LUCENE - TOKEN

Introduction

Token represents text or word in a document with relevant details like its metadata *position, startoffset, endoffset, tokentypeanditspositionincrement*.

Class declaration

Following is the declaration for **org.apache.lucene.analysis.Token** class:

```
public class Token
   extends TermAttributeImpl
      implements TypeAttribute, PositionIncrementAttribute,
                 FlagsAttribute, OffsetAttribute,
                 PayloadAttribute, PositionLengthAttribute
```

Fields

- **static AttributeSource.AttributeFactory TOKEN_ATTRIBUTE_FACTORY** - Convenience factory that returns Token as implementation for the basic attributes and return the default impl *with " Impl " appended* for all other attributes.

Class constructors

S.N.	Constructor & Description
1	**Token** Constructs a Token will null text.
2	**Token*char[]startTermBuffer, inttermBufferOffset, inttermBufferLength, intstart, intend*** Constructs a Token with the given term buffer offset & length, start and end offsets
3	**Token*intstart, intend*** Constructs a Token with null text and start & end offsets.
4	**Token*intstart, intend, intflags*** Constructs a Token with null text and start & end offsets plus flags.
5	**Token*intstart, intend, Stringtyp*** Constructs a Token with null text and start & end offsets plus the Token type.
6	**Token*Stringtext, intstart, intend*** Constructs a Token with the given term text, and start & end offsets.
7	**Token*Stringtext, intstart, intend, intflags***

Constructs a Token with the given text, start and end offsets, & type.

8

Token*Stringtext, intstart, intend, Stringtyp*

Constructs a Token with the given text, start and end offsets, & type.

Class methods

S.N.	Method & Description

1

void clear

Resets the term text, payload, flags, and positionIncrement, startOffset, endOffset and token type to default.

2

Object clone

Shallow clone.

3

Token clone
char*[]*newTermBuffer, intnewTermOffset, intnewTermLength, intnewStartOffset, intnewEndOffset

Makes a clone, but replaces the term buffer & start/end offset in the process.

4

void copyTo*AttributeImpltarget*

Copies the values from this Attribute into the passed-in target attribute.

5

int endOffset

Returns this Token's ending offset, one greater than the position of the last character corresponding to this token in the source text.

6

boolean equals*Objectobj*

7

int getFlags

Get the bitset for any bits that have been set.

8

Payload getPayload

Returns this Token's payload.

9

int getPositionIncrement

Returns the position increment of this Token.

10

int getPositionLength

Get the position length.

11

int hashCode

12

void reflectWith*AttributeReflectorreflector*

This method is for introspection of attributes, it should simply add the key/values this attribute holds to the given AttributeReflector.

13

Token reinit
char*[]*newTermBuffer, intnewTermOffset, intnewTermLength, intnewStartOffset, intnewEndOffset

Shorthand for calling clear, CharTermAttributeImpl.copyBuffer*char*[], *int*, *int*, setStartOffset*int*, setEndOffset*int* setType*java. lang. String* on Token.DEFAULT_TYPE

14

Token reinit
char*[]*newTermBuffer, intnewTermOffset, intnewTermLength, intnewStartOffset, intnewEndOffset, StringnewType

Shorthand for calling clear, CharTermAttributeImpl.copyBuffer*char*[], *int*, *int*, setStartOffset*int*, setEndOffset*int*, setType*java. lang. String*

15

Token reinit*StringnewTerm, intnewStartOffset, intnewEndOffset*
Shorthand for calling clear, CharTermAttributeImpl.append*CharSequence*, setStartOffset*int*, setEndOffset*int* setType*java. lang. String* on Token.DEFAULT_TYPE

16

Token reinit*StringnewTerm, intnewTermOffset, intnewTermLength, intnewStartOffset, intnewEndOffset*

Shorthand for calling clear, CharTermAttributeImpl.append*CharSequence*, *int*, *int*, setStartOffset*int* , setEndOffset*int* setType*java. lang. String* on Token.DEFAULT_TYPE

17

Token reinit
StringnewTerm, intnewTermOffset, intnewTermLength, intnewStartOffset, intnewEndOffset, StringnewType

Shorthand for calling clear, CharTermAttributeImpl.append*CharSequence*, *int*, *int*, setStartOffset*int* , setEndOffset*int* setType*java. lang. String*

18

Token reinit*StringnewTerm, intnewStartOffset, intnewEndOffset, StringnewType*

Shorthand for calling clear, CharTermAttributeImpl.append*CharSequence*, setStartOffset*int*, setEndOffset*int* setType*java. lang. String*

19

void reinit*Tokenprototype*

Copy the prototype token's fields into this one.

20

void reinit*Tokenprototype, char*[]*newTermBuffer, intoffset, intlength*

Copy the prototype token's fields into this one, with a different term.

21

void reinit*Tokenprototype, StringnewTerm*

Copy the prototype token's fields into this one, with a different term.

22

void setEndOffset*intoffset*

Set the ending offset.

23

void setFlags*intflags*

24

void setOffset*intstartOffset, intendOffset*

Set the starting and ending offset.

25

void setPayload*Payloadpayload*

Sets this Token's payload.

26

void setPositionIncrement*intpositionIncrement*

Set the position increment.

27

void setPositionLength*intpositionLength*

Set the position length.

28

void setStartOffset*intoffset*

Set the starting offset.

29

void setType*Stringtype*

Set the lexical type.

30

int startOffset

Returns this Token's starting offset, the position of the first character corresponding to this token in the source text.

31

String type

Returns this Token's lexical type.

Methods inherited

This class inherits methods from the following classes:

- org.apache.lucene.analysis.tokenattributes.TermAttributeImpl
- org.apache.lucene.analysis.tokenattributes.CharTermAttributeImpl

- org.apache.lucene.util.AttributeImpl
- java.lang.Object