JavaScript is disabled on your browser.

jminusminus

Class TokenMgrError

- java.lang.Object
  - java.lang.Throwable
    - java.lang.Error
      - jminusminus.TokenMgrError

- All Implemented Interfaces:
  Serializable

```
public class TokenMgrError
extends Error
```

See Also:
Serialized Form

- **Field Summary**

Fields

Modifier and Type	Field and Description
(package private) int	**errorCode** Indicates the reason why the exception is thrown.
(package private) static int	**INVALID_LEXICAL_STATE** Tried to change to an invalid lexical state.
(package private) static int	**LEXICAL_ERROR** Lexical error occured.
(package private) static int	**LOOP_DETECTED**

	Detected (and bailed out of) an infinite loop in the token manager.
(package private) static int	**STATIC_LEXER_ERROR** An attempt wass made to create a second instance of a static token manager.

- **Constructor Summary**

Constructors

Constructor and Description

TokenMgrError()
TokenMgrError(boolean EOFSeen, int lexState, int errorLine, int errorColumn, String errorAfter, char curChar, int reason)
TokenMgrError(String message, int reason)

- **Method Summary**

Methods

Modifier and Type	Method and Description
protected static String	**addEscapes**(String str) Replaces unprintable characters by their espaced (or unicode escaped) equivalents in the given string
String	**getMessage**() You can also modify the body of this method to customize your error messages.
protected static String	**LexicalError**(boolean EOFSeen, int lexState, int errorLine, int errorColumn, String errorAfter, char curChar) Returns a detailed message for the Error when it is thrown by the token manager to indicate a lexical error.

- **Methods inherited from class java.lang.Throwable**

  addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

- **Methods inherited from class java.lang.Object**

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

- **Field Detail**

  - **LEXICAL_ERROR**

    static finalint LEXICAL_ERROR

Lexical error occured.

See Also:

    Constant Field Values

- **STATIC_LEXER_ERROR**

```
static finalint STATIC_LEXER_ERROR
```

An attempt wass made to create a second instance of a static token manager.

See Also:

    Constant Field Values

- **INVALID_LEXICAL_STATE**

```
static finalint INVALID_LEXICAL_STATE
```

Tried to change to an invalid lexical state.

See Also:

    Constant Field Values

- **LOOP_DETECTED**

```
static finalint LOOP_DETECTED
```

Detected (and bailed out of) an infinite loop in the token manager.

See Also:

    Constant Field Values

- **errorCode**

```
int errorCode
```

Indicates the reason why the exception is thrown. It will have one of the above 4 values.

- **Constructor Detail**

  - **TokenMgrError**

```
publicTokenMgrError()
```

  - **TokenMgrError**

```
publicTokenMgrError(Stringmessage,
            intreason)
```

  - **TokenMgrError**

```
publicTokenMgrError(booleanEOFSeen,
            intlexState,
            interrorLine,
            interrorColumn,
            StringerrorAfter,
            charcurChar,
            intreason)
```

- **Method Detail**

  - **addEscapes**

```
protected static finalStringaddEscapes(Stringstr)
```

Replaces unprintable characters by their espaced (or unicode escaped) equivalents in the given string

- **LexicalError**

```
protected staticStringLexicalError(booleanEOFSeen,
                   intlexState,
                   interrorLine,
                   interrorColumn,
                   StringerrorAfter,
                   charcurChar)
```

Returns a detailed message for the Error when it is thrown by the token manager to indicate a lexical error. Parameters : EOFSeen : indicates if EOF caused the lexicl error curLexState : lexical state in which this error occured errorLine : line number when the error occured errorColumn : column number when the error occured errorAfter : prefix that was seen before this error occured curchar : the offending character Note: You can customize the lexical error message by modifying this method.

- **getMessage**

```
publicStringgetMessage()
```

You can also modify the body of this method to customize your error messages. For example, cases like LOOP_DETECTED and INVALID_LEXICAL_STATE are not of end-users concern, so you can return something like : "Internal Error : Please file a bug report .... " from this method for such cases in the release version of your parser.

Overrides:

getMessage in class Throwable

- Summary:
- Nested |
- Field |
- Constr |
- Method

- Detail:
- Field |
- Constr |
- Method