

# Table Of Content

---

<a href="#">api</a> .....	2
<a href="#">Generator</a> .....	2
<a href="#">Parser</a> .....	5
<a href="#">ServiceGenerator</a> .....	6
<a href="#">library</a> .....	10
<a href="#">protocole</a> .....	11
<a href="#">HL7Reader</a> .....	11
<a href="#">HL7Writer</a> .....	12
<a href="#">LowerLayerProtocol</a> .....	13
<a href="#">MinLLPReader</a> .....	14
<a href="#">MinLLPWriter</a> .....	15
<a href="#">MinLowerLayerProtocol</a> .....	16
<a href="#">ServeurProtocol</a> .....	17
<a href="#">Index</a> .....	19

# Package api

## Interface Summary

### [ServiceGenerator](#)

Interface qui permet de g  n  rer des messages HL7

## Class Summary

### [Generator](#)

Classe qui permet de g  n  rer des messages HL7

### [Parser](#)

Classe d'analyse syntaxique des messages HL7

---

api

## Class Generator

```
java.lang.Object
|
+--api.Generator
```

### All Implemented Interfaces:

[ServiceGenerator](#)

---

< [Constructors](#) > < [Methods](#) >

---

```
public class Generator
extends java.lang.Object
implements ServiceGenerator
```

Classe qui permet de g  n  rer des messages HL7

### Author:

Anthony CROUZET Polytech'Grenoble TIS3

### Version:

1.0

## Constructors

### Generator

```
public Generator()
```

## Methods

## admit

```
public java.lang.String admit(library.interfaces.Patient patient)
```

GÃ©nÃ©re un message d'admission ADT A01

**Parameters:**

patient - le patient Ã  admettre

**Returns:**

le message HL7 ADT A01

---

## admit

```
public java.lang.String admit(library.interfaces.Patient patient,  
                                library.interfaces.Action action)
```

GÃ©nÃ©re un message d'admission de patient ADT A01

**Parameters:**

patient - le patient Ã  admettre

action - action de l'Ã©vÃ©nement

**Returns:**

le message HL7 ADT A01

---

## endPat

```
public java.lang.String endPat(library.interfaces.Patient patient)
```

GÃ©nÃ©re un message de sorti de patient ADT A03

**Parameters:**

patient - le patient Ã  dÃ©charger

**Returns:**

le message HL7 ADT A03

---

## endPat

```
public java.lang.String endPat(library.interfaces.Patient patient,  
                                library.interfaces.Action action)
```

GÃ©nÃ©re un message de sorti de patient ADT A03

**Parameters:**

patient - le patient Ã  dÃ©charger

action - action de l'Ã©vÃ©nement

**Returns:**

le message HL7 ADT A03

---

## msgAck1

```
public java.lang.String msgAck1(java.lang.String id,  
                                java.lang.String code)
```

G n re un message ACK A01 qui accuse de la bonne r ception du message ainsi que de son traitement dans le cas applicatif.

**Parameters:**

id - l'identifiant du message qui est accus   
code - Code d'accus  de r ception

**Returns:**

le message HL7 ACK A01

---

## msgAck2

```
public java.lang.String msgAck2(java.lang.String id,  
                                java.lang.String code)
```

G n re un message ACK A02 qui accuse de la bonne r ception du message ainsi que de son traitement dans le cas applicatif.

**Parameters:**

id - l'identifiant du message qui est accus   
code - Code d'accus  de r ception

**Returns:**

le message HL7 ACK A02

---

## msgAck3

```
public java.lang.String msgAck3(java.lang.String id,  
                                java.lang.String code)
```

G n re un message ACK A03 qui accuse de la bonne r ception du message ainsi que de son traitement dans le cas applicatif.

**Parameters:**

id - l'identifiant du message qui est accus   
code - Code d'accus  de r ception

**Returns:**

le message HL7 ACK A03

---

## transPat

```
public java.lang.String transPat(library.interfaces.Patient patient)
```

G  n  re un message de transfert de patient ADT A02

**Parameters:**

patient - le patient    transf  rer

**Returns:**

le message HL7 ADT A02

---

## transPat

```
public java.lang.String transPat(library.interfaces.Patient patient,  
                                   library.interfaces.Action action)
```

G  n  re un message de transfert ADT A02

**Parameters:**

patient - le patient    transf  rer

action - action de l'  v  nement

**Returns:**

le message HL7 ADT A02

---

api

## Class Parser

```
java.lang.Object  
|  
+--api.Parser
```

---

< [Constructors](#) > < [Methods](#) >

---

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

Classe d'analyse syntaxique des messages HL7

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

**Version:**

1.0

## Constructors

## Parser

```
public Parser(java.lang.String msg)
```

Constructeur du analyseur syntaxique des messages HL7

**Parameters:**

msg - message HL7 Ã analyser

## Methods

### getMessage

```
public library.structure.groupe.messages.Message getMessage()
```

Accesseur du message HL7 analysÃ©

**Returns:**

le message HL7 analysÃ©

### getMessageInterface

```
public library.interfaces.MessageInterface getMessageInterface()
```

Accesseur du message HL7 analysÃ©

**Returns:**

le message accusÃ© de rÃ©ception HL7

### getPatient

```
public library.interfaces.Patient getPatient()
```

Accesseur du patient

**Returns:**

le patient dÃ©crit dans le message HL7

---

api

## Interface ServiceGenerator

---

< [Methods](#) >

---

```
public interface ServiceGenerator
```

Interface qui permet de générer des messages HL7

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

**Version:**

1.0

## Methods

### admit

```
public java.lang.String admit(library.interfaces.Patient patient)
```

Génère un message d'admission ADT A01

**Parameters:**

patient - le patient à admettre

**Returns:**

le message HL7 ADT A01

---

### admit

```
public java.lang.String admit(library.interfaces.Patient patient,  
                                library.interfaces.Action action)
```

Génère un message d'admission de patient ADT A01

**Parameters:**

patient - le patient à admettre

action - action de l'événement

**Returns:**

le message HL7 ADT A01

---

### endPat

```
public java.lang.String endPat(library.interfaces.Patient patient)
```

Génère un message de sorti de patient ADT A03

**Parameters:**

patient - le patient à décharger

**Returns:**

le message HL7 ADT A03

---

## endPat

```
public java.lang.String endPat(library.interfaces.Patient patient,  
                                library.interfaces.Action action)
```

G n re un message de sorti de patient ADT A03

**Parameters:**

patient - le patient   d charger  
action - action de l' v nement

**Returns:**

le message HL7 ADT A03

---

## msgAck1

```
public java.lang.String msgAck1(java.lang.String id,  
                                   java.lang.String code)
```

G n re un message ACK A01 qui accuse de la bonne r ception du message ainsi que de son traitement dans le cas applicatif.

**Parameters:**

id - l'identifiant du message qui est accus    
code - Code d'accus   de r ception

**Returns:**

le message HL7 ACK A01

---

## msgAck2

```
public java.lang.String msgAck2(java.lang.String id,  
                                   java.lang.String code)
```

G n re un message ACK A02 qui accuse de la bonne r ception du message ainsi que de son traitement dans le cas applicatif.

**Parameters:**

id - l'identifiant du message qui est accus    
code - Code d'accus   de r ception

**Returns:**

le message HL7 ACK A02

---



## msgAck3

```
public java.lang.String msgAck3(java.lang.String id,  
                                java.lang.String code)
```

G n re un message ACK A03 qui accuse de la bonne r ception du message ainsi que de son traitement dans le cas applicatif.

**Parameters:**

id - l'identifiant du message qui est accus   
code - Code d'accus  de r ception

**Returns:**

le message HL7 ACK A03

---

## transPat

```
public java.lang.String transPat(library.interfaces.Patient patient)
```

G n re un message de transfert de patient ADT A02

**Parameters:**

patient - le patient   transf rer

**Returns:**

le message HL7 ADT A02

---

## transPat

```
public java.lang.String transPat(library.interfaces.Patient patient,  
                                library.interfaces.Action action)
```

G n re un message de transfert ADT A02

**Parameters:**

patient - le patient   transf rer  
action - action de l' v nement

**Returns:**

le message HL7 ADT A02

# Package library

# Package protocole

## Interface Summary

### [HL7Reader](#)

un HL7Reader implemente la lecture d'un input Stream utilisant un LLP particulier.

### [HL7Writer](#)

un HL7Writer implemente l'écriture d'un output Stream utilisant un LLP particulier.

## Class Summary

### [LowerLayerProtocol](#)

### [MinLLPReader](#)

### [MinLLPWriter](#)

### [MinLowerLayerProtocol](#)

### [ServeurProtocol](#)

---

protocole

## Interface HL7Reader

---

< [Methods](#) >

---

public interface **HL7Reader**

un HL7Reader implemente la lecture d'un input Stream utilisant un LLP particulier.

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

## Methods

### close

public void **close()**

Ferme le stream sous-jacent et toutes les autres ressources qui devraient être fermées.

---

## getMessage

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

---

## setInputStream

```
public void setInputStream(java.io.InputStream in)
```

---

protocole

# Interface HL7Writer

---

< [Methods](#) >

---

```
public interface HL7Writer
```

un HL7Writer implemente l'écriture d'un output Stream utilisant un LLP particulier.

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

## Methods

### close

```
public void close()
```

---

### setOutputStream

```
public void setOutputStream(java.io.OutputStream out)
```

---

### writeMessage

```
public void writeMessage(java.lang.String message)
```

---

protocole

# Class LowerLayerProtocol

```
java.lang.Object
|
+--protocole.LowerLayerProtocol
```

Direct Known Subclasses:

[MinLowerLayerProtocol](#)

---

< [Constructors](#) > < [Methods](#) >

---

public abstract class **LowerLayerProtocol**  
extends java.lang.Object

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

## Constructors

### LowerLayerProtocol

```
public LowerLayerProtocol()
```

## Methods

### getReader

```
public abstract HL7Reader getReader(java.io.InputStream in)
```

---

### getWriter

```
public abstract HL7Writer getWriter(java.io.OutputStream out)
```

---

protocole

# Class MinLLPReader

```
java.lang.Object
|
+--protocole.MinLLPReader
```

All Implemented Interfaces:

[HL7Reader](#)

---

< [Fields](#) > < [Constructors](#) > < [Methods](#) >

---

```
public class MinLLPReader
extends java.lang.Object
implements HL7Reader
```

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

## Fields

### CHARSET\_KEY

```
public static final java.lang.String CHARSET_KEY
```

## Constructors

### MinLLPReader

```
public MinLLPReader(java.io.InputStream in)
```

## Methods

### close

```
public synchronized void close()
```

---

### getMessage

```
public synchronized java.lang.String getMessage()
```

---

## setInputStream

```
public synchronized void setInputStream(java.io.InputStream in)
```

---

protocole

## Class MinLLPWriter

```
java.lang.Object
|
+--protocole.MinLLPWriter
```

All Implemented Interfaces:

[HL7Writer](#)

---

< [Constructors](#) > < [Methods](#) >

---

```
public class MinLLPWriter
extends java.lang.Object
implements HL7Writer
```

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

## Constructors

### MinLLPWriter

```
public MinLLPWriter(java.io.OutputStream out)
```

## Methods

### close

```
public synchronized void close()
```

---

## setOutputStream

```
public synchronized void setOutputStream(java.io.OutputStream out)
```

---

## writeMessage

```
public synchronized void writeMessage(java.lang.String message)
```

---

## writeMessage

```
public synchronized void writeMessage(java.lang.String message,  
                                         java.lang.String charset)
```

---

protocole

# Class MinLowerLayerProtocol

```
java.lang.Object  
|  
+--LowerLayerProtocol  
    |  
    +--protocole.MinLowerLayerProtocol
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class MinLowerLayerProtocol  
extends LowerLayerProtocol
```

**Author:**

Anthony CROUZET Polytech'Grenoble TIS3

## Constructors

## MinLowerLayerProtocol

```
public MinLowerLayerProtocol()
```

## Methods



## getReader

```
public HL7Reader getReader( java.io.InputStream in)
```

Overrides:

[getReader](#) in class [LowerLayerProtocol](#)

---

## getWriter

```
public HL7Writer getWriter( java.io.OutputStream out)
```

Overrides:

[getWriter](#) in class [LowerLayerProtocol](#)

---

protocole

# Class ServeurProtocol

```
java.lang.Object
|
+--protocole.ServeurProtocol
```

---

< [Constructors](#) > < [Methods](#) >

---

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

## Constructors

### ServeurProtocol

```
public ServeurProtocol()
```

## Methods

### getParser

```
public Parser getParser()
```

---

## **processInput**

```
public java.lang.String processInput(java.lang.String theInput)
```

# INDEX

## A

[admit](#) ... 3  
[admit](#) ... 3  
[admit](#) ... 7  
[admit](#) ... 7

## C

[close](#) ... 11  
[close](#) ... 12  
[close](#) ... 14  
[close](#) ... 15  
[CHARSET\\_KEY](#) ... 14

## E

[endPat](#) ... 3  
[endPat](#) ... 3  
[endPat](#) ... 7  
[endPat](#) ... 8

## G

[getMessage](#) ... 6  
[getMessage](#) ... 12  
[getMessage](#) ... 14  
[getMessageInterface](#) ... 6  
[getParser](#) ... 17  
[getPatient](#) ... 6  
[getReader](#) ... 13  
[getReader](#) ... 17  
[getWriter](#) ... 13  
[getWriter](#) ... 17  
[Generator](#) ... 2  
[Generator](#) ... 2

## H

[HL7Reader](#) ... 11  
[HL7Writer](#) ... 12

## L

[LowerLayerProtocol](#) ... 13  
[LowerLayerProtocol](#) ... 13

## M

[msgAck1](#) ... 4  
[msgAck1](#) ... 8  
[msgAck2](#) ... 4  
[msgAck2](#) ... 8  
[msgAck3](#) ... 4  
[msgAck3](#) ... 9  
[MinLLPReader](#) ... 14  
[MinLLPReader](#) ... 14  
[MinLLPWriter](#) ... 15  
[MinLLPWriter](#) ... 15  
[MinLowerLayerProtocol](#) ... 16  
[MinLowerLayerProtocol](#) ... 16

## P

[processInput](#) ... 18  
[Parser](#) ... 5  
[Parser](#) ... 6

## S

[setInputStream](#) ... 12  
[setInputStream](#) ... 15  
[setOutputStream](#) ... 12  
[setOutputStream](#) ... 15  
[ServeurProtocol](#) ... 17  
[ServeurProtocol](#) ... 17  
[ServiceGenerator](#) ... 6

## T

[transPat](#) ... 5  
[transPat](#) ... 5  
[transPat](#) ... 9  
[transPat](#) ... 9

## W

[writeMessage](#) ... 12  
[writeMessage](#) ... 16  
[writeMessage](#) ... 16