JavaScript is disabled on your browser.

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/LexicalCategory.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-files/index-1.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/simplenlg/framework/InflectedWordElement.html)
* [Next Class](http://docs.google.com/simplenlg/framework/ListElement.html)
* [Frames](http://docs.google.com/index.html?simplenlg/framework/LexicalCategory.html)
* [No Frames](http://docs.google.com/LexicalCategory.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* [Enum Constants](#id.3znysh7)|
* Field |
* [Method](#id.2et92p0)
* Detail:
* [Enum Constants](#id.1t3h5sf)|
* Field |
* [Method](#id.4i7ojhp)

simplenlg.framework

## Enum LexicalCategory

* java.lang.Object
  + java.lang.Enum<[LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)>
    - simplenlg.framework.LexicalCategory

All Implemented Interfaces: java.io.Serializable, java.lang.Comparable<[LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)>, [ElementCategory](http://docs.google.com/simplenlg/framework/ElementCategory.html)

* public enum LexicalCategory  
  extends java.lang.Enum<[LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)>  
  implements [ElementCategory](http://docs.google.com/simplenlg/framework/ElementCategory.html)  
  This enumeration defines the different lexical components. The categories define the well understood role each word takes in language. For example, *dog* is a noun, *chase* is a verb, *the* is a determiner, and so on.Version: 4.0 Author: A. Gatt and D. Westwater, University of Aberdeen.

### Enum Constant SummaryEnum Constants

|  |
| --- |
| * + Enum Constant and Description |
| * + [**ADJECTIVE**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#ADJECTIVE) An adjective element. |
| * + [**ADVERB**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#ADVERB) An adverb element. |
| * + [**ANY**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#ANY) A default value, indicating an unspecified category. |
| * + [**AUXILIARY**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#AUXILIARY) An auxiliary verb element. |
| * + [**COMPLEMENTISER**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#COMPLEMENTISER) A complementiser element. |
| * + [**CONJUNCTION**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#CONJUNCTION) A conjunction element. |
| * + [**DETERMINER**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#DETERMINER) A determiner element often referred to as a specifier. |
| * + [**MODAL**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#MODAL) A modal element. |
| * + [**NOUN**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#NOUN) A noun element. |
| * + [**PREPOSITION**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#PREPOSITION) A preposition element. |
| * + [**PRONOUN**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#PRONOUN) A pronoun element. |
| * + [**SYMBOL**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#SYMBOL) The element represents a symbol. |
| * + [**VERB**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#VERB) A verb element. |

### Method SummaryMethods

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + boolean | * + [**equalTo**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#equalTo(java.lang.Object))(java.lang.Object checkObject) Checks to see if the given object is equal to this lexical category. |
| * + static [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) | * + [**valueOf**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#valueOf(java.lang.String))(java.lang.String name) Returns the enum constant of this type with the specified name. |
| * + static [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)[] | * + [**values**](http://docs.google.com/simplenlg/framework/LexicalCategory.html#values())() Returns an array containing the constants of this enum type, in the order they are declared. |

### Methods inherited from class java.lang.EnumcompareTo, equals, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

### Methods inherited from class java.lang.ObjectgetClass, notify, notifyAll, wait, wait, wait

### Enum Constant Detail

#### ANY public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) ANY A default value, indicating an unspecified category.

#### SYMBOL public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) SYMBOL The element represents a symbol.

#### NOUN public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) NOUN A noun element.

#### ADJECTIVE public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) ADJECTIVE An adjective element.

#### ADVERB public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) ADVERB An adverb element.

#### VERB public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) VERB A verb element.

#### DETERMINER public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) DETERMINER A determiner element often referred to as a specifier.

#### PRONOUN public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) PRONOUN A pronoun element.

#### CONJUNCTION public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) CONJUNCTION A conjunction element.

#### PREPOSITION public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) PREPOSITION A preposition element.

#### COMPLEMENTISER public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) COMPLEMENTISER A complementiser element.

#### MODAL public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) MODAL A modal element.

#### AUXILIARY public static final [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) AUXILIARY An auxiliary verb element.

### Method Detail

#### values public static [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)[] values() Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows: for (LexicalCategory c : LexicalCategory.values())   System.out.println(c); Returns:an array containing the constants of this enum type, in the order they are declared

#### valueOf public static [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) valueOf(java.lang.String name) Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)Parameters:name - the name of the enum constant to be returned. Returns:the enum constant with the specified name Throws: java.lang.IllegalArgumentException - if this enum type has no constant with the specified name java.lang.NullPointerException - if the argument is null

#### equalTo public boolean equalTo(java.lang.Object checkObject) Checks to see if the given object is equal to this lexical category. This is done by checking the enumeration if the object is of the type LexicalCategory or by converting the object and this category to strings and comparing the strings. For example, LexicalCategory.NOUN will match another LexicalCategory.NOUN but will also match the string *"noun"* as well.**Specified by:** [equalTo](http://docs.google.com/simplenlg/framework/ElementCategory.html#equalTo(java.lang.Object)) in interface [ElementCategory](http://docs.google.com/simplenlg/framework/ElementCategory.html) Parameters:checkObject - the object to be checked against. Returns:true if the object matches, false otherwise.

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/LexicalCategory.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-files/index-1.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/simplenlg/framework/InflectedWordElement.html)
* [Next Class](http://docs.google.com/simplenlg/framework/ListElement.html)
* [Frames](http://docs.google.com/index.html?simplenlg/framework/LexicalCategory.html)
* [No Frames](http://docs.google.com/LexicalCategory.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* [Enum Constants](#id.3znysh7)|
* Field |
* [Method](#id.2et92p0)
* Detail:
* [Enum Constants](#id.1t3h5sf)|
* Field |
* [Method](#id.4i7ojhp)