Package com.prowidesoftware.swift.model.mt.mt1xx

Class MT103

java.lang.Object
com.prowidesoftware.swift.model.AbstractMessage
com.prowidesoftware.swift.model.mt.AbstractMT
com.prowidesoftware.swift.model.mt.mt1xx.MT103

All Implemented Interfaces:

JsonSerializable, java.io.Serializable

```
@Generated
public class MT103
extends AbstractMT
implements java.io.Serializable
```

MT 103 - Single Customer Credit Transfer.

SWIFT MT103 (ISO 15022) message structure:

- Field 20 (M)
- Field 13 C (O) (repetitive)
- Field 23 B (M)
- Field 23 E (O) (repetitive)
- Field 26 T (O)
- Field 32 A (M)
- Field 33 B (O)
- Field 36 (O)
- Field 50 A,F,K (M)
- Field 51 A (O)
- Field 52 A,D (O)
- Field 53 A,B,D (O)
- Field 54 A,B,D (O)
- Field 55 A,B,D (O)
- Field 56 A,C,D (O)
- Field 57 A,B,C,D (O)
- Field 59 A,F,NONE (M)
- Field 70 (O)
- Field 71 A (M)
- Field 71 F (O) (repetitive)
- Field 71 G (O)
- Field 72 (O)
- Field 77 B (O)

This source code is specific to release **SRU 2024**

For additional resources check https://www.prowidesoftware.com/resources

Field Summary

Fields

Modifier and Type	Field	Description
static java.lang.String	NAME	Constant for MT name, this is part of the classname, after MT.
static int	SRU	Constant identifying the SRU to which this class belongs to.

Fields inherited from class com.prowidesoftware.swift.model.mt.AbstractMT

m

Constructor Summary

Constructors

Constructor	Description
MT103()	Creates and initializes a new MT103 input message setting TEST BICS as sender and receiver.
MT103(MtSwiftMessage m)	Creates an MT103 initialized with the parameter MtSwiftMessage.
MT103(SwiftMessage m)	Creates an MT103 initialized with the parameter SwiftMessage.
MT103(java.io.File file)	Creates a new MT103 by parsing a file with the message content in its swift FIN format.
MT103(java.io.InputStream stream)	Creates a new MT103 by parsing a input stream with the message content in its swift FIN format, using "UTF-8" as encoding.
MT103(java.lang.String fin)	Creates a new MT103 by parsing a String with the message content in its swift FIN format.

Method Summary

All Methods **Static Methods** **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
MT103	append(Field... fields)	Add all the fields to the end of the block4.
MT103	append (SwiftTagListBlock block)	Add all tags from block to the end of the block4.
MT103	append(Tag... tags)	Add all tags to the end of the block4.
static MT103	fromJson (java.lang.String json)	Creates an MT103 messages from its JSON representation.
java.util.List<Field13C>	getField13C()	Iterates through block4 fields and return all occurrences of fields whose names matches 13C, or Collections.emptyList() if none is found.
Field20	getField20()	Iterates through block4 fields and return the first one whose name matches 20, or null if none is found.
Field23B	getField23B()	Iterates through block4 fields and return the first one whose name matches 23B, or null if none is found.
java.util.List<Field23E>	getField23E()	Iterates through block4 fields and return all occurrences of fields whose names matches 23E, or Collections.emptyList() if none is found.
Field26T	getField26T()	Iterates through block4 fields and return the first

		fields and return the first one whose name matches 32A, or null if none is found.
Field33B	**getField33B**()	Iterates through block4 fields and return the first one whose name matches 33B, or null if none is found.
Field36	**getField36**()	Iterates through block4 fields and return the first one whose name matches 36, or null if none is found.
Field50A	**getField50A**()	Iterates through block4 fields and return the first one whose name matches 50A, or null if none is found.
Field50F	**getField50F**()	Iterates through block4 fields and return the first one whose name matches 50F, or null if none is found.
Field50K	**getField50K**()	Iterates through block4 fields and return the first one whose name matches 50K, or null if none is found.
Field51A	**getField51A**()	Iterates through block4 fields and return the first one whose name matches 51A, or null if none is found.
Field52A	**getField52A**()	Iterates through block4 fields and return the first one whose name matches 52A, or null if none is found.
Field52D	**getField52D**()	Iterates through block4 fields and return the first one whose name matches

		one whose name matches 53A, or null if none is found.
Field53B	**getField53B**()	Iterates through block4 fields and return the first one whose name matches 53B, or null if none is found.
Field53D	**getField53D**()	Iterates through block4 fields and return the first one whose name matches 53D, or null if none is found.
Field54A	**getField54A**()	Iterates through block4 fields and return the first one whose name matches 54A, or null if none is found.
Field54B	**getField54B**()	Iterates through block4 fields and return the first one whose name matches 54B, or null if none is found.
Field54D	**getField54D**()	Iterates through block4 fields and return the first one whose name matches 54D, or null if none is found.
Field55A	**getField55A**()	Iterates through block4 fields and return the first one whose name matches 55A, or null if none is found.
Field55B	**getField55B**()	Iterates through block4 fields and return the first one whose name matches 55B, or null if none is found.
Field55D	**getField55D**()	Iterates through block4 fields and return the first one whose name matches

		one whose name matches 56A, or null if none is found.
Field56C	**getField56C**()	Iterates through block4 fields and return the first one whose name matches 56C, or null if none is found.
Field56D	**getField56D**()	Iterates through block4 fields and return the first one whose name matches 56D, or null if none is found.
Field57A	**getField57A**()	Iterates through block4 fields and return the first one whose name matches 57A, or null if none is found.
Field57B	**getField57B**()	Iterates through block4 fields and return the first one whose name matches 57B, or null if none is found.
Field57C	**getField57C**()	Iterates through block4 fields and return the first one whose name matches 57C, or null if none is found.
Field57D	**getField57D**()	Iterates through block4 fields and return the first one whose name matches 57D, or null if none is found.
Field59	**getField59**()	Iterates through block4 fields and return the first one whose name matches 59, or null if none is found.
Field59A	**getField59A**()	Iterates through block4 fields and return the first one whose name matches 59A, or null if none is

		59F, or null if none is found.
Field70	**getField70**()	Iterates through block4 fields and return the first one whose name matches 70, or null if none is found.
Field71A	**getField71A**()	Iterates through block4 fields and return the first one whose name matches 71A, or null if none is found.
java.util.List<**Field71F**>	**getField71F**()	Iterates through block4 fields and return all occurrences of fields whose names matches 71F, or `Collections.emptyList()` if none is found.
Field71G	**getField71G**()	Iterates through block4 fields and return the first one whose name matches 71G, or null if none is found.
Field72	**getField72**()	Iterates through block4 fields and return the first one whose name matches 72, or null if none is found.
Field77B	**getField77B**()	Iterates through block4 fields and return the first one whose name matches 77B, or null if none is found.
java.lang.String	**getMessageType**()	Returns this MT number.
java.lang.String	**getUETR**()	Gets the Unique End to End Transaction Reference (field 121 from block 3).
static **MT103**	**parse**(**MtSwiftMessage** m)	Creates an MT103 initialized with the parameter MtSwiftMessage.

static **MT103**	**parse**(java.io.InputStream stream)	Creates a new MT103 by parsing a input stream with the message content in its swift FIN format, using "UTF-8" as encoding.
static **MT103**	**parse**(java.lang.String fin)	Creates a new MT103 by parsing a String with the message content in its swift FIN format.

Methods inherited from class com.prowidesoftware.swift.model.mt.AbstractMT

addField, containsSequence, containsSequenceList, create, create, getApplicationId, getFields, getLogicalTerminal, getMessagePriority, getMtId, getReceiver, getSender, getSequence, getSequence, getSequence, getSequenceList, getSequenceList, getSequenceNumber, getServiceId, getSessionNumber, getSignature, getSwiftMessage, getSwiftMessageNotNullOrException, getVariant, isIncoming, isInput, isOutgoing, isOutput, isType, message, nameFromClass, read, setReceiver, setReceiver, setSender, setSender, setSignature, setSwiftMessage, tag, tags, toJson, toString, write, write, xml

Methods inherited from class com.prowidesoftware.swift.model.AbstractMessage

getMessageStandardType, isMT, isMX

Methods inherited from class java.lang.Object

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

Field Detail

SRU

public static final int SRU

NAME

`public static final java.lang.String NAME`

Constant for MT name, this is part of the classname, after MT.

See Also:

Constant Field Values

Constructor Detail

MT103

`public MT103(SwiftMessage m)`

Creates an MT103 initialized with the parameter SwiftMessage.

Parameters:

`m` - swift message with the MT103 content

MT103

`public MT103(MtSwiftMessage m)`

Creates an MT103 initialized with the parameter MtSwiftMessage.

Parameters:

`m` - swift message with the MT103 content, the parameter can not be null

See Also:

`MT103(String)`

MT103

`public MT103()`

Creates and initializes a new MT103 input message setting TEST BICS as sender and receiver. All mandatory header attributes are completed with default values.

MT103

```
public MT103(java.lang.String sender,
             java.lang.String receiver)
```

Creates and initializes a new MT103 input message from sender to receiver. All mandatory header attributes are completed with default values. In particular the sender and receiver addresses will be filled with proper default LT identifier and branch codes if not provided,

Parameters:

`sender` - the sender address as a bic8, bic11 or full logical terminal consisting of 12 characters

`receiver` - the receiver address as a bic8, bic11 or full logical terminal consisting of 12 characters

Since:

7.7

MT103

```
public MT103(java.lang.String fin)
```

Creates a new MT103 by parsing a String with the message content in its swift FIN format. If the fin parameter is null or the message cannot be parsed, the internal message object will be initialized (blocks will be created) but empty. If the string contains multiple messages, only the first one will be parsed.

Parameters:

`fin` - a string with the MT message in its FIN swift format

Since:

7.7

MT103

```
public MT103(java.io.InputStream stream)
      throws java.io.IOException
```

Creates a new MT103 by parsing a input stream with the message content in its swift FIN format, using "UTF-8" as encoding. If the message content is null or cannot be parsed, the internal message object will be initialized (blocks will be created) but empty. If the stream contains multiple messages, only the first one will be parsed.

Parameters:

Since:

7.7

MT103

```
public MT103(java.io.File file)
      throws java.io.IOException
```

Creates a new MT103 by parsing a file with the message content in its swift FIN format. If the file content is null or cannot be parsed as a message, the internal message object will be initialized (blocks will be created) but empty. If the file contains multiple messages, only the first one will be parsed.

Parameters:

`file` - a file with the MT message in its FIN swift format.

Throws:

`java.io.IOException` - if the file content cannot be read

Since:

7.7

Method Detail

parse

```
public static MT103 parse(MtSwiftMessage m)
```

Creates an MT103 initialized with the parameter MtSwiftMessage.

Parameters:

`m` - swift message with the MT103 content

Returns:

the created object or null if the parameter is null

Since:

7.7

See Also:

`MT103(String)`

Creates a new MT103 by parsing a String with the message content in its swift FIN format. If the fin parameter cannot be parsed, the returned MT103 will have its internal message object initialized (blocks will be created) but empty. If the string contains multiple messages, only the first one will be parsed.

Parameters:

`fin` - a string with the MT message in its FIN swift format. *fin may be null in which case this method returns null*

Returns:

a new instance of MT103 or null if fin is null

Since:

7.7

parse

```
public static MT103 parse(java.io.InputStream stream) throws
java.io.IOException
```

Creates a new MT103 by parsing a input stream with the message content in its swift FIN format, using "UTF-8" as encoding. If the stream contains multiple messages, only the first one will be parsed.

Parameters:

`stream` - an input stream in UTF-8 encoding with the MT message in its FIN swift format.

Returns:

a new instance of MT103 or null if stream is null or the message cannot be parsed

Throws:

`java.io.IOException` - if the stream data cannot be read

Since:

7.7

parse

```
public static MT103 parse(java.io.File file) throws java.io.IOException
```

Creates a new MT103 by parsing a file with the message content in its swift FIN format. If the file contains multiple messages, only the first one will be parsed.

Parameters:

`file` - a file with the MT message in its FIN swift format.

`java.io.IOException` - if the file content cannot be read

Since:

7.7

getMessageType

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

Returns this MT number.

Specified by:

`getMessageType` in class `AbstractMT`

Returns:

the message type number of this MT

Since:

6.4

getUETR

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

Gets the Unique End to End Transaction Reference (field 121 from block 3).

This field is used by the SWIFT gpi service to track payments messages.

Returns:

the UETR value or null if block3 or field 121 in block3 are not present

Since:

7.10.0

append

```
public MT103 append(SwiftTagListBlock block)
```

Add all tags from block to the end of the block4.

Overrides:

`append` in class `AbstractMT`

Parameters:

7.6

append

```
public MT103 append(Tag... tags)
```

Add all tags to the end of the block4.

Overrides:

append in class AbstractMT

Parameters:

tags - to append

Returns:

this object to allow method chaining

Since:

7.6

append

```
public MT103 append(Field... fields)
```

Add all the fields to the end of the block4.

Overrides:

append in class AbstractMT

Parameters:

fields - to append

Returns:

this object to allow method chaining

Since:

7.6

fromJson

```
public static MT103 fromJson(java.lang.String json)
```

Creates an MT103 messages from its JSON representation.

Returns:

a new instance of MT103

Since:

7.10.3

getField20

```
public Field20 getField20()
```

Iterates through block4 fields and return the first one whose name matches 20, or null if none is found. The first occurrence of field 20 at MT103 is expected to be the only one.

Returns:

a Field20 object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField23B

```
public Field23B getField23B()
```

Iterates through block4 fields and return the first one whose name matches 23B, or null if none is found. The first occurrence of field 23B at MT103 is expected to be the only one.

Returns:

a Field23B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField26T

```
public Field26T getField26T()
```

a Field26T object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField32A

`public Field32A getField32A()`

Iterates through block4 fields and return the first one whose name matches 32A, or null if none is found. The first occurrence of field 32A at MT103 is expected to be the only one.

Returns:

a Field32A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField33B

`public Field33B getField33B()`

Iterates through block4 fields and return the first one whose name matches 33B, or null if none is found. The first occurrence of field 33B at MT103 is expected to be the only one.

Returns:

a Field33B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField36

Returns:

a Field36 object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField50A

```
public Field50A getField50A()
```

Iterates through block4 fields and return the first one whose name matches 50A, or null if none is found. The first occurrence of field 50A at MT103 is expected to be the only one.

Returns:

a Field50A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField50F

```
public Field50F getField50F()
```

Iterates through block4 fields and return the first one whose name matches 50F, or null if none is found. The first occurrence of field 50F at MT103 is expected to be the only one.

Returns:

a Field50F object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

null if none is found. The first occurrence of field 50K at MT103 is expected to be the only one.

Returns:

a Field50K object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField51A

```
public Field51A getField51A()
```

Iterates through block4 fields and return the first one whose name matches 51A, or null if none is found. The first occurrence of field 51A at MT103 is expected to be the only one.

Returns:

a Field51A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField52A

```
public Field52A getField52A()
```

Iterates through block4 fields and return the first one whose name matches 52A, or null if none is found. The first occurrence of field 52A at MT103 is expected to be the only one.

Returns:

a Field52A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

Iterates through block4 fields and return the first one whose name matches 52D, or null if none is found. The first occurrence of field 52D at MT103 is expected to be the only one.

Returns:

a Field52D object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField53A

```
public Field53A getField53A()
```

Iterates through block4 fields and return the first one whose name matches 53A, or null if none is found. The first occurrence of field 53A at MT103 is expected to be the only one.

Returns:

a Field53A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

null if none is found. The first occurrence of field 53B at MT103 is expected to be the only one.

Returns:

a Field53B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField53D

```
public Field53D getField53D()
```

Iterates through block4 fields and return the first one whose name matches 53D, or null if none is found. The first occurrence of field 53D at MT103 is expected to be the only one.

Returns:

a Field53D object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField54A

```
public Field54A getField54A()
```

Iterates through block4 fields and return the first one whose name matches 54A, or null if none is found. The first occurrence of field 54A at MT103 is expected to be the only one.

Returns:

a Field54A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

Iterates through block4 fields and return the first one whose name matches 54B, or null if none is found. The first occurrence of field 54B at MT103 is expected to be the only one.

Returns:

a Field54B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField54D

```
public Field54D getField54D()
```

Iterates through block4 fields and return the first one whose name matches 54D, or null if none is found. The first occurrence of field 54D at MT103 is expected to be the only one.

Returns:

a Field54D object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

null if none is found. The first occurrence of field 55A at MT103 is expected to be the only one.

Returns:

a Field55A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField55B

```
public Field55B getField55B()
```

Iterates through block4 fields and return the first one whose name matches 55B, or null if none is found. The first occurrence of field 55B at MT103 is expected to be the only one.

Returns:

a Field55B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField55D

```
public Field55D getField55D()
```

Iterates through block4 fields and return the first one whose name matches 55D, or null if none is found. The first occurrence of field 55D at MT103 is expected to be the only one.

Returns:

a Field55D object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

Iterates through block4 fields and return the first one whose name matches 56A, or null if none is found. The first occurrence of field 56A at MT103 is expected to be the only one.

Returns:

a Field56A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField56C

```
public Field56C getField56C()
```

Iterates through block4 fields and return the first one whose name matches 56C, or null if none is found. The first occurrence of field 56C at MT103 is expected to be the only one.

Returns:

a Field56C object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

null if none is found. The first occurrence of field 56D at MT103 is expected to be the only one.

Returns:

a Field56D object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField57A

```
public Field57A getField57A()
```

Iterates through block4 fields and return the first one whose name matches 57A, or null if none is found. The first occurrence of field 57A at MT103 is expected to be the only one.

Returns:

a Field57A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField57B

```
public Field57B getField57B()
```

Iterates through block4 fields and return the first one whose name matches 57B, or null if none is found. The first occurrence of field 57B at MT103 is expected to be the only one.

Returns:

a Field57B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

Iterates through block4 fields and return the first one whose name matches 57C, or null if none is found. The first occurrence of field 57C at MT103 is expected to be the only one.

Returns:

a Field57C object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField57D

```
public Field57D getField57D()
```

Iterates through block4 fields and return the first one whose name matches 57D, or null if none is found. The first occurrence of field 57D at MT103 is expected to be the only one.

Returns:

a Field57D object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

null if none is found. The first occurrence of field 59A at MT103 is expected to be the only one.

Returns:

a Field59A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField59F

```
public Field59F getField59F()
```

Iterates through block4 fields and return the first one whose name matches 59F, or null if none is found. The first occurrence of field 59F at MT103 is expected to be the only one.

Returns:

a Field59F object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField59

```
public Field59 getField59()
```

Iterates through block4 fields and return the first one whose name matches 59, or null if none is found. The first occurrence of field 59 at MT103 is expected to be the only one.

Returns:

a Field59 object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

Iterates through block4 fields and return the first one whose name matches 70, or null if none is found. The first occurrence of field 70 at MT103 is expected to be the only one.

Returns:

a Field70 object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField71A

```
public Field71A getField71A()
```

Iterates through block4 fields and return the first one whose name matches 71A, or null if none is found. The first occurrence of field 71A at MT103 is expected to be the only one.

Returns:

a Field71A object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

null if none is found. The first occurrence of field 71G at MT103 is expected to be the only one.

Returns:

a Field71G object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField72

```
public Field72 getField72()
```

Iterates through block4 fields and return the first one whose name matches 72, or null if none is found. The first occurrence of field 72 at MT103 is expected to be the only one.

Returns:

a Field72 object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

getField77B

```
public Field77B getField77B()
```

Iterates through block4 fields and return the first one whose name matches 77B, or null if none is found. The first occurrence of field 77B at MT103 is expected to be the only one.

Returns:

a Field77B object or null if the field is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagByName(String)`

Iterates through block4 fields and return all occurrences of fields whose names matches 13C, or `Collections.emptyList()` if none is found. Multiple occurrences of field 13C at MT103 are expected at one sequence or across several sequences.

Returns:

a List of Field13C objects or `Collections.emptyList()` if none is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagsByName(String)`

getField23E

```
public java.util.List<Field23E> getField23E()
```

Iterates through block4 fields and return all occurrences of fields whose names matches 23E, or `Collections.emptyList()` if none is found. Multiple occurrences of field 23E at MT103 are expected at one sequence or across several sequences.

Returns:

a List of Field23E objects or `Collections.emptyList()` if none is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagsByName(String)`

matches 71F, or `Collections.emptyList()` if none is found. Multiple occurrences of field 71F at MT103 are expected at one sequence or across several sequences.

Returns:

a List of Field71F objects or `Collections.emptyList()` if none is not found

Throws:

`java.lang.IllegalStateException` - if SwiftMessage object is not initialized

See Also:

`SwiftTagListBlock.getTagsByName(String)`