType pieceType )`

Hash int.

Parameters

pieceType	the piece type

Returns

the int

1.37.3.15 isBlackSquare() `boolean isBlackSquare ( )`

Is black square boolean.

Returns

the boolean

1.37.3.16 isInBounds() `static boolean isInBounds ( Location loc ) [static]`

Is in bounds boolean.

Parameters

loc	the loc

Returns

the boolean

1.37.3.17 isWhiteSquare() `boolean isWhiteSquare ( )`

Is white square boolean.

Returns

the boolean

1.37.3.18 locsMat() `static Location[ ][ ] locsMat ( ) [static]`

Locs mat location [ ] [ ].

Returns

the location [ ] [ ]

1.37.3.19 locsMatStr() `static String locsMatStr ( ) [static]`

Locs mat str string.

Returns

the string

1.37.3.20 matrixStr() `String matrixStr ( )`

Matrix str string.

Returns

the string

1.37.3.21 toString() `String toString ( )`

To string string.

Returns

the string

1.37.3.22 valueOf()

```
static Location valueOf (
        int locIndex ) [static]
```

Value of location.

Parameters

locIndex	the locIndex

Returns

the location if the provided index is inside the bounds(0...63). null otherwise

1.37.4 Member Data Documentation

1.37.4.1 asInt `final int asInt`

the calculated index

1.37.4.2 asLong `final long asLong`

a long value with a bit set on this location

1.37.4.3 col `final int col`

this location's column

1.37.4.4 row `final int row`

this location's row

1.38 LoginInfo

Public Member Functions

- **LoginInfo** ()
- **LoginInfo** (String username, String password, **LoginType** loginType)
- **LoginInfo** (**LoginType** loginType)
- String **getProfilePic** ()
- void **setProfilePic** (String profilePic)
- boolean **isGuest** ()
- **LoginType getLoginType** ()
- void **setLoginType** (**LoginType** loginType)
- String **getUsername** ()
- void **setUsername** (String username)
- String **getPassword** ()
- void **setPassword** (String password)
- String **toString** ()
- boolean **asUser** ()
- void **initDebugLoginValues** ()

1.38.1 Detailed Description

Login info.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.38.2 Constructor & Destructor Documentation

1.38.2.1 LoginInfo() [1/3] `LoginInfo ( )`

Instantiates a new Login info.

1.38.2.2 LoginInfo() [2/3] `LoginInfo (`

```
String username,
String password,
 LoginType loginType )
```

Instantiates a new Login info.

Parameters

username	the username
password	the password
loginType	the login type

1.38.2.3 LoginInfo() [3/3] `LoginInfo (`

```
 LoginType loginType )
```

Instantiates a new Login info.

Parameters

loginType	the login type

1.38.3 Member Function Documentation

1.38.3.1 asUser() `boolean asUser ( )`

As user boolean.

Returns

the boolean

1.38.3.2 getLoginType() `LoginType getLoginType ( )`

Gets login type.

Returns

the login type

1.38.3.3 getPassword() `String getPassword ( )`

Gets password.

Returns

the password

1.38.3.4 getProfilePic() `String getProfilePic ( )`

Gets profile pic.

Returns

the profile pic

1.38.3.5 getUsername() `String getUsername ( )`

Gets username.

Returns

the username

1.38.3.6 initDebugLoginValues() `void initDebugLoginValues ( )`

Init debug login values.

1.38.3.7 isGuest() `boolean isGuest ( )`

Is guest boolean.

Returns

the boolean

1.38.3.8 setLoginType() `void setLoginType (`

```
            LoginType loginType )
```

Sets login type.

Parameters

loginType	the login type

1.38.3.9 setPassword()

```
void setPassword (
        String password )
```

Sets password.

Parameters

password	the password

1.38.3.10 setProfilePic()

```
void setProfilePic (
        String profilePic )
```

Sets profile pic.

Parameters

profilePic	the profile pic

1.38.3.11 setUsername()

```
void setUsername (
        String username )
```

Sets username.

Parameters

username	the username

1.38.3.12 toString()

```
String toString ( )
```

To string string.

Returns

the string

1.39 LoginType Enum Reference

Public Member Functions

- boolean **asUser** ()
- String **toString** ()

Public Attributes

- **LOGIN**
- **REGISTER**
- **GUEST**
- **CANCEL**
- **NOT_SET_YET**

1.39.1 Detailed Description

Login Type.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.39.2 Member Function Documentation

1.39.2.1 asUser() `boolean asUser ( )`

As user boolean.

Returns

the boolean

1.39.2.2 toString() `String toString ( )`

To string string.

Returns

the string

1.39.3 Member Data Documentation

1.39.3.1 CANCEL `CANCEL`

Cancel.

1.39.3.2 GUEST `GUEST`

Guest.

1.39.3.3 LOGIN `LOGIN`

Login.

1.39.3.4 NOT_SET_YET `NOT_SET_YET`

Not set yet.

1.39.3.5 REGISTER `REGISTER`

Register.

1.40 Math Enum Reference

Public Member Functions

- **Col execute** (**Col col**, Object value)
- **Col execute** (**Col col**, Object value, boolean changeSelf)
- String **strSource** ()

Static Public Member Functions

- static String **nullIf0** (Object val)
- static String **formatNum** (Object num)
- static String **strVal** (Object val)
- static String **formatNum** (Object num, String format)
- static String **str** (Object obj)
- static String **asFloat** (Object num)

Public Attributes

- **Plus**
- **Mult**
- **Div**
- **Col col**

Protected Member Functions

- void **zeroIfNull** ()
- abstract void **apply** (Object value)
- void **simpleOperation** (Object a, String operation, Object b)

1.40.1 Detailed Description

Math (p. 109) - allows for math actions on columns and some math-related utilities for columns.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.40.2 Member Function Documentation

1.40.2.1 apply() `abstract void apply ( Object value ) [abstract], [protected]`

Apply.

Parameters

value	the value

1.40.2.2 asFloat() `static String asFloat ( Object num ) [static]`

As float string.

Parameters

num	the num

Returns

the string

1.40.2.3 execute() [1/2] `Col execute ( Col col, Object value )`

Execute col.

Parameters

col	the col
value	the value

Returns

the col

1.40.2.4 execute() [2/2]

```
Col execute (
    Col col,
    Object value,
    boolean changeSelf )
```

Execute col.

Parameters

col	the col
value	the value
changeSelf	the change self

Returns

the col

1.40.2.5 formatNum() [1/2]

```
static String formatNum (
    Object num ) [static]
```

Format num string.

Parameters

num	the num

Returns

the string

1.40.2.6 formatNum() [2/2]

```
static String formatNum (
    Object num,
    String format ) [static]
```

Format num string.

Parameters

num	the num
format	the format

Returns

the string

1.40.2.7 nullIf0()

```
static String nullIf0 (
        Object val ) [static]
```

Null if 0 string.

Parameters

val	the val

Returns

the string

1.40.2.8 simpleOperation()

```
void simpleOperation (
        Object a,
        String operation,
        Object b ) [protected]
```

Simple operation.

Parameters

a	the a
operation	the operation
b	the b

1.40.2.9 str()

```
static String str (
        Object obj ) [static]
```

Str string.

Parameters

obj	the obj

Returns

the string

1.40.2.10 strSource() `String strSource ( )`

Str source string.

Returns

the string

1.40.2.11 strVal() `static String strVal ( Object val ) [static]`

Str val string.

Parameters

val	the val

Returns

the string

1.40.2.12 zeroIfNull() `void zeroIfNull ( ) [protected]`

Zero if null.

1.40.3 Member Data Documentation

1.40.3.1 col `Col col`

The **Col** (p. 33).

1.40.3.2 Div `Div`

The Div.

1.40.3.3 Mult `Mult`

The Mult.

1.40.3.4 Plus `Plus`

The Plus.

1.41 Message

Public Member Functions

- **Message** (MessageType messageType)
- **Message** (MessageType messageType, String subject, String respondingToMsgId)
- **Message** (MessageType messageType, **Message** respondingTo)
- **Message** (MessageType messageType, String subject)
- void **setDbRequest** (**DBRequest** dbRequest)
- **MyError** **getError** ()
- void **setError** (**MyError** error)
- **DBRequest** **getDBRequest** ()
- ArrayList< String > **getUsernameSuggestions** ()
- void **setUsernameSuggestions** (ArrayList< String > usernameSuggestions)
- **DBResponse** **getDBResponse** ()
- void **setDBResponse** (**DBResponse** requestedStats)
- Stack< **Move** > **getMoveStack** ()
- void **setMoveStack** (Stack< **Move** > moveStack)
- SyncedItems<?>[ ] **getSyncedLists** ()
- void **setSyncedLists** (SyncedItems<?>... syncedLists)
- String **getRespondingToMsgId** ()
- void **setRespondingToMsgId** (String respondingToMsgId)
- void **setRespondingTo** (**Message** msg)
- boolean **isResponse** ()
- String **getOtherPlayer** ()
- void **setOtherPlayer** (String otherPlayer)
- **Question** **getQuestion** ()
- void **setQuestion** (**Question** question)
- Boolean **getAvailable** ()
- void **setAvailable** (Boolean available)
- String **getUsername** ()
- void **setUsername** (String username)
- GameSettings **getGameSettings** ()
- void **setGameSettings** (GameSettings gameSettings)
- ArrayList< **Move** > **getPreMoves** ()
- void **setPreMoves** (ArrayList< **Move** > preMoves)
- String **getSubject** ()
- **PlayerColor** **getPlayerColor** ()
- void **setPlayerColor** (**PlayerColor** playerColor)
- **Move** **getMove** ()
- void **setMove** (**Move** move)
- GameTime **getGameTime** ()
- void **setGameTime** (GameTime gameTime)
- **GameStatus** **getGameStatus** ()
- void **setGameStatus** (**GameStatus** gameStatus)
- MovesList **getPossibleMoves** ()
- void **setPossibleMoves** (MovesList possibleMoves)
- **Board** **getBoard** ()
- void **setBoard** (**Board** board)
- **LoginInfo** **getLoginInfo** ()
- void **setLoginInfo** (**LoginInfo** loginInfo)
- MessageType **getMessageType** ()
- boolean **isSubject** ()

Static Public Member Functions

- static **Message** **askForLogin** ()
- static **Message** **returnLogin** (**LoginInfo** loginInfo, **Message** respondingTo)
- static **Message** **welcomeMessage** (String str, **LoginInfo** loginInfo)
- static **Message** **initGame** (**Board** board, String opponent, **PlayerColor** player, GameTime gameTime, Stack< **Move** > moveStack)
- static **Message** **throwError** (**MyError** error)
- static **Message** **updateByMove** (**Move** move, GameTime gameTime)
- static **Message** **waitForYourTurn** (String waitingForName, GameTime gameTime)
- static **Message** **waitForMatch** ()
- static **Message** **gameOver** (**GameStatus** gameStatus)
- static **Message** **askForGameSettings** (SyncedItems joinableGames, SyncedItems resumableGames)
- static **Message** **returnGameSettings** (GameSettings gameSettings, **Message** respondingTo)
- static **Message** **interrupt** ()
- static **Message** **bye** (String subject)
- static **Message** **error** (String err)
- static **Message** **returnMove** (**Move** move, **Message** respondingTo)
- static **Message** **answerQuestion** (**Question.Answer** answer, **Message** respondingTo)
- static **Message** **askQuestion** (**Question** question)
- static **Message** **checkUsernameAvailability** (String username)
- static **Message** **returnUsernameNotAvailable** (ArrayList< String > usernameSuggestions, **Message** request)
- static **Message** **returnUsernameAvailable** (**Message** request)
- static **Message** **dbRequest** (**DBRequest** dbRequest)
- static **Message** **returnDBResponse** (**DBResponse** **DBResponse**, **Message** request)
- static **Message** **syncLists** (SyncedItems<?>... syncedLists)

Public Attributes

- final String **messageID**

1.41.1 Detailed Description

1.41.1.1 Message. by Ilan Peretz (`ilanperets@gmail.com`) 10/11/2021

1.41.2 Constructor & Destructor Documentation

1.41.2.1 Message() [1/4] Message (

```
MessageType messageType )
```

Instantiates a new **Message** (p. 114).

Parameters

messageType	the message type

1.41.2.2 Message() [2/4] Message (

```
MessageType messageType,
String subject,
String respondingToMsgId )
```

Instantiates a new **Message** (p. 114).

Parameters

messageType	the message type
subject	the subject
respondingTo↩ MsgId	the responding to msg id

1.41.2.3 Message() [3/4] Message (

```
MessageType messageType,
 Message respondingTo )
```

Instantiates a new **Message** (p. 114).

Parameters

messageType	the message type
respondingTo	the responding to

1.41.2.4 Message() [4/4] Message (

```
MessageType messageType,
String subject )
```

Instantiates a new **Message** (p. 114).

Parameters

messageType	the message type
subject	the subject

1.41.3 Member Function Documentation

1.41.3.1 answerQuestion() `static Message answerQuestion ( Question.Answer answer, Message respondingTo ) [static]`

Answer question message.

Parameters

respondingTo	the responding to

Returns

the message

1.41.3.2 askForGameSettings() `static Message askForGameSettings ( SyncedItems joinableGames, SyncedItems resumableGames ) [static]`

Ask for game settings message.

Parameters

joinableGames	the joinable games
resumableGames	the resumable games

Returns

the message

1.41.3.3 askForLogin() `static Message askForLogin ( ) [static]`

Ask for login message.

Returns

the message

1.41.3.4 askQuestion() `static Message askQuestion ( Question question ) [static]`

Ask question message.

Parameters

question	the question

Returns

the message

1.41.3.5 bye()

`static Message bye ( String subject ) [static]`

Bye message.

Parameters

subject	the subject

Returns

the message

1.41.3.6 checkUsernameAvailability()

`static Message checkUsernameAvailability ( String username ) [static]`

Check username availability message.

Parameters

username	the username

Returns

the message

1.41.3.7 dbRequest()

`static Message dbRequest ( DBRequest dbRequest ) [static]`

Db request message.

Parameters

dbRequest	the db request

Returns

the message

1.41.3.8 error() `static Message error ( String err ) [static]`

Error message.

Parameters

err	the err

Returns

the message

1.41.3.9 gameOver() `static Message gameOver ( GameStatus gameStatus ) [static]`

Game over message.

Parameters

gameStatus	the game status

Returns

the message

1.41.3.10 getAvailable() `Boolean getAvailable ( )`

Gets available.

Returns

the available

1.41.3.11 getBoard() `Board getBoard ( )`

Gets board.

Returns

the board

1.41.3.12 getDBRequest() `DBRequest getDBRequest ( )`

Gets db request.

Returns

the db request

1.41.3.13 getDBResponse() `DBResponse getDBResponse ( )`

Gets db response.

Returns

the db response

1.41.3.14 getError() `MyError getError ( )`

Gets error.

Returns

the error

1.41.3.15 getGameSettings() `GameSettings getGameSettings ( )`

Gets game settings.

Returns

the game settings

1.41.3.16 getGameStatus() **GameStatus** getGameStatus ( )

Gets game status.

Returns

the game status

1.41.3.17 getGameTime() GameTime getGameTime ( )

Gets game time.

Returns

the game time

1.41.3.18 getLoginInfo() **LoginInfo** getLoginInfo ( )

Gets login info.

Returns

the login info

1.41.3.19 getMessageType() MessageType getMessageType ( )

Gets message type.

Returns

the message type

1.41.3.20 getMove() **Move** getMove ( )

Gets move.

Returns

the move

1.41.3.21 getMoveStack() `Stack< Move > getMoveStack ( )`

Gets move stack.

Returns

the move stack

1.41.3.22 getOtherPlayer() `String getOtherPlayer ( )`

Gets other player.

Returns

the other player

1.41.3.23 getPlayerColor() `PlayerColor getPlayerColor ( )`

Gets player color.

Returns

the player color

1.41.3.24 getPossibleMoves() `MovesList getPossibleMoves ( )`

Gets possible moves.

Returns

the possible moves

1.41.3.25 getPreMoves() `ArrayList< Move > getPreMoves ( )`

Gets pre moves.

Returns

the pre moves

1.41.3.26 getQuestion() `Question getQuestion ( )`

Gets question.

Returns

the question

1.41.3.27 getRespondingToMsgId() `String getRespondingToMsgId ( )`

Gets responding to msg id.

Returns

the responding to msg id

1.41.3.28 getSubject() `String getSubject ( )`

Gets subject.

Returns

the subject

1.41.3.29 getSyncedLists() `SyncedItems<?>[ ] getSyncedLists ( )`

Get synced lists synced items [ ].

Returns

the synced items [ ]

1.41.3.30 getUsername() `String getUsername ( )`

Gets username.

Returns

the username

1.41.3.31 getUsernameSuggestions() `ArrayList< String > getUsernameSuggestions ( )`

Gets username suggestions.

Returns

the username suggestions

1.41.3.32 initGame() `static Message initGame (`

```
Board board,
String opponent,
PlayerColor player,
GameTime gameTime,
Stack< Move > moveStack ) [static]
```

Init game message.

Parameters

board	the board
opponent	the opponent
player	the player
gameTime	the game time
moveStack	the move stack

Returns

the message

1.41.3.33 interrupt() `static Message interrupt ( ) [static]`

Interrupt message.

Returns

the message

1.41.3.34 isResponse() `boolean isResponse ( )`

Is response boolean.

Returns

the boolean

1.41.3.35 isSubject() `boolean isSubject ( )`

Is subject boolean.

Returns

the boolean

1.41.3.36 returnDBResponse() `static Message returnDBResponse (`

```
DBResponse DBResponse,
Message request )  [static]
```

Return db response message.

Parameters

DBResponse	the db response
request	the request

Returns

the message

1.41.3.37 returnGameSettings() `static Message returnGameSettings (`

```
GameSettings gameSettings,
Message respondingTo )  [static]
```

Return game settings message.

Parameters

gameSettings	the game settings
respondingTo	the responding to

Returns

the message

1.41.3.38 returnLogin() `static Message returnLogin (`

```
LoginInfo loginInfo,
Message respondingTo )  [static]
```

Return login message.

Parameters

loginInfo	the login info
respondingTo	the responding to

Returns

the message

1.41.3.39 returnMove() `static Message returnMove ( Move move, Message respondingTo ) [static]`

Return move message.

Parameters

move	the move
respondingTo	the responding to

Returns

the message

1.41.3.40 returnUsernameAvailable() `static Message returnUsernameAvailable ( Message request ) [static]`

Return username available message.

Parameters

request	the request

Returns

the message

1.41.3.41 returnUsernameNotAvailable() `static Message returnUsernameNotAvailable ( ArrayList< String > usernameSuggestions, Message request ) [static]`

Return username not available message.

Parameters

usernameSuggestions	the username suggestions
request	the request

Returns

the message

1.41.3.42 setAvailable() `void setAvailable ( Boolean available )`

Sets available.

Parameters

available	the available

1.41.3.43 setBoard() `void setBoard ( Board board )`

Sets board.

Parameters

board	the board

1.41.3.44 setDbRequest() `void setDbRequest ( DBRequest dbRequest )`

Sets db request.

Parameters

dbRequest	the db request

1.41.3.45 setDBResponse() `void setDBResponse ( DBResponse requestedStats )`

Sets db response.

Parameters

requestedStats	the requested stats

1.41.3.46 setError() `void setError ( MyError error )`

Sets error.

Parameters

error	the error

1.41.3.47 setGameSettings() `void setGameSettings ( GameSettings gameSettings )`

Sets game settings.

Parameters

gameSettings	the game settings

1.41.3.48 setGameStatus() `void setGameStatus ( GameStatus gameStatus )`

Sets game status.

Parameters

gameStatus	the game status

1.41.3.49 setGameTime() `void setGameTime ( GameTime gameTime )`

Sets game time.

Parameters

gameTime	the game time

1.41.3.50 setLoginInfo()

```
void setLoginInfo (
    LoginInfo loginInfo )
```

Sets login info.

Parameters

loginInfo	the login info

1.41.3.51 setMove()

```
void setMove (
    Move move )
```

Sets move.

Parameters

move	the move

1.41.3.52 setMoveStack()

```
void setMoveStack (
    Stack< Move > moveStack )
```

Sets move stack.

Parameters

moveStack	the move stack

1.41.3.53 setOtherPlayer()

```
void setOtherPlayer (
    String otherPlayer )
```

Sets other player.

Parameters

otherPlayer	the other player

1.41.3.54 setPlayerColor()

```
void setPlayerColor (
    PlayerColor playerColor )
```

Sets player color.

Parameters

playerColor	the player color

1.41.3.55 setPossibleMoves() `void setPossibleMoves ( MovesList possibleMoves )`

Sets possible moves.

Parameters

possibleMoves	the possible moves

1.41.3.56 setPreMoves() `void setPreMoves ( ArrayList< Move > preMoves )`

Sets pre moves.

Parameters

preMoves	the pre moves

1.41.3.57 setQuestion() `void setQuestion ( Question question )`

Sets question.

Parameters

question	the question

1.41.3.58 setRespondingTo() `void setRespondingTo ( Message msg )`

Sets responding to.

Parameters

msg	the msg

1.41.3.59 setRespondingToMsgId() `void setRespondingToMsgId ( String respondingToMsgId )`

Sets responding to msg id.

Parameters

respondingTo↩ MsgId	the responding to msg id

1.41.3.60 setSyncedLists() `void setSyncedLists ( SyncedItems<?>... syncedLists )`

Sets synced lists.

Parameters

syncedLists	the synced lists

1.41.3.61 setUsername() `void setUsername ( String username )`

Sets username.

Parameters

username	the username

1.41.3.62 setUsernameSuggestions() `void setUsernameSuggestions ( ArrayList< String > usernameSuggestions )`

Sets username suggestions.

Parameters

usernameSuggestions	the username suggestions

1.41.3.63 syncLists() `static Message syncLists ( SyncedItems<?>... syncedLists ) [static]`

Sync lists message.

Parameters

syncedLists	the synced lists

Returns

the message

1.41.3.64 throwError() `static Message throwError ( MyError error ) [static]`

Throw error message.

Parameters

error	if null interrupts

Returns

the message

1.41.3.65 updateByMove() `static Message updateByMove ( Move move, GameTime gameTime ) [static]`

Update by move message.

Parameters

move	the move
gameTime	the game time

Returns

the message

1.41.3.66 waitForMatch() `static Message waitForMatch ( ) [static]`

Wait for match message.

Returns

the message

1.41.3.67 waitForYourTurn() `static Message waitForYourTurn ( String waitingForName, GameTime gameTime ) [static]`

Wait for your turn message.

Parameters

waitingForName	the waiting for name
gameTime	the game time

Returns

the message

1.41.3.68 welcomeMessage() `static Message welcomeMessage ( String str, LoginInfo loginInfo ) [static]`

Welcome message message.

Parameters

str	the str
loginInfo	the login info

Returns

the message

1.41.4 Member Data Documentation

1.41.4.1 messageID `final String messageID`

The **Message** (p. 114) id.

1.42 MessageCallback

Public Member Functions

- void **onMsg** (**Message** message)

1.42.1 Detailed Description

Message callback - a message callback.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.42.2 Member Function Documentation

1.42.2.1 onMsg()

```
void onMsg (
        Message message )
```

On msg.

Parameters

message	the message

1.43 MessagesHandler

Public Member Functions

- **MessagesHandler** (**AppSocket socket**)
- **MessageCallback onCancelQuestion** ()
- void **interruptBlocking** (**MyError** err)
- **Message blockTilRes** (**Message** request)
- void **noBlockRequest** (**Message** request, **MessageCallback** onRes)
- void **receivedMessage** (**Message** message)
- void **prepareForDisconnect** ()
- void **onAnyMsg** (**Message** message)
- final void **onDisconnected** ()
- **MessageCallback onLogin** ()
- **MessageCallback onResign** ()
- **MessageCallback onAddTime** ()
- **MessageCallback onOfferDraw** ()
- **MessageCallback onWelcomeMessage** ()
- **MessageCallback onGetGameSettings** ()
- **MessageCallback onWaitForMatch** ()
- **MessageCallback onInitGame** ()

- **MessageCallback onWaitTurn** ()
- **MessageCallback onGetMove** ()
- **MessageCallback onUpdateByMove** ()
- **MessageCallback onGameOver** ()
- **MessageCallback onError** ()
- **MessageCallback onQuestion** ()
- **MessageCallback onBye** ()
- **MessageCallback onUsernameAvailability** ()
- **MessageCallback onDBRequest** ()
- **MessageCallback onDBResponse** ()
- **MessageCallback onUpdateSyncedList** ()
- **MessageCallback onInterrupt** ()
- **MessageCallback onIsAlive** ()
- **MessageCallback onAlive** ()

Protected Member Functions

- void **onAnyDisconnection** ()
- void **onPlannedDisconnect** ()
- void **onUnplannedDisconnect** ()
- MyError.DisconnectedError **createDisconnectedError** ()

Protected Attributes

- final **AppSocket socket**

1.43.1 Detailed Description

The type Messages handler.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.43.2 Constructor & Destructor Documentation

1.43.2.1 MessagesHandler() `MessagesHandler (`
`AppSocket socket )`

Instantiates a new Messages handler.

Parameters

socket	the socket

1.43.3 Member Function Documentation

1.43.3.1 blockTilRes()

```
Message blockTilRes (
            Message request )
```

Block til res message.

Parameters

request	the request

Returns

the message

1.43.3.2 createDisconnectedError()

```
MyError.DisconnectedError createDisconnectedError ( )  [protected]
```

Create disconnected error my error . disconnected error.

Returns

the my error . disconnected error

1.43.3.3 interruptBlocking()

```
void interruptBlocking (
            MyError err )
```

Interrupt blocking.

Parameters

err	the err

1.43.3.4 noBlockRequest()

```
void noBlockRequest (
            Message request,
            MessageCallback onRes )
```

No block request.

Parameters

request	the request
onRes	the on res

1.43.3.5 onAddTime() `MessageCallback onAddTime ( )`

On add time message callback.

Returns

the message callback

1.43.3.6 onAlive() `MessageCallback onAlive ( )`

On alive message callback.

Returns

the message callback

1.43.3.7 onAnyDisconnection() `void onAnyDisconnection ( ) [protected]`

On any disconnection.

1.43.3.8 onAnyMsg()

```
void onAnyMsg (
            Message message )
```

On any msg.

Parameters

message	the message

1.43.3.9 onBye() `MessageCallback onBye ( )`

On bye message callback.

Returns

the message callback

1.43.3.10 onCancelQuestion() `MessageCallback onCancelQuestion ( )`

On cancel question message callback.

Returns

the message callback

1.43.3.11 onDBRequest() `MessageCallback onDBRequest ( )`

On db request message callback.

Returns

the message callback

1.43.3.12 onDBResponse() `MessageCallback onDBResponse ( )`

On db response message callback.

Returns

the message callback

1.43.3.13 onDisconnected() `final void onDisconnected ( )`

On disconnected.

1.43.3.14 onError() `MessageCallback onError ( )`

On error message callback.

Returns

the message callback

1.43.3.15 onGameOver() `MessageCallback onGameOver ( )`

On game over message callback.

Returns

the message callback

1.43.3.16 onGetGameSettings() `MessageCallback onGetGameSettings ( )`

On get game settings message callback.

Returns

the message callback

1.43.3.17 onGetMove() `MessageCallback onGetMove ( )`

On get move message callback.

Returns

the message callback

1.43.3.18 onInitGame() `MessageCallback onInitGame ( )`

On init game message callback.

Returns

the message callback

1.43.3.19 onInterrupt() `MessageCallback onInterrupt ( )`

On interrupt message callback.

Returns

the message callback

1.43.3.20 onIsAlive() `MessageCallback onIsAlive ( )`

On is alive message callback.

Returns

the message callback

1.43.3.21 onLogin() `MessageCallback onLogin ( )`

On login message callback.

Returns

the message callback

1.43.3.22 onOfferDraw() `MessageCallback onOfferDraw ( )`

On offer draw message callback.

Returns

the message callback

1.43.3.23 onPlannedDisconnect() `void onPlannedDisconnect ( ) [protected]`

On planned disconnect.

1.43.3.24 onQuestion() `MessageCallback onQuestion ( )`

On question message callback.

Returns

the message callback

1.43.3.25 onResign() `MessageCallback onResign ( )`

On resign message callback.

Returns

the message callback

1.43.3.26 onUnplannedDisconnect() `void onUnplannedDisconnect ( ) [protected]`

On unplanned disconnect.

1.43.3.27 onUpdateByMove() `MessageCallback onUpdateByMove ( )`

On update by move message callback.

Returns

the message callback

1.43.3.28 onUpdateSyncedList() `MessageCallback onUpdateSyncedList ( )`

On update synced list message callback.

Returns

the message callback

1.43.3.29 onUsernameAvailability() `MessageCallback onUsernameAvailability ( )`

On username availability message callback.

Returns

the message callback

1.43.3.30 onWaitForMatch() `MessageCallback onWaitForMatch ( )`

On wait for match message callback.

Returns

the message callback

1.43.3.31 onWaitTurn() `MessageCallback onWaitTurn ( )`

On wait turn message callback.

Returns

the message callback

1.43.3.32 onWelcomeMessage() `MessageCallback onWelcomeMessage ( )`

On welcome message message callback.

Returns

the message callback

1.43.3.33 prepareForDisconnect() `void prepareForDisconnect ( )`

Prepare for disconnect.

1.43.3.34 receivedMessage() `void receivedMessage ( Message message )`

Received message.

Parameters

message	the message

1.43.4 Member Data Documentation

1.43.4.1 socket `final` **`AppSocket`** `socket [protected]`

The Socket.

1.44 MinimaxMove

Public Member Functions

- **MinimaxMove** (**Move** move, **Evaluation** moveEvaluation)
- **MinimaxMove** (**Evaluation** moveEvaluation)
- **MinimaxMove** (**MinimaxMove** other)
- boolean **isDeeperAndBetterThan** (**MinimaxMove** other)
- int **getMoveDepth** ()
- **Move** **getMove** ()
- void **setMove** (**Move** move)
- **Evaluation** **getMoveEvaluation** ()
- void **setMoveEvaluation** (**Evaluation** moveEvaluation)
- int **hashCode** ()
- boolean **equals** (Object o)
- String **toString** ()
- String **getShortPrintingStr** ()
- int **compareTo** (**MinimaxMove** o)

1.44.1 Detailed Description

Minimax move.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.44.2 Constructor & Destructor Documentation

1.44.2.1 MinimaxMove() `[1/3]` **`MinimaxMove`** `(`

```
          Move move,
          Evaluation moveEvaluation )
```

Instantiates a new Minimax move.

Parameters

move	the move
moveEvaluation	the move evaluation

1.44.2.2 MinimaxMove() [2/3] `MinimaxMove ( Evaluation moveEvaluation )`

Instantiates a new Minimax move.

Parameters

moveEvaluation	the move evaluation

1.44.2.3 MinimaxMove() [3/3] `MinimaxMove ( MinimaxMove other )`

Instantiates a new Minimax move.

Parameters

other	the other

1.44.3 Member Function Documentation

1.44.3.1 compareTo() `int compareTo ( MinimaxMove o )`

Compare to int.

Parameters

o	the o

Returns

the int

1.44.3.2 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

1.44.3.3 getMove() `Move getMove ( )`

Gets move.

Returns

the move

1.44.3.4 getMoveDepth() `int getMoveDepth ( )`

Gets move depth.

Returns

the move depth

1.44.3.5 getMoveEvaluation() `Evaluation getMoveEvaluation ( )`

Gets move evaluation.

Returns

the move evaluation

1.44.3.6 getShortPrintingStr() `String getShortPrintingStr ( )`

Gets short printing str.

Returns

the short printing str

1.44.3.7 hashCode() `int hashCode ( )`

Hash code int.

Returns

the int

1.44.3.8 isDeeperAndBetterThan() `boolean isDeeperAndBetterThan ( MinimaxMove other )`

Is deeper and better than given minimax move.

Parameters

other	the other

Returns

the boolean

1.44.3.9 setMove() `void setMove ( Move move )`

```
void setMove (
        Move move )
```

Sets move.

Parameters

move	the move

1.44.3.10 setMoveEvaluation()

```
void setMoveEvaluation (
        Evaluation moveEvaluation )
```

Sets move evaluation.

Parameters

moveEvaluation	the move evaluation

1.44.3.11 toString()

```
String toString ( )
```

To string string.

Returns

the string

1.45 Move

Classes

- enum **MoveFlag**
- enum **ThreefoldStatus**

Public Member Functions

- **Move** (**Location** movingFrom, **Location** movingTo, **PieceType** capturingPieceType)
- **Move** (**Location** movingFrom, **Location** movingTo)
- **Move** (**ThreefoldStatus** threefoldStatus)
- **Move** (**Move** other)
- MovesList **getCreatorList** ()
- void **setCreatorList** (MovesList creatorList)
- void **setMoveAnnotation** (String moveAnnotation)
- byte **getDisabledCastling** ()
- void **setDisabledCastling** (byte disabledCastling)
- void **setThreefoldOption** ()
- **PlayerColor getMovingColor** ()
- void **setMovingColor** (**PlayerColor** movingPlayerColor)
- int **getPrevFullMoveClock** ()
- void **setPrevFullMoveClock** (int prevFullMoveClock)
- int **getPrevHalfMoveClock** ()
- void **setPrevHalfMoveClock** (int prevHalfMoveClock)
- **PieceType getPromotingTo** ()
- void **setPromotingTo** (**PieceType** promotingTo)
- **BasicMove getIntermediateMove** ()
- void **setIntermediateMove** (**BasicMove** intermediateMove)
- boolean **isCheck** ()
- **Evaluation getMoveEvaluation** ()
- void **setMoveEvaluation** (**Evaluation** moveEvaluation)
- **PieceType getCapturingPieceType** ()
- boolean **isReversible** ()
- void **setReversible** (boolean reversible)
- boolean **isCapturing** ()
- void **setCapturing** (**PieceType** pieceType)
- boolean **equals** (Object o)
- String **toString** ()
- String **getAnnotation** ()
- boolean **strictEquals** (**Move** move)
- **MoveFlag getMoveFlag** ()
- void **setMoveFlag** (**MoveFlag** moveFlag)
- int **compareTo** (**Move** o)
- **Location getEnPassantLoc** ()
- void **setEnPassantLoc** (**Location** epsnLoc)

Static Public Member Functions

- static **Move castling** (**Location** movingFrom, **Location** movingTo, CastlingRights.Side side)
- static **Move threefoldClaim** ()
- static **Move flipMove** (**Move** move)
- static **Move copyMove** (**Move** move)

1.45.1 Detailed Description

Move (p. 146) - represents a "heavy" move. with a lot of info.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.45.2 Class Documentation

1.45.2.1 enum ver14::SharedClasses::Game::Moves::Move::ThreefoldStatus

Threefold status - represents a threefold draw status.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Enumerator

CAN_CLAIM	Can claim threefold status.
CLAIMED	Claimed threefold status.
NONE	None threefold status.

1.45.3 Constructor & Destructor Documentation

1.45.3.1 Move() [1/4]

```
Move (
        Location movingFrom,
        Location movingTo,
        PieceType capturingPieceType )
```

Instantiates a new **Move** (p. 146).

Parameters

movingFrom	the moving from
movingTo	the moving to
capturingPieceType	the capturing piece type

1.45.3.2 Move() [2/4]

```
Move (
        Location movingFrom,
        Location movingTo )
```

Instantiates a new **Move** (p. 146).

Parameters

movingFrom	the moving from
movingTo	the moving to

1.45.3.3 Move() `[3/4]` `Move ( ThreefoldStatus threefoldStatus )`

Instantiates a new **Move** (p. 146).

Parameters

threefoldStatus	the threefold status

1.45.3.4 Move() `[4/4]` `Move ( Move other )`

Instantiates a new **Move** (p. 146).

Parameters

other	the other

1.45.4 Member Function Documentation

1.45.4.1 castling() `static Move castling ( Location movingFrom, Location movingTo, CastlingRights.Side side ) [static]`

creates a Castling move.

Parameters

movingFrom	the moving from
movingTo	the moving to
side	the castling side

Returns

the move

1.45.4.2 compareTo() `int compareTo ( Move o )`

Compare to another move.

Parameters

o	the other move

Returns

the comparison result

1.45.4.3 copyMove() `static Move copyMove ( Move move ) [static]`

Copy move.

Parameters

move	the move

Returns

the move

1.45.4.4 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

Reimplemented from **BasicMove** (p. 18).

1.45.4.5 flipMove() `static Move flipMove ( Move move ) [static]`

Flip move move.

Parameters

move	the move

Returns

the move

1.45.4.6 getAnnotation() `String getAnnotation ( )`

Gets annotation.

Returns

the annotation

1.45.4.7 getCapturingPieceType() `PieceType getCapturingPieceType ( )`

Gets capturing piece type.

Returns

the capturing piece type

1.45.4.8 getCreatorList() `MovesList getCreatorList ( )`

Gets creator list.

Returns

the creator list

1.45.4.9 getDisabledCastling() `byte getDisabledCastling ( )`

Gets disabled castling.

Returns

the disabled castling

1.45.4.10 getEnPassantLoc() `Location getEnPassantLoc ( )`

Gets en passant loc.

Returns

the en passant loc

1.45.4.11 getIntermediateMove() `BasicMove getIntermediateMove ( )`

Gets intermediate move.

Returns

the intermediate move

1.45.4.12 getMoveEvaluation() `Evaluation getMoveEvaluation ( )`

Gets move evaluation.

Returns

the move evaluation

1.45.4.13 getMoveFlag() `MoveFlag getMoveFlag ( )`

Gets move flag.

Returns

the move flag

1.45.4.14 getMovingColor() `PlayerColor getMovingColor ( )`

Gets moving color.

Returns

the moving color

1.45.4.15 getPrevFullMoveClock() `int getPrevFullMoveClock ( )`

Gets prev full move clock.

Returns

the prev full move clock

1.45.4.16 getPrevHalfMoveClock() `int getPrevHalfMoveClock ( )`

Gets prev half move clock.

Returns

the prev half move clock

1.45.4.17 getPromotingTo() `PieceType getPromotingTo ( )`

Gets promoting to.

Returns

the promoting to

1.45.4.18 isCapturing() `boolean isCapturing ( )`

Is capturing boolean.

Returns

the boolean

1.45.4.19 isCheck() `boolean isCheck ( )`

Is check boolean.

Returns

the boolean

1.45.4.20 isReversible() `boolean isReversible ( )`

Is reversible boolean.

Returns

the boolean

1.45.4.21 setCapturing()

```
void setCapturing (
        PieceType pieceType )
```

Sets capturing.

Parameters

pieceType	the piece type

1.45.4.22 setCreatorList()

```
void setCreatorList (
        MovesList creatorList )
```

Sets creator list.

Parameters

creatorList	the creator list

1.45.4.23 setDisabledCastling()

```
void setDisabledCastling (
        byte disabledCastling )
```

Sets disabled castling.

Parameters

disabledCastling	the disabled castling

1.45.4.24 setEnPassantLoc()

```
void setEnPassantLoc (
        Location epsnLoc )
```

Sets en passant loc.

Parameters

epsnLoc	the epsn loc

1.45.4.25 setIntermediateMove()

```
void setIntermediateMove (
        BasicMove intermediateMove )
```

Sets intermediate move.

Parameters

intermediateMove	the intermediate move

1.45.4.26 setMoveAnnotation() `void setMoveAnnotation ( String moveAnnotation )`

Sets move annotation.

Parameters

moveAnnotation	the move annotation

1.45.4.27 setMoveEvaluation() `void setMoveEvaluation ( Evaluation moveEvaluation )`

Sets move evaluation.

Parameters

moveEvaluation	the move evaluation

1.45.4.28 setMoveFlag() `void setMoveFlag ( MoveFlag moveFlag )`

Sets move flag.

Parameters

moveFlag	the move flag

1.45.4.29 setMovingColor() `void setMovingColor ( PlayerColor movingPlayerColor )`

Sets moving color.

Parameters

movingPlayerColor	the moving player color

1.45.4.30 setPrevFullMoveClock() `void setPrevFullMoveClock ( int prevFullMoveClock )`

Sets prev full move clock.

Parameters

prevFullMoveClock	the prev full move clock

1.45.4.31 setPrevHalfMoveClock() `void setPrevHalfMoveClock ( int prevHalfMoveClock )`

Sets prev half move clock.

Parameters

prevHalfMoveClock	the prev half move clock

1.45.4.32 setPromotingTo() `void setPromotingTo ( PieceType promotingTo )`

Sets promoting to.

Parameters

promotingTo	the promoting to

1.45.4.33 setReversible() `void setReversible ( boolean reversible )`

Sets reversible.

Parameters

reversible	the reversible

1.45.4.34 setThreefoldOption() `void setThreefoldOption ( )`

Sets threefold option.

1.45.4.35 strictEquals() `boolean strictEquals ( Move move )`

checks source and destination equals and move flags equals.

Parameters

move	the move

Returns

the boolean

1.45.4.36 threefoldClaim() `static Move threefoldClaim ( ) [static]`

Threefold claim move.

Returns

the move

1.45.4.37 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented from **BasicMove** (p. 20).

1.46 MoveAnnotation

Static Public Member Functions

- static String **annotate** (**Move** move, **Piece** movingPiece)
- static String **annotate** (**Move** move, **Piece** movingPiece, String unique)
- static String **basicAnnotate** (**BasicMove** move)

1.46.1 Detailed Description

Move (p. 146) annotation - utility class that annotates moves.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.46.2 Member Function Documentation

1.46.2.1 annotate() [1/2]

```
static String annotate (
        Move move,
        Piece movingPiece ) [static]
```

Annotate move.

Parameters

move	the move
movingPiece	the moving piece

Returns

the annotation

1.46.2.2 annotate() [2/2]

```
static String annotate (
    Move move,
    Piece movingPiece,
    String unique ) [static]
```

Annotate move with a unique string.

Parameters

move	the move
movingPiece	the moving piece
unique	the unique string

Returns

the string

1.46.2.3 basicAnnotate()

```
static String basicAnnotate (
    BasicMove move ) [static]
```

Basic annotate a move. just the source and destination.

Parameters

move	the move

Returns

the string

1.47 Move.MoveFlag Enum Reference

Public Member Functions

- **MoveFlag** ()
- **MoveFlag** (CastlingRights.Side side)
- boolean **equals** (**Move** myMove, **Move** otherMove)

Public Attributes

- **NormalMove**
- **EnPassant**
- **DoublePawnPush**
- **Promotion**
- **ShortCastle**
- **LongCastle**
- final boolean **isCastling**
- final CastlingRights.Side **castlingSide**

Static Public Attributes

- static final **MoveFlag**[ ] **CASTLING_FLAGS**

1.47.1 Detailed Description

Move (p. 146) flag - which type of move this is.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.47.2 Constructor & Destructor Documentation

1.47.2.1 MoveFlag() `[1/2]` `MoveFlag ( )`

Instantiates a new **Move** (p. 146) flag.

1.47.2.2 MoveFlag() `[2/2]` `MoveFlag (`
`CastlingRights.Side` *side* `)`

Instantiates a new **Move** (p. 146) flag.

Parameters

side	the side

1.47.3 Member Function Documentation

1.47.3.1 equals() `boolean equals ( Move myMove, Move otherMove )`

```
boolean equals (
    Move myMove,
    Move otherMove )
```

checks if the flags are equals.

Parameters

myMove	the my move
otherMove	the other move

Returns

the boolean

1.47.4 Member Data Documentation

1.47.4.1 CASTLING_FLAGS `final MoveFlag [ ] CASTLING_FLAGS [static]`

The Castling flags.

1.47.4.2 castlingSide `final CastlingRights.Side castlingSide`

if this is a castling flag, which side is it.

1.47.4.3 DoublePawnPush `DoublePawnPush`

Double pawn push move flag.

1.47.4.4 EnPassant `EnPassant`

En passant move flag.

1.47.4.5 isCastling `final boolean isCastling`

Is this move flag a castling flag.

1.47.4.6 LongCastle `LongCastle`

Long castle move flag.

1.47.4.7 NormalMove `NormalMove`

Normal move move flag.

1.47.4.8 Promotion `Promotion`

The Promotion.

1.47.4.9 ShortCastle `ShortCastle`

Short castle move flag.

1.48 MyError

Public Member Functions

- **MyError** ()
- **MyError** (Throwable throwable)
- **MyError** (String message)
- **MyError** (String message, Throwable cause)
- String **getHandledStr** ()
- String **toString** ()

Static Public Member Functions

- static String **errToString** (Throwable error)

1.48.1 Detailed Description

My error - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.48.2 Constructor & Destructor Documentation

1.48.2.1 MyError() [1/4] `MyError ( )`

Instantiates a new My error.

1.48.2.2 MyError() [2/4] `MyError (`

```
Throwable throwable )
```

Instantiates a new My error.

Parameters

throwable	the throwable

1.48.2.3 MyError() [3/4] MyError (String *message*)

Instantiates a new My error.

Parameters

message	the message

1.48.2.4 MyError() [4/4] MyError (String *message*, Throwable *cause*)

Instantiates a new My error.

Parameters

message	the message
cause	the cause

1.48.3 Member Function Documentation

1.48.3.1 errToString() static String errToString (Throwable *error*) [static]

Err to string string.

Parameters

error	the error

Returns

the string

1.48.3.2 getHandledStr() String getHandledStr ( )

Gets handled str.

Returns

the handled str

1.48.3.3 toString() `String toString ( )`

To string string.

Returns

the string

1.49 MyJButton

Public Member Functions

- **MyJButton** (String text, **VoidCallback** callback)
- **MyJButton** (String text)
- void **setOnClick** (**VoidCallback** onClick)
- void **setText** (String text)
- **MyJButton** ()
- **MyJButton** (String text, Font font, **VoidCallback** onClick, JPanel addTo)
- **MyJButton** (String text, Font font, **VoidCallback** onClick)
- **MyJButton** (String text, Font font)
- void **setFont** (Font font)
- int **getMinSize** ()
- void **replaceWithCancel** (**VoidCallback** onCancelled)
- void **resetState** (boolean e)

1.49.1 Detailed Description

MyJButton- represents a button.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.49.2 Constructor & Destructor Documentation

1.49.2.1 MyJButton() [1/6]

```
MyJButton (
        String text,
        VoidCallback callback )
```

Instantiates a new My j button.

Parameters

text	the text
callback	the callback

1.49.2.2 MyJButton() `[2/6]` `MyJButton ( String text )`

Instantiates a new My j button.

Parameters

text	the text

1.49.2.3 MyJButton() `[3/6]` `MyJButton ( )`

Instantiates a new My j button.

1.49.2.4 MyJButton() `[4/6]`

```
MyJButton (
        String text,
        Font font,
        VoidCallback onClick,
        JPanel addTo )
```

Instantiates a new My j button.

Parameters

text	the text
font	the font
onClick	the on click
addTo	the add to

1.49.2.5 MyJButton() `[5/6]`

```
MyJButton (
        String text,
        Font font,
        VoidCallback onClick )
```

Instantiates a new My j button.

Parameters

text	the text
font	the font
onClick	the on click

1.49.2.6 MyJButton() [6/6] MyJButton (

```
String text,
Font font )
```

Instantiates a new My j button.

Parameters

text	the text
font	the font

1.49.3 Member Function Documentation

1.49.3.1 getMinSize() int getMinSize ( )

Gets min size.

Returns

the min size

1.49.3.2 replaceWithCancel() void replaceWithCancel (

```
VoidCallback onCancelled )
```

Replace with cancel.

Parameters

onCancelled	the on cancelled

1.49.3.3 resetState() void resetState (

```
boolean e )
```

Reset state.

Parameters

e	the e

1.49.3.4 setFont() `void setFont ( Font font )`

Sets font.

Parameters

font	the font

1.49.3.5 setOnClick() `void setOnClick ( VoidCallback onClick )`

Sets on click.

Parameters

onClick	the on click

1.49.3.6 setText() `void setText ( String text )`

Sets text.

Parameters

text	the text

1.50 MyJFrame

Classes

- interface **BooleanClosing**
- interface **Closing**
- interface **StringClosing**

Public Member Functions

- **MyJFrame** () throws HeadlessException
- MyAdapter **getMyAdapter** ()
- void **setOnExit** (**Closing**<?> **onClose**)
- void **doXClick** ()
- void **setOnResize** (**VoidCallback** onResize)

Static Public Member Functions

- static MyAdapter **debugAdapter** (Window addTo)

Protected Attributes

- **Closing**<?> **onClose**

1.50.1 Detailed Description

The type My j frame.

1.50.2 Constructor & Destructor Documentation

1.50.2.1 MyJFrame() `MyJFrame ( ) throws HeadlessException`

Instantiates a new My j frame.

Exceptions

HeadlessException	the headless exception

1.50.3 Member Function Documentation

1.50.3.1 debugAdapter() `static MyAdapter debugAdapter ( Window addTo ) [static]`

Debug adapter my adapter.

Parameters

addTo	the add to

Returns

the my adapter

1.50.3.2 doXClick() `void doXClick ( )`

Do x click.

1.50.3.3 getMyAdapter() `MyAdapter getMyAdapter ( )`

Gets my adapter.

Returns

the my adapter

1.50.3.4 setOnExit() `void setOnExit ( Closing<?> onClose )`

Sets on exit.

Parameters

onClose	the on close

1.50.3.5 setOnResize() `void setOnResize ( VoidCallback onResize )`

Sets on resize.

Parameters

onResize	the on resize

1.50.4 Member Data Documentation

1.50.4.1 onClose `Closing<?> onClose [protected]`

The On close.

1.51 MyThread

Public Member Functions

- **MyThread** ()
- void **stopRun** ()
- void **ignoreErrs** ()
- void **reactivateErrs** ()

Static Public Member Functions

- static void **currentThread** (**Callback**< **MyThread** > run)

Protected Member Functions

- abstract void **handledRun** () throws Throwable

1.51.1 Detailed Description

The type My thread.

1.51.2 Constructor & Destructor Documentation

1.51.2.1 MyThread() `MyThread ( )`

Instantiates a new My thread.

1.51.3 Member Function Documentation

1.51.3.1 currentThread() `static void currentThread ( Callback< MyThread > run ) [static]`

Current thread. will only execute code if inside a **MyThread** (p. 168)

Parameters

run	the run

1.51.3.2 handledRun() `abstract void handledRun ( ) throws Throwable [abstract], [protected]`

Handled run.

Exceptions

Throwable	the throwable

Reimplemented in **AppSocket** (p. 4), and **HandledThread** (p. 94).

1.51.3.3 ignoreErrs() `void ignoreErrs ( )`

Ignore errs.

1.51.3.4 reactivateErrs() `void reactivateErrs ( )`

Reactivate errs.

1.51.3.5 stopRun() `void stopRun ( )`

Stop run.

1.52 ObjCallback< T >

1.52.1 Detailed Description

Parameters

<*T*>	the type

1.53 Piece Enum Reference

Public Member Functions

- String **getPieceIcon** ()
- String **getFen** ()
- boolean **isOnMyTeam** (**Piece** otherPiece)
- boolean **isOnMyTeam** (**PlayerColor** otherPlayerColor)

Static Public Member Functions

- static **Piece** **getPieceFromFen** (char c)
- static **Piece** **getPiece** (**PieceType** **pieceType**, **PlayerColor** **playerColor**)
- static String **getPieceIcon** (**Piece** piece)

Public Attributes

- **W_P**
- **W_R**
- **W_B**
- **W_N**
- **W_Q**
- **W_K**
- **B_P**

- **B_R**
- **B_B**
- **B_N**
- **B_Q**
- **B_K**
- final **PieceType** **pieceType**
- final **PlayerColor** **playerColor**

Static Public Attributes

- static final **Piece**[ ] **ALL_PIECES**

1.53.1 Detailed Description

Piece (p. 170) - represents a combination of a piece type, and a color.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

See also

PieceType (p. 175)

PlayerColor (p. 181)

1.53.2 Member Function Documentation

1.53.2.1 getFen() `String getFen ( )`

Gets fen.

Returns

the fen

1.53.2.2 getPiece() `static Piece getPiece (`

```
PieceType pieceType,
PlayerColor playerColor )  [static]
```

Gets piece.

Parameters

pieceType	the piece type
playerColor	the player color

Returns

the piece

1.53.2.3 getPieceFromFen() `static Piece getPieceFromFen ( char c ) [static]`

Gets piece from fen.

Parameters

c	the c

Returns

the piece from fen

1.53.2.4 getPieceIcon() `[1/2]` `String getPieceIcon ( )`

Gets piece icon.

Returns

the piece icon

1.53.2.5 getPieceIcon() `[2/2]` `static String getPieceIcon ( Piece piece ) [static]`

Gets piece icon.

Parameters

piece	the piece

Returns

the piece icon

1.53.2.6 isOnMyTeam() `[1/2]` `boolean isOnMyTeam ( Piece otherPiece )`

Is on my team boolean.

Parameters

otherPiece	the other piece

Returns

the boolean

1.53.2.7 isOnMyTeam() [2/2] `boolean isOnMyTeam ( PlayerColor otherPlayerColor )`

Is on my team boolean.

Parameters

otherPlayerColor	the other player color

Returns

the boolean

1.53.3 Member Data Documentation

1.53.3.1 ALL_PIECES `final Piece [] ALL_PIECES [static]`

The constant ALL_PIECES.

1.53.3.2 B_B `B_B`

represents a Black Bishop.

1.53.3.3 B_K `B_K`

represents a Black King.

1.53.3.4 B_N `B_N`

represents a Black Knight.

1.53.3.5 B_P `B_P`

represents a Black p piece.

1.53.3.6 B_Q `B_Q`

represents a Black Queen.

1.53.3.7 B_R `B_R`

represents a Black Rook.

1.53.3.8 pieceType `final PieceType pieceType`

The **Piece** (p. 170) type.

1.53.3.9 playerColor `final PlayerColor playerColor`

The Player color.

1.53.3.10 W_B `W_B`

represents a White Bishop.

1.53.3.11 W_K `W_K`

represents a White King.

1.53.3.12 W_N `W_N`

represents a White Knight.

1.53.3.13 W_P `W_P`

represents a White p piece.

1.53.3.14 W_Q `W_Q`

represents a White Queen.

1.53.3.15 W_R `W_R`

represents a White Rook.

1.54 PieceType Enum Reference

Public Member Functions

- String **getWhitePieceFen** ()
- String **getPieceIcon** (**PlayerColor** playerColor)
- boolean **compareMovementType** (**PieceType** compareTo)
- String **getPieceName** ()
- boolean **isAttack** (**Direction** direction, int maxDistance)
- **Direction**[ ] **getAttackingDirections** ()
- **Direction**[ ] **getWalkingDirections** ()

Static Public Member Functions

- static **PieceType** **getPieceType** (int pieceType)
- static **Direction**[ ] **getAttackingDirections** (**PieceType** pieceType)
- static boolean **compareMovementType** (**PieceType** piece1Type, **PieceType** piece2Type)
- static boolean **isDiagonalPiece** (**PieceType** pieceType)
- static boolean **isLinePiece** (**PieceType** pieceType)

Public Attributes

- **PAWN**
- **ROOK**
- **BISHOP**
- **KNIGHT**
- **QUEEN**
- **KING**
- final String **whiteIcon**
- final String **blackIcon**
- final int **value**
- final boolean **isSliding**
- final int **asInt**

Static Public Attributes

- static final int **NUM_OF_PIECE_TYPES**
- static final **PieceType**[ ] **PIECE_TYPES**
- static final **PieceType**[ ] **UNIQUE_MOVES_PIECE_TYPES**
- static final **PieceType**[ ] **MINOR_PIECES**
- static final **PieceType**[ ] **MAJOR_PIECES**
- static final **PieceType**[ ] **CAN_PROMOTE_TO**
- static final **PieceType**[ ] **ATTACKING_PIECE_TYPES**
- static final **Direction**[ ][ ] **ATTACKING_DIRECTIONS**

1.54.1 Detailed Description

Piece (p. 170) type - represents the **Piece** (p. 170) Type .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.54.2 Member Function Documentation

1.54.2.1 compareMovementType() [1/2]

```
boolean compareMovementType (
            PieceType compareTo )
```

Compare movement type boolean.

Parameters

compareTo	the compare to

Returns

the boolean

1.54.2.2 compareMovementType() [2/2]

```
static boolean compareMovementType (
            PieceType piece1Type,
            PieceType piece2Type ) [static]
```

Compare movement type boolean.

Parameters

piece1Type	the piece 1 type
piece2Type	the piece 2 type

Returns

the boolean

1.54.2.3 getAttackingDirections() [1/2]

```
Direction[] getAttackingDirections ( )
```

Get attacking directions direction [ ].

Returns

the direction [ ]

1.54.2.4 getAttackingDirections() [2/2]

```
static Direction[] getAttackingDirections (
            PieceType pieceType ) [static]
```

Get attacking directions direction [ ].

Parameters

pieceType	the piece type

Returns

the direction [ ]

1.54.2.5 getPieceIcon() `String getPieceIcon ( PlayerColor playerColor )`

Gets piece icon.

Parameters

playerColor	the player color

Returns

the piece icon

1.54.2.6 getPieceName() `String getPieceName ( )`

Gets piece name.

Returns

the piece name

1.54.2.7 getPieceType() `static PieceType getPieceType ( int pieceType ) [static]`

Gets piece type.

Parameters

pieceType	the piece type

Returns

the piece type

1.54.2.8 getWalkingDirections() `Direction[] getWalkingDirections ( )`

Get walking directions direction [ ].

Returns

the direction [ ]

1.54.2.9 getWhitePieceFen() `String getWhitePieceFen ( )`

Gets white piece fen.

Returns

the white piece fen

1.54.2.10 isAttack()

```
boolean isAttack (
    Direction direction,
    int maxDistance )
```

Is attack boolean.

Parameters

direction	the direction
maxDistance	the max distance

Returns

the boolean

1.54.2.11 isDiagonalPiece()

```
static boolean isDiagonalPiece (
    PieceType pieceType ) [static]
```

Is diagonal piece boolean.

Parameters

pieceType	the piece type

Returns

the boolean

1.54.2.12 isLinePiece() `static boolean isLinePiece ( PieceType pieceType ) [static]`

Is line piece boolean.

Parameters

pieceType	the piece type

Returns

the boolean

1.54.3 Member Data Documentation

1.54.3.1 asInt `final int asInt`

The As int.

1.54.3.2 ATTACKING_DIRECTIONS `final Direction [ ][ ] ATTACKING_DIRECTIONS [static]`

The Attacking directions. each piece type has its own set of attacking direction.

1.54.3.3 ATTACKING_PIECE_TYPES `final PieceType [ ] ATTACKING_PIECE_TYPES [static]`

The constant ATTACKING_PIECE_TYPES.

1.54.3.4 BISHOP `BISHOP`

Bishop piece type.

1.54.3.5 blackIcon `final String blackIcon`

The Black icon.

1.54.3.6 CAN_PROMOTE_TO `final PieceType [ ] CAN_PROMOTE_TO [static]`

The types of pieces a pawn Can promote to.

1.54.3.7 isSliding `final boolean isSliding`

The Is sliding.

1.54.3.8 KING `KING`

The King.

1.54.3.9 KNIGHT `KNIGHT`

The Knight.

1.54.3.10 MAJOR_PIECES `final PieceType [ ] MAJOR_PIECES [static]`

The Major pieces.

1.54.3.11 MINOR_PIECES `final PieceType [ ] MINOR_PIECES [static]`

The Minor pieces.

1.54.3.12 NUM_OF_PIECE_TYPES `final int NUM_OF_PIECE_TYPES [static]`

The constant NUM_OF_PIECE_TYPES.

1.54.3.13 PAWN `PAWN`

Pawn **Piece** (p. 170) Type.

1.54.3.14 PIECE_TYPES `final PieceType [ ] PIECE_TYPES [static]`

The constant PIECE_TYPES.

1.54.3.15 QUEEN `QUEEN`

Queen piece type.

1.54.3.16 ROOK `ROOK`

Rook piece type.

1.54.3.17 UNIQUE_MOVES_PIECE_TYPES `final PieceType [ ] UNIQUE_MOVES_PIECE_TYPES [static]`

The Unique moves piece types.

1.54.3.18 value `final int value`

The Value.

1.54.3.19 whiteIcon `final String whiteIcon`

The White icon.

1.55 PlayerColor Enum Reference

Public Member Functions

- **PlayerColor** (int **startingRow**, int **diff**)
- abstract **PlayerColor getOpponent** ()
- String **toString** ()
- String **getName** ()

Static Public Member Functions

- static **PlayerColor getColor** (int clr)
- static **PlayerColor getPlayerFromFen** (String playerToMove)

Public Attributes

- **WHITE**
- **BLACK**
- **NO_PLAYER**
- final int **asInt**
- final int **indexOf2**
- final int **startingRow**
- final int **diff**

Static Public Attributes

- static final **PlayerColor**[ ] **PLAYER_COLORS**
- static final int **NUM_OF_PLAYERS**

1.55.1 Detailed Description

Player color - represents a player color.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.55.2 Constructor & Destructor Documentation

1.55.2.1 PlayerColor()

```
PlayerColor (
        int startingRow,
        int diff )
```

Instantiates a new Player color.

Parameters

startingRow	the starting row
diff	the diff

1.55.3 Member Function Documentation

1.55.3.1 getColor()

```
static PlayerColor getColor (
        int clr ) [static]
```

Gets color.

Parameters

clr	the clr

Returns

the color

1.55.3.2 getName()

```
String getName ( )
```

Gets name.

Returns

the name

1.55.3.3 getOpponent()

```
abstract PlayerColor getOpponent ( ) [abstract]
```

Gets opponent.

Returns

the opponent

1.55.3.4 getPlayerFromFen()

```
static PlayerColor getPlayerFromFen (
        String playerToMove ) [static]
```

Gets player from fen.

Parameters

playerToMove	the player to move

Returns

the player from fen

1.55.3.5 toString() `String toString ( )`

To string string.

Returns

the string

1.55.4 Member Data Documentation

1.55.4.1 asInt `final int asInt`

The As int.

1.55.4.2 BLACK `BLACK`

Black.

1.55.4.3 diff `final int diff`

The moving up ratio.

1.55.4.4 indexOf2 `final int indexOf2`

index of two. calculated at initialization for performance. calculation: **`asInt`** (p. 183)*2

1.55.4.5 NO_PLAYER `NO_PLAYER`

No player.

1.55.4.6 NUM_OF_PLAYERS `final int NUM_OF_PLAYERS [static]`

The constant NUM_OF_PLAYERS.

1.55.4.7 PLAYER_COLORS `final PlayerColor [] PLAYER_COLORS [static]`

The constant PLAYER_COLORS.

1.55.4.8 startingRow `final int startingRow`

The player's Starting row.

1.55.4.9 WHITE `WHITE`

White.

1.56 PreMadeRequest

Classes

- interface **VariationCreator**

Public Member Functions

- **PreMadeRequest**[ ] **getRequestVariations** ()
- **RequestBuilder** **createBuilder** ()

Public Attributes

- final int **authSettings**

Static Public Attributes

- static final **PreMadeRequest** **TopPlayers**
- static final **PreMadeRequest** **Games**
- static final **PreMadeRequest** **DeleteUnfGames**
- static final **PreMadeRequest** **StatsByTimeOfDay**
- static final **PreMadeRequest** **ChangeProfilePic**
- static final **PreMadeRequest**[ ] **statistics**

1.56.1 Detailed Description

Pre made request - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.56.2 Member Function Documentation

1.56.2.1 createBuilder() `RequestBuilder createBuilder ( )`

Create builder request builder.

Returns

the request builder

1.56.2.2 getRequestVariations() `PreMadeRequest[] getRequestVariations ( )`

Get request variations pre made request [ ].

Returns

the pre made request [ ]

1.56.3 Member Data Documentation

1.56.3.1 authSettings `final int authSettings`

The Auth settings.

1.56.3.2 ChangeProfilePic `final PreMadeRequest ChangeProfilePic [static]`

The constant ChangeProfilePic.

1.56.3.3 DeleteUnfGames `final PreMadeRequest DeleteUnfGames [static]`

The constant DeleteUnfGames.

1.56.3.4 Games `final PreMadeRequest Games [static]`

The constant Games.

1.56.3.5 statistics `final PreMadeRequest [ ] statistics [static]`

The Statistics.

1.56.3.6 StatsByTimeOfDay `final PreMadeRequest StatsByTimeOfDay [static]`

The constant StatsByTimeOfDay.

1.56.3.7 TopPlayers `final` **`PreMadeRequest`** `TopPlayers [static]`

The constant TopPlayers.

1.57 Question

Classes

- enum **QuestionType**

Public Member Functions

- **Question** (String **questionStr**, Answer... possibleAnswers)
- **Question** (String **questionStr**, **QuestionType questionType**, Answer... possibleAnswers)
- **Answer getDefaultAnswer** ()
- void **setDefaultAnswer** (**Answer** defaultAnswer)
- String **getQuestionStr** ()
- **Answer**[ ] **getPossibleAnswers** ()
- boolean **equals** (Object o)
- String **toString** ()
- record **Answer** (String answerStr) implements Serializable

Static Public Member Functions

- static **Question drawOffer** (String offeringPlayer)

Public Attributes

- final String **questionStr**
- final **QuestionType questionType**

Static Public Attributes

- static final **Question Threefold**
- static final **Question Rematch**

1.57.1 Detailed Description

Question (p. 186).

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.57.2 Class Documentation

1.57.2.1 enum ver14::SharedClasses::Misc::Question::QuestionType **Question** (p. 186) type.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Enumerator

DRAW_OFFER	Draw offer question type.
NO_TYPE	No type question type.
REMATCH	Rematch question type.
THREEFOLD	Threefold question type.

1.57.3 Constructor & Destructor Documentation

1.57.3.1 Question() [1/2]

```
Question (
        String questionStr,
        Answer...  possibleAnswers )
```

Instantiates a new **Question** (p. 186).

Parameters

questionStr	the question str
possibleAnswers	the possible answers

1.57.3.2 Question() [2/2]

```
Question (
        String questionStr,
         QuestionType questionType,
        Answer...  possibleAnswers )
```

Instantiates a new **Question** (p. 186).

Parameters

questionStr	the question str
questionType	the question type
possibleAnswers	the possible answers

1.57.4 Member Function Documentation

1.57.4.1 Answer()

```
record Answer (
        String answerStr )
```

Answer.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.57.4.2 drawOffer() `static Question drawOffer ( String offeringPlayer ) [static]`

Draw offer question.

Parameters

offeringPlayer	the offering player

Returns

the question

1.57.4.3 equals() `boolean equals ( Object o )`

Equals boolean.

Parameters

o	the o

Returns

the boolean

1.57.4.4 getDefaultAnswer() `Answer getDefaultAnswer ( )`

Gets default answer.

Returns

the default answer

1.57.4.5 getPossibleAnswers() `Answer[] getPossibleAnswers ( )`

Get possible answers answer [ ].

Returns

the answer [ ]

1.57.4.6 getQuestionStr() `String getQuestionStr ( )`

Gets question str.

Returns

the question str

1.57.4.7 setDefaultAnswer() `void setDefaultAnswer ( Answer defaultAnswer )`

Sets default answer.

Parameters

defaultAnswer	the default answer

1.57.4.8 toString() `String toString ( )`

To string string.

Returns

the string

1.57.5 Member Data Documentation

1.57.5.1 questionStr `final String questionStr`

The **Question** (p. 186) str.

1.57.5.2 questionType `final QuestionType questionType`

The **Question** (p. 186) type.

1.57.5.3 Rematch `final Question Rematch [static]`

The constant Rematch.

1.57.5.4 Threefold `final Question Threefold [static]`

The constant Threefold.

1.58 RegEx

Public Member Functions

- **RegEx** (@Language("RegExp") String regEx, String details, String... **dontMatch**)
- **RegEx** (@Language("RegExp") String regEx, String details, boolean useDontMatch, String... **dontMatch**)
- **RegEx** **canBeEmpty** (boolean bool, String emptyDetails)
- **RegEx** **get** (boolean useDontMatch)
- void **setUseDontMatch** (boolean useDontMatch)
- String **getDetails** ()
- boolean **check** (String str)
- String **getRegex** ()

Static Public Member Functions

- static boolean **isSavedDate** (String str)

Public Attributes

- final String[ ] **dontMatch**

Static Public Attributes

- static final **RegEx** **Fen**
- static final **RegEx** **Username**
- static final **RegEx** **Password**
- static final **RegEx** **Icon**
- static final **RegEx** **StrUtilSkip**
- static final **RegEx** **Numbers**
- static final **RegEx** **URL**
- static final **RegEx** **DontSaveGame**
- static final **RegEx** **IPPAddress**
- static final **RegEx** **Any**

1.58.1 Detailed Description

Reg ex.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.58.2 Constructor & Destructor Documentation

1.58.2.1 RegEx() [1/2]

```
RegEx (
        @Language("RegExp") String regEx,
        String details,
        String...  dontMatch )
```

Instantiates a new Reg ex.

Parameters

regEx	the reg ex
details	the details
dontMatch	the dont match

1.58.2.2 RegEx() [2/2]

```
RegEx (
        @Language("RegExp") String regEx,
        String details,
        boolean useDontMatch,
        String...  dontMatch )
```

Instantiates a new Reg ex.

Parameters

regEx	the reg ex
details	the details
useDontMatch	the use dont match
dontMatch	the dont match

1.58.3 Member Function Documentation

1.58.3.1 canBeEmpty() **RegEx** canBeEmpty (boolean bool , String *emptyDetails*)

Can be empty reg ex.

Parameters

bool	the bool
emptyDetails	the empty details

Returns

the reg ex

1.58.3.2 check() boolean check (String *str*)

Check boolean.

Parameters

str	the str

Returns

the boolean

1.58.3.3 get() **RegEx** get (boolean *useDontMatch*)

Get reg ex.

Parameters

useDontMatch	the use dont match

Returns

the reg ex

1.58.3.4 getDetails() String getDetails ( )

Gets details.

Returns

the details

1.58.3.5 getRegex() `String getRegex ( )`

Gets regex.

Returns

the regex

1.58.3.6 isSavedDate() `static boolean isSavedDate ( String str ) [static]`

Is saved date boolean.

Parameters

str	the str

Returns

the boolean

1.58.3.7 setUseDontMatch() `void setUseDontMatch ( boolean useDontMatch )`

Sets use dont match.

Parameters

useDontMatch	the use dont match

1.58.4 Member Data Documentation

1.58.4.1 Any `final RegEx Any [static]`

The constant Any.

1.58.4.2 dontMatch `final String [] dontMatch`

The Dont match.

1.58.4.3 DontSaveGame `final RegEx DontSaveGame [static]`

The constant DontSaveGame.

1.58.4.4 Fen `final RegEx Fen [static]`

The constant Fen.

1.58.4.5 Icon `final RegEx Icon [static]`

The constant Icon.

1.58.4.6 IPPAddress `final RegEx IPPAddress [static]`

The constant IPPAddress.

1.58.4.7 Numbers `final RegEx Numbers [static]`

The constant Numbers.

1.58.4.8 Password `final RegEx Password [static]`

The constant Password.

1.58.4.9 StrUtilSkip `final RegEx StrUtilSkip [static]`

The constant StrUtilSkip.

1.58.4.10 URL `final RegEx URL [static]`

The constant URL.

1.58.4.11 Username `final RegEx Username [static]`

The constant Username.

1.59 RequestBuilder

Public Member Functions

- **RequestBuilder** (**DBRequest** request, PreMadeRequest.Variation variation)
- **RequestBuilder** (**SQLStatement statement**, String **name**, Arg... **args**)
- **RequestBuilder** (**SQLStatement statement**, String **name**, String desc, Arg... **args**)
- **RequestBuilder** (**SQLStatement statement**, String **name**, String **postDescription**, String **preDescription**, Arg... **args**)
- void **setSubBuilder** (**RequestBuilder subBuilder**)
- ArrayList< SyncedListType > **getShouldSync** ()
- String **getArgVal** (int index)
- **DBResponse createResponse** (ResultSet rs, **DBRequest** request)
- String **getPreDescription** ()
- String **getPostDescription** ()
- String **getName** ()
- **Arg**[ ] **getArgs** ()
- **DBRequest build** (Object... argsVals)

Static Public Member Functions

- static **RequestBuilder createVariation** (**ObjCallback**< **RequestBuilder** > og, PreMadeRequest.VariationCreator variationCreator)
- static **RequestBuilder changePassword** ()
- static **RequestBuilder changeProfilePic** ()
- static **RequestBuilder deleteAllUnFinishedGames** ()
- static **RequestBuilder games** ()
- static **RequestBuilder statsByTimeOfDay** ()
- static **RequestBuilder top** ()

Public Attributes

- final **Arg**[ ] **args**

Static Public Attributes

- static final String **TIE_STR**

Protected Member Functions

- void **addShouldSync** (SyncedListType listType)

Protected Attributes

- final **SQLStatement statement**
- final String **name**
- String **postDescription**
- String **preDescription**
- **RequestBuilder subBuilder**

1.59.1 Detailed Description

Request builder - creates builders capable of generating complete sql statements. after building with the required arguments .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.59.2 Constructor & Destructor Documentation

1.59.2.1 RequestBuilder() [1/4]

```
RequestBuilder (
    DBRequest request,
    PreMadeRequest.Variation variation )
```

Instantiates a new Request builder.

Parameters

request	the request
variation	the variation

1.59.2.2 RequestBuilder() [2/4]

```
RequestBuilder (
    SQLStatement statement,
    String name,
    Arg... args )
```

Instantiates a new Request builder.

Parameters

statement	the statement
name	the name
args	the args

1.59.2.3 RequestBuilder() [3/4]

```
RequestBuilder (
    SQLStatement statement,
    String name,
    String desc,
    Arg... args )
```

Instantiates a new Request builder.

Parameters

statement	the statement
name	the name
desc	the desc
args	the args

1.59.2.4 RequestBuilder() [4/4]

```
RequestBuilder (
        SQLStatement statement,
        String name,
        String postDescription,
        String preDescription,
        Arg... args )
```

Instantiates a new Request builder.

Parameters

statement	the statement
name	the name
postDescription	the post description
preDescription	the pre description
args	the args

1.59.3 Member Function Documentation

1.59.3.1 addShouldSync()

```
void addShouldSync (
        SyncedListType listType ) [protected]
```

Add should sync.

Parameters

listType	the list type

1.59.3.2 build()

```
DBRequest build (
        Object... argsVals )
```

Build db request.

Parameters

argsVals	the args vals

Returns

the db request

1.59.3.3 changePassword() `static RequestBuilder changePassword ( ) [static]`

Change password request builder.

Returns

the request builder

1.59.3.4 changeProfilePic() `static RequestBuilder changeProfilePic ( ) [static]`

Change profile pic request builder.

Returns

the request builder

1.59.3.5 createResponse()

```
DBResponse createResponse (
        ResultSet rs,
        DBRequest request )
```

Create response db response.

Parameters

rs	the rs
request	the request

Returns

the db response

1.59.3.6 createVariation() `static RequestBuilder createVariation ( ObjCallback< RequestBuilder > og, PreMadeRequest.VariationCreator variationCreator ) [static]`

Create variation request builder.

Parameters

og	the og
variationCreator	the variation creator

Returns

the request builder

1.59.3.7 deleteAllUnFinishedGames() `static RequestBuilder deleteAllUnFinishedGames ( ) [static]`

Delete all un finished games request builder.

Returns

the request builder

1.59.3.8 games() `static RequestBuilder games ( ) [static]`

Games request builder.

Returns

the request builder

1.59.3.9 getArgs() `Arg[ ] getArgs ( )`

Get args arg [ ].

Returns

the arg [ ]

1.59.3.10 getArgVal() `String getArgVal ( int index )`

Gets arg val.

Parameters

index	the index

Returns

the arg val

1.59.3.11 getName() `String getName ( )`

Gets name.

Returns

the name

1.59.3.12 getPostDescription() `String getPostDescription ( )`

Gets post description.

Returns

the post description

1.59.3.13 getPreDescription() `String getPreDescription ( )`

Gets pre description.

Returns

the pre description

1.59.3.14 getShouldSync() `ArrayList< SyncedListType > getShouldSync ( )`

Gets should sync.

Returns

the should sync

1.59.3.15 setSubBuilder()

```
void setSubBuilder (
        RequestBuilder subBuilder )
```

Sets sub builder.

Parameters

subBuilder	the sub builder

1.59.3.16 statsByTimeOfDay() `static RequestBuilder statsByTimeOfDay ( ) [static]`

Stats by time of day request builder.

Returns

the request builder

1.59.3.17 top() `static RequestBuilder top ( ) [static]`

Top request builder.

Returns

the request builder

1.59.4 Member Data Documentation

1.59.4.1 args `final Arg [] args`

The request Arguments.

1.59.4.2 name `final String name [protected]`

The Name.

1.59.4.3 postDescription `String postDescription [protected]`

The Post description.

1.59.4.4 preDescription `String preDescription [protected]`

The Pre description.

1.59.4.5 statement `final` **`SQLStatement`** `statement [protected]`

The Statement.

1.59.4.6 subBuilder **`RequestBuilder`** `subBuilder [protected]`

The Sub builder.

1.59.4.7 TIE_STR `final String TIE_STR [static]`

The constant TIE_STR.

1.60 Selection

Classes

- interface **Join**
- interface **Order**

Public Member Functions

- **Selection** (Object selectFrom, Object[ ] select)
- **Selection** (Object selectFrom, **Condition** condition, Object[ ] select)
- **Selection nestMe** (Col... outerSelect)
- void **top** (Object top)
- void **join** (@ **Join** String joinType, **Table** joinWith, **Condition** condition, Col... groupBy)
- void **orderBy** (**Col** col, @ **Order** String order)

Protected Member Functions

- String **createStatement** ()

Additional Inherited Members

1.60.1 Detailed Description

Selection (p. 202) - a selection sql statement.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.60.2 Class Documentation

1.60.2.1 interface ver14::SharedClasses::DBActions::Statements::Selection::Join **Join** (p. **??**) - selection join.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Class Members

String	LEFT	The constant LEFT.

1.60.2.2 interface ver14::SharedClasses::DBActions::Statements::Selection::Order **Order** (p. **??**) - selection order by.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

Class Members

String	ASC	The constant ASC.
String	DESC	The constant DESC.

1.60.3 Constructor & Destructor Documentation

1.60.3.1 Selection() [1/2]

```
Selection (
        Object selectFrom,
        Object[] select )
```

Instantiates a new **Selection** (p. 202).

Parameters

selectFrom	the select from
select	the select

1.60.3.2 Selection() [2/2]

```
Selection (
        Object selectFrom,
        Condition condition,
        Object[] select )
```

Instantiates a new **Selection** (p. 202).

Parameters

selectFrom	the select from
condition	the condition
select	the select

1.60.4 Member Function Documentation

1.60.4.1 createStatement() `String createStatement ( ) [protected]`

Create statement string.

Returns

the string

Reimplemented from **SQLStatement** (p. 211).

1.60.4.2 join()

```
void join (
    @ Join String joinType,
    Table joinWith,
    Condition condition,
    Col... groupBy )
```

Join (p. **??**).

Parameters

joinType	the join type
joinWith	the join with
condition	the condition
groupBy	the group by

1.60.4.3 nestMe()

```
Selection nestMe (
    Col... outerSelect )
```

Nest me selection.

Parameters

outerSelect	the outer select

Returns

the selection

1.60.4.4 orderBy()

```
void orderBy (
    Col col,
    @ Order String order )
```

Order (p. **??**) by.

Parameters

col	the col
order	the order

1.60.4.5 top()

```
void top (
    Object top )
```

Top.

Parameters

top	the top

1.61 CastlingRights.Side Enum Reference

Public Member Functions

- **Side** (int **castledKingCol**, int **rookStartingCol**, int **castledRookCol**)
- String **toString** ()
- **Location kingFinalLoc** (**Location** currentKingLoc)

Public Attributes

- **KING**
- **QUEEN**
- final int **rookStartingCol**
- final int **castledRookCol**
- final int **castledKingCol**
- final int **kingTravelDistance**
- final String **castlingNotation**
- final int **asInt**
- final int **mult**

Static Public Attributes

- static final **Side**[ ] **SIDES**

1.61.1 Detailed Description

Side (p. 205) - Castling side.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.61.2 Constructor & Destructor Documentation

1.61.2.1 Side()

```
Side (
        int castledKingCol,
        int rookStartingCol,
        int castledRookCol )
```

Instantiates a new **Side** (p. 205).

Parameters

castledKingCol	the castled king col
rookStartingCol	the rook starting col
castledRookCol	the castled rook col

1.61.3 Member Function Documentation

1.61.3.1 kingFinalLoc()

```
Location kingFinalLoc (
        Location currentKingLoc )
```

King final loc location.

Parameters

currentKingLoc	the current king loc

Returns

the location

1.61.3.2 toString()

```
String toString ( )
```

To string string.

Returns

the string @hidden

1.61.4 Member Data Documentation

1.61.4.1 asInt `final int asInt`

The As int.

1.61.4.2 castledKingCol `final int castledKingCol`

The Castled king col.

1.61.4.3 castledRookCol `final int castledRookCol`

The Castled rook col.

1.61.4.4 castlingNotation `final String castlingNotation`

The Castling notation.

1.61.4.5 KING `KING`

King side.

1.61.4.6 kingTravelDistance `final int kingTravelDistance`

The King travel distance.

1.61.4.7 mult `final int mult`

The King Movement **Direction** (p. 63) Mult

1.61.4.8 QUEEN `QUEEN`

Queen side.

1.61.4.9 rookStartingCol `final int rookStartingCol`

The Rook starting col.

1.61.4.10 SIDES `final Side [] SIDES [static]`

The constant SIDES.

1.62 GameStatus.SpecificStatus Enum Reference

Public Member Functions

- **SpecificStatus** ()
- **SpecificStatus** (**GameStatusType gameStatusType**)
- String **toString** ()

Public Attributes

- **Checkmate**
- **TimedOut**
- **TimedOutVsInsufficientMaterial**
- **Resignation**
- **GameGoesOn**
- **ThreeFoldRepetition**
- **Stalemate**
- **InsufficientMaterial**
- **FiftyMoveRule**
- **TieByAgreement**
- **PlayerDisconnectedVsAi**
- **PlayerDisconnectedVsReal**
- **ServerStoppedGame**
- final **GameStatusType gameStatusType**

1.62.1 Detailed Description

Specific status - specific game status .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.62.2 Constructor & Destructor Documentation

1.62.2.1 SpecificStatus() [1/2] `SpecificStatus ( )`

Instantiates a new Specific status.

1.62.2.2 SpecificStatus() [2/2] `SpecificStatus ( GameStatusType gameStatusType )`

Instantiates a new Specific status.

Parameters

gameStatusType	the game status type

1.62.3 Member Function Documentation

1.62.3.1 toString() `String toString ( )`

To string string.

Returns

the string

1.62.4 Member Data Documentation

1.62.4.1 Checkmate `Checkmate`

Checkmate.

1.62.4.2 FiftyMoveRule `FiftyMoveRule`

Fifty move rule.

1.62.4.3 GameGoesOn `GameGoesOn`

Game goes on .

1.62.4.4 gameStatusType `final GameStatusType gameStatusType`

The Game status type.

1.62.4.5 InsufficientMaterial `InsufficientMaterial`

Insufficient material.

1.62.4.6 PlayerDisconnectedVsAi `PlayerDisconnectedVsAi`

The Player disconnected vs ai.

1.62.4.7 PlayerDisconnectedVsReal `PlayerDisconnectedVsReal`

The Player disconnected vs real.

1.62.4.8 Resignation `Resignation`

Resignation.

1.62.4.9 ServerStoppedGame `ServerStoppedGame`

Server stopped game.

1.62.4.10 Stalemate `Stalemate`

Stalemate.

1.62.4.11 ThreeFoldRepetition `ThreeFoldRepetition`

Three fold repetition.

1.62.4.12 TieByAgreement `TieByAgreement`

The Tie by agreement.

1.62.4.13 TimedOut `TimedOut`

Timed out.

1.62.4.14 TimedOutVsInsufficientMaterial `TimedOutVsInsufficientMaterial`

Timed out vs insufficient material.

1.63 SQLStatement

Public Member Functions

- **SQLStatement** (DBRequest.Type **type**)
- void **replace** (String replacing, String replaceWith)
- String **getStatement** ()
- String **toString** ()

Public Attributes

- final DBRequest.Type **type**

Protected Member Functions

- abstract String **createStatement** ()

1.63.1 Detailed Description

Sql statement - represents an sql statement.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.63.2 Constructor & Destructor Documentation

1.63.2.1 SQLStatement() `SQLStatement ( DBRequest.Type type )`

Instantiates a new Sql statement.

Parameters

type	the type

1.63.3 Member Function Documentation

1.63.3.1 createStatement() `abstract String createStatement ( ) [abstract], [protected]`

Create statement string.

Returns

the string

Reimplemented in **CustomStatement** (p. 56), **Delete** (p. 63), **Selection** (p. 204), and **Update** (p. 240).

1.63.3.2 getStatement() `String getStatement ( )`

Gets statement.

Returns

the statement

1.63.3.3 replace() `void replace ( String replacing, String replaceWith )`

```
void replace (
        String replacing,
        String replaceWith )
```

Replace.

Parameters

replacing	the replacing
replaceWith	the replace with

1.63.3.4 toString() `String toString ( )`

To string string.

Returns

the string

1.63.4 Member Data Documentation

1.63.4.1 type `final DBRequest.Type type`

The request Type.

1.64 Square

Public Member Functions

- **Square** (**Location** loc)
- void **setEmpty** ()
- **Square** (**Piece** piece, **Location** loc)
- **Piece** **getPiece** ()
- void **setPiece** (**Piece** piece)
- **Location** **getLoc** ()
- String **toString** ()
- String **getFen** ()
- boolean **isEmpty** ()
- String **getPieceIcon** ()

Static Public Attributes

- static final **Piece** **EMPTY_PIECE**

1.64.1 Detailed Description

Square (p. 212) represents a square on the logic board.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.64.2 Constructor & Destructor Documentation

1.64.2.1 Square() `[1/2]` `Square ( Location loc )`

Instantiates a new **Square** (p. 212).

Parameters

loc	the loc

1.64.2.2 Square() `[2/2]` `Square ( Piece piece, Location loc )`

Instantiates a new **Square** (p. 212).

Parameters

piece	the piece
loc	the loc

1.64.3 Member Function Documentation

1.64.3.1 getFen() `String getFen ( )`

Gets fen.

Returns

the fen

1.64.3.2 getLoc() `Location getLoc ( )`

Gets loc.

Returns

the loc

1.64.3.3 getPiece() `Piece getPiece ( )`

Gets piece.

Returns

the piece

1.64.3.4 getPieceIcon() `String getPieceIcon ( )`

Gets piece icon.

Returns

the piece icon

1.64.3.5 isEmpty() `boolean isEmpty ( )`

Is empty boolean.

Returns

the boolean

1.64.3.6 setEmpty() `void setEmpty ( )`

Sets empty.

1.64.3.7 setPiece() `void setPiece (`

```
Piece piece )
```

Sets piece.

Parameters

piece	the piece

1.64.3.8 toString() `String toString ( )`

To string string.

Returns

the string

1.64.4 Member Data Documentation

1.64.4.1 EMPTY_PIECE `final Piece EMPTY_PIECE [static]`

The constant EMPTY_PIECE.

1.65 StatusResponse

Public Member Functions

- **StatusResponse** (**Status status**, **DBRequest request**, int updatedRows)
- **StatusResponse** (**Status status**, String details, **DBRequest request**, int updatedRows)
- String **getDetails** ()
- boolean **isAnyData** ()
- **DBResponse clean** ()

Additional Inherited Members

1.65.1 Detailed Description

Status response - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.65.2 Constructor & Destructor Documentation

1.65.2.1 StatusResponse() [1/2]

```
StatusResponse (
    Status status,
    DBRequest request,
    int updatedRows )
```

Instantiates a new Status response.

Parameters

status	the status
request	the request
updatedRows	the updated rows

1.65.2.2 StatusResponse() `[2/2]`

```
StatusResponse (
    Status status,
    String details,
    DBRequest request,
    int updatedRows )
```

Instantiates a new Status response.

Parameters

status	the status
details	the details
request	the request
updatedRows	the updated rows

1.65.3 Member Function Documentation

1.65.3.1 clean()

```
DBResponse clean ( )
```

Clean db response.

Returns

the db response

Reimplemented from **DBResponse** (p. 60).

1.65.3.2 getDetails()

```
String getDetails ( )
```

Gets details.

Returns

the details

1.65.3.3 isAnyData() `boolean isAnyData ( )`

Is any data boolean.

Returns

the boolean

Reimplemented from **DBResponse** (p. 61).

1.66 MyJFrame.StringClosing

Public Member Functions

- default String **show** ()
- default boolean **checkClosingVal** (String val)

Additional Inherited Members

1.66.1 Detailed Description

The interface String closing.

1.66.2 Member Function Documentation

1.66.2.1 checkClosingVal() `default boolean checkClosingVal ( String val )`

Check closing val boolean.

Parameters

val	the val

Returns

the boolean

1.66.2.2 show() `default String show ( )`

Show string.

Returns

the string

1.67 StrUtils

Classes

- interface **IterationThingy**

Static Public Member Functions

- static String **dontCapFull** (String str)
- static String **htmlNewLines** (String str)
- static int **countMatches** (String str, @Language("RegExp") String match)
- static String[ ][ ] **format** (String[ ][ ] mat)
- static String[ ] **format** (String[ ] arr)
- static String **format** (String str)
- static boolean **isEmpty** (String str)
- static String **formatDate** (String longStr)
- static String **formatDate** (Date date)
- static String **formatDate** (Date date, String **format**)
- static String **parseURLS** (String str)
- static String **uppercase** (String str)
- static String **dontCapWord** (String str)
- static String **formatDateSQL** (Date date)
- static int **getPort** (Socket socket)
- static int **getPort** (SocketAddress socketAddress)
- static String **getUrl** (Socket socket)
- static String **getUrl** (SocketAddress socketAddress)
- static String **fitInside** (String str, JComponent comp)
- static String **fixHtml** (String str)
- static String **strINN** (Object... objs)
- static String **splitArr** (Object[ ] arr)
- static String **splitArr** (String divide, Object[ ] arr)
- static String **splitArr** (String divide, Object[ ] arr, boolean **format**)
- static String **clean** (String str)
- static String **createTimeGreeting** ()
- static void **main** (String[ ] args)
- static String **createTimeStr** (long millis)
- static String **awful** (String og)
- static String **dateTimePrefix** (String str)
- static String **repeat** (**IterationThingy**< String > iterationThingy, int numOfIterations)

1.67.1 Detailed Description

Str utils.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.67.2 Member Function Documentation

1.67.2.1 awful()

```
static String awful (
        String og )  [static]
```

Awful string.

Parameters

og	the og

Returns

the string

1.67.2.2 clean()

```
static String clean (
        String str )  [static]
```

Clean string.

Parameters

str	the str

Returns

the string

1.67.2.3 countMatches()

```
static int countMatches (
        String str,
        @Language("RegExp") String match )  [static]
```

Count matches int.

Parameters

str	the str
match	the match

Returns

the int

1.67.2.4 createTimeGreeting()

```
static String createTimeGreeting ( )  [static]
```

Create time greeting string.

Returns

the string

1.67.2.5 createTimeStr() `static String createTimeStr ( long millis ) [static]`

Create time str string.

Parameters

millis	the millis

Returns

the string

1.67.2.6 dateTimePrefix() `static String dateTimePrefix ( String str ) [static]`

Date time prefix string.

Parameters

str	the str

Returns

the string

1.67.2.7 dontCapFull() `static String dontCapFull ( String str ) [static]`

Dont cap full string.

Parameters

str	the str

Returns

the string

1.67.2.8 dontCapWord() `static String dontCapWord ( String str ) [static]`

Dont cap word string.

Parameters

str	the str

Returns

the string

1.67.2.9 fitInside()

```
static String fitInside (
        String str,
        JComponent comp ) [static]
```

Fit inside string.

Parameters

str	the str
comp	the comp

Returns

the string

1.67.2.10 fixHtml()

```
static String fixHtml (
        String str ) [static]
```

Fix html string.

Parameters

str	the str

Returns

the string

1.67.2.11 format() [1/3]

```
static String format (
        String str ) [static]
```

Format string.

Parameters

str	the str

Returns

the string

1.67.2.12 format() [2/3]

```
static String[] format (
    String[] arr ) [static]
```

Format string [ ].

Parameters

arr	the arr

Returns

the string [ ]

1.67.2.13 format() [3/3]

```
static String[][] format (
    String mat[][] ) [static]
```

Format string [ ] [ ].

Parameters

mat	the mat

Returns

the string [ ] [ ]

1.67.2.14 formatDate() [1/3]

```
static String formatDate (
    Date date ) [static]
```

Format date string.

Parameters

date	the date

Returns

the string

1.67.2.15 formatDate() [2/3]

```
static String formatDate (
            Date date,
            String format )  [static]
```

Format date string.

Parameters

date	the date
format	the format

Returns

the string

1.67.2.16 formatDate() [3/3]

```
static String formatDate (
            String longStr )  [static]
```

Format date string.

Parameters

longStr	the long str

Returns

the string

1.67.2.17 formatDateSQL()

```
static String formatDateSQL (
            Date date )  [static]
```

Format date sql string.

Parameters

date	the date

Returns

the string

1.67.2.18 getPort() [1/2]

```
static int getPort (
        Socket socket )  [static]
```

Gets port.

Parameters

socket	the socket

Returns

the port

1.67.2.19 getPort() [2/2]

```
static int getPort (
        SocketAddress socketAddress )  [static]
```

Gets port.

Parameters

socketAddress	the socket address

Returns

the port

1.67.2.20 getUrl() [1/2]

```
static String getUrl (
        Socket socket )  [static]
```

Gets url.

Parameters

socket	the socket

Returns

the url

1.67.2.21 getUrl() [2/2] `static String getUrl ( SocketAddress socketAddress ) [static]`

Gets url.

Parameters

socketAddress	the socket address

Returns

the url

1.67.2.22 htmlNewLines() `static String htmlNewLines ( String str ) [static]`

Html new lines string.

Parameters

str	the str

Returns

the string

1.67.2.23 isEmpty() `static boolean isEmpty ( String str ) [static]`

Is empty boolean.

Parameters

str	the str

Returns

the boolean

1.67.2.24 main() `static void main ( String[ ] args ) [static]`

The entry point of application.

Parameters

args	the input arguments

1.67.2.25 parseURLS() `static String parseURLS ( String str ) [static]`

Parse urls string.

Parameters

str	the str

Returns

the string

1.67.2.26 repeat() `static String repeat ( IterationThingy< String > iterationThingy, int numOfIterations ) [static]`

Repeat string.

Parameters

iterationThingy	the iteration thingy
numOfIterations	the num of iterations

Returns

the string

1.67.2.27 splitArr() [1/3] `static String splitArr ( Object[] arr ) [static]`

Split arr string.

Parameters

arr	the arr

Returns

the string

1.67.2.28 splitArr() [2/3]

```
static String splitArr (
        String divide,
        Object[] arr ) [static]
```

Split arr string.

Parameters

divide	the divide
arr	the arr

Returns

the string

1.67.2.29 splitArr() [3/3]

```
static String splitArr (
        String divide,
        Object[] arr,
        boolean format ) [static]
```

Split arr string.

Parameters

divide	the divide
arr	the arr
format	the format

Returns

the string

1.67.2.30 strINN()

```
static String strINN (
        Object... objs ) [static]
```

Str inn string.

Parameters

objs	the objs

Returns

the string

1.67.2.31 uppercase() `static String uppercase ( String str ) [static]`

Uppercase string.

Parameters

str	the str

Returns

the string

1.68 SwitchCase

Public Member Functions

- **SwitchCase** (**Condition** condition, **Col** ifTrue)
- **Condition condition** ()
- **Col ifTrue** ()
- String **toString** ()

Static Public Member Functions

- static **SwitchCase equals** (**Col** col, String value, **Col** ifTrue)
- static **SwitchCase defaultCase** (**Col** ifTrue)

1.68.1 Detailed Description

Switch case - represents a case that is meant to be used inside a switch case col. if the **`condition`** (p. **??**) is true, the **`ifTrue`** (p. **??**) col will display in the switch case col

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.68.2 Constructor & Destructor Documentation

1.68.2.1 SwitchCase() `SwitchCase ( Condition condition, Col ifTrue )`

Instantiates a new Switch case.

Parameters

condition	the condition
ifTrue	the if true

1.68.3 Member Function Documentation

1.68.3.1 condition() `Condition condition ( )`

Condition (p. 43) condition.

Returns

the condition

1.68.3.2 defaultCase() `static SwitchCase defaultCase ( Col ifTrue ) [static]`

Default case switch case.

Parameters

ifTrue	the if true

Returns

the switch case

1.68.3.3 equals() `static SwitchCase equals ( Col col, String value, Col ifTrue ) [static]`

Equals switch case.

Parameters

col	the col
value	the value
ifTrue	the if true

Returns

the switch case

1.68.3.4 ifTrue() `Col ifTrue ( )`

If true col.

Returns

the col

1.68.3.5 toString() `String toString ( )`

To string string.

Returns

the string

1.69 Table Enum Reference

Public Member Functions

- **Table** (Col... **cols**)
- String **tableAndValues** ()

Static Public Member Functions

- static String **escapeValues** (Object[ ] values, boolean quotes, boolean parentheses)

Public Attributes

- **Games**
- **UnfinishedGames**
- **Users**
- final **Col**[ ] **cols**

1.69.1 Detailed Description

Table (p. 230) - represents the tables in the db.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.69.2 Constructor & Destructor Documentation

1.69.2.1 Table()

```
Table (
        Col... cols )
```

Instantiates a new **Table** (p. 230).

Parameters

cols	the cols

1.69.3 Member Function Documentation

1.69.3.1 escapeValues()

```
static String escapeValues (
        Object[] values,
        boolean quotes,
        boolean parentheses )  [static]
```

Escape values string.

Parameters

values	the values
quotes	quotes
parentheses	parentheses

Returns

the escaped string

1.69.3.2 tableAndValues()

```
String tableAndValues ( )
```

Table (p. 230) and values string.

Returns

the string

1.69.4 Member Data Documentation

1.69.4.1 cols

```
final Col [] cols
```

The Cols in this table

1.69.4.2 Games

```
Games
```

Games table.

1.69.4.3 UnfinishedGames `UnfinishedGames`

Unfinished games table.

1.69.4.4 Users `Users`

Users table.

1.70 TableDBResponse

Public Member Functions

- **TableDBResponse** (String[ ] **columns**, String[ ][ ] **rows**, **DBRequest request**)
- **TableDBResponse** (String[ ] **columns**, String[ ][ ] **rows**, **Status status**, **DBRequest request**)
- int **numOfRows** ()
- String[ ] **getFirstRow** ()
- String **getCell** (int row, String col)
- String **getCell** (int row, **Col** col)
- boolean **isAnyData** ()
- **TableDBResponse clean** ()
- String **toString** ()
- String **rowToString** (String[ ] row)
- String[ ] **getColumns** ()
- String[ ][ ] **getRows** ()

Protected Member Functions

- **TableDBResponse** ()
- int **getColumnIndex** (String column)

Protected Attributes

- String[ ] **columns**
- String[ ][ ] **rows**

1.70.1 Detailed Description

Table db response - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.70.2 Constructor & Destructor Documentation

1.70.2.1 TableDBResponse() [1/3]

```
TableDBResponse (
        String[ ] columns,
        String rows[ ][ ],
        DBRequest request )
```

Instantiates a new Table db response.

Parameters

columns	the columns
rows	the rows
request	the request

1.70.2.2 TableDBResponse() [2/3]

```
TableDBResponse (
        String[] columns,
        String rows[][],
        Status status,
        DBRequest request )
```

Instantiates a new Table db response.

Parameters

columns	the columns
rows	the rows
status	the status
request	the request

1.70.2.3 TableDBResponse() [3/3]

```
TableDBResponse ( ) [protected]
```

Instantiates a new Table db response.

1.70.3 Member Function Documentation

1.70.3.1 clean()

```
TableDBResponse clean ( )
```

Clean table db response.

Returns

the table db response

Reimplemented from **DBResponse** (p. 60).

1.70.3.2 getCell() [1/2]

```
String getCell (
        int row,
        Col col )
```

Gets cell.

Parameters

row	the row
col	the col

Returns

the cell

1.70.3.3 getCell() [2/2]

```
String getCell (
        int row,
        String col )
```

Gets cell.

Parameters

row	the row
col	the col

Returns

the cell

1.70.3.4 getColumnIndex()

```
int getColumnIndex (
        String column ) [protected]
```

Gets column index.

Parameters

column	the column

Returns

the column index

1.70.3.5 getColumns()

```
String[ ] getColumns ( )
```

Get columns string [ ].

Returns

the string [ ]

1.70.3.6 getFirstRow() `String[] getFirstRow ( )`

Get first row string [ ].

Returns

the string [ ]

1.70.3.7 getRows() `String[][] getRows ( )`

Get rows string [ ] [ ].

Returns

the string [ ] [ ]

1.70.3.8 isAnyData() `boolean isAnyData ( )`

Is any data boolean.

Returns

the boolean

Reimplemented from **DBResponse** (p. 61).

1.70.3.9 numOfRows() `int numOfRows ( )`

Num of rows int.

Returns

the int

1.70.3.10 rowToString()

```
String rowToString (
        String[] row )
```

Row to string string.

Parameters

row	the row

Returns

the string

1.70.3.11 toString() `String toString ( )`

To string string.

Returns

the string

Reimplemented from **DBResponse** (p. 62).

1.70.4 Member Data Documentation

1.70.4.1 columns `String [ ] columns [protected]`

The Columns.

1.70.4.2 rows `String [ ][ ] rows [protected]`

The Rows.

1.71 ThreadsManager

Static Public Member Functions

- static void **handleErrors** (ThrowingRunnable runnable)
- static **MyThread** **createThread** (ThrowingRunnable runnable, boolean start)

1.71.1 Detailed Description

The type Threads manager.

1.71.2 Member Function Documentation

1.71.2.1 createThread()

```
static MyThread createThread (
        ThrowingRunnable runnable,
        boolean start ) [static]
```

Create thread my thread.

Parameters

runnable	the runnable
start	the start

Returns

the my thread

1.71.2.2 handleErrors() `static void handleErrors ( ThrowingRunnable runnable ) [static]`

Handle errors.

Parameters

runnable	the runnable

1.72 ThrowingCallback< T >

Public Member Functions

- void **callback** (T obj) throws Exception

1.72.1 Detailed Description

Throwing callback - a callback that might throw an exception .

Parameters

<T>	the callback type

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.72.2 Member Function Documentation

1.72.2.1 callback() `void callback ( T obj ) throws Exception`

Callback (p. 26).

Parameters

obj	the obj

Exceptions

Exception	the exception

1.73 UnfinishedGame

Public Member Functions

- **UnfinishedGame** (String **gameId**, String **creatorUsername**, GameSettings **gameSettings**, String **opponentUsername**, **PlayerColor** **playerColorToMove**, String **playerToMove**, Stack< **Move** > moveStack)
- boolean **isCreatorToMove** ()

Public Attributes

- final **PlayerColor** **playerColorToMove**
- final String **playerToMove**

Additional Inherited Members

1.73.1 Detailed Description

The type Unfinished game.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.73.2 Constructor & Destructor Documentation

1.73.2.1 UnfinishedGame() **UnfinishedGame** (

```
String gameId,
String creatorUsername,
GameSettings gameSettings,
String opponentUsername,
PlayerColor playerColorToMove,
String playerToMove,
Stack< Move > moveStack )
```

Instantiates a new Unfinished game.

Parameters

gameId	the game id
creatorUsername	the creator username
gameSettings	the game settings
opponentUsername	the opponent username
playerColorToMove	the player color to move
playerToMove	the player to move
moveStack	the move stack

1.73.3 Member Function Documentation

1.73.3.1 isCreatorToMove() `boolean isCreatorToMove ( )`

Is creator to move boolean.

Returns

the boolean

1.73.4 Member Data Documentation

1.73.4.1 playerColorToMove `final` **`PlayerColor`** `playerColorToMove`

The Player color to move.

1.73.4.2 playerToMove `final String playerToMove`

The Player to move.

1.74 Update

Public Member Functions

- **Update** (**Table** updating, **Condition** condition, NewValue... newValues)

Protected Member Functions

- String **createStatement** ()

Additional Inherited Members

1.74.1 Detailed Description

Update (p. 239) - a sql update statement.

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.74.2 Constructor & Destructor Documentation

1.74.2.1 Update()

```
Update (
    Table updating,
    Condition condition,
   NewValue...  newValues )
```

Instantiates a new **Update** (p. 239).

Parameters

updating	the updating
condition	the condition
newValues	the new values

1.74.3 Member Function Documentation

1.74.3.1 createStatement() `String createStatement ( )  [protected]`

Create statement string.

Returns

the string

Reimplemented from **SQLStatement** (p. 211).

1.75 ValueBtn< T >

Public Member Functions

- **ValueBtn** (String text, Font font, T value, **Callback**< T > onClick)

1.75.1 Detailed Description

Value btn - a value holding button.

Parameters

<T>	the value's type

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.75.2 Constructor & Destructor Documentation

1.75.2.1 ValueBtn()

```
ValueBtn (
    String text,
    Font font,
    T value,
    Callback< T > onClick )
```

Instantiates a new Value btn.

Parameters

text	the text
font	the font
value	the value
onClick	the on click

1.76 PreMadeRequest.VariationCreator

Public Member Functions

- Variation **create** (**RequestBuilder** actualBuilder)

1.76.1 Detailed Description

Variation creator - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.76.2 Member Function Documentation

1.76.2.1 create()

```
Variation create (
    RequestBuilder actualBuilder )
```

Create variation.

Parameters

actualBuilder	the actual builder

Returns

the variation

1.77 VoidCallback

Public Member Functions

- void **callback** ()

1.77.1 Detailed Description

Void callback - .

Author

Bezalel Avrahami (`bezalel3250@gmail.com`)

1.77.2 Member Function Documentation

1.77.2.1 callback() `void callback ( )`

Callback (p. 26).

Index