JavaScript is disabled on your browser.

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* [Class](http://docs.google.com/simplenlg/framework/LexicalCategory.html)
* Use
* [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
* Next
* [Frames](http://docs.google.com/index.html?simplenlg/framework/class-use/LexicalCategory.html)
* [No Frames](http://docs.google.com/LexicalCategory.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)

## 

Uses of Class

simplenlg.framework.LexicalCategory

* Packages that use [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)

|  |  |
| --- | --- |
| * Package | * Description |
| * [simplenlg.framework](#id.3znysh7) | * The framework package contains the base components required for running the SimpleNLG package. |
| * [simplenlg.lexicon](#id.2et92p0) |  |

### Uses of [**LexicalCategory**](http://docs.google.com/simplenlg/framework/LexicalCategory.html) in [**simplenlg.framework**](http://docs.google.com/simplenlg/framework/package-summary.html)Methods in [simplenlg.framework](http://docs.google.com/simplenlg/framework/package-summary.html) that return [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + static [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) | * + LexicalCategory.[**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)[] | * + LexicalCategory.[**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 in [simplenlg.framework](http://docs.google.com/simplenlg/framework/package-summary.html) with parameters of type [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + [NLGElement](http://docs.google.com/simplenlg/framework/NLGElement.html) | * + NLGFactory.[**createInflectedWord**](http://docs.google.com/simplenlg/framework/NLGFactory.html#createInflectedWord(java.lang.Object,%20simplenlg.framework.LexicalCategory))(java.lang.Object word, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) Create an inflected word element. |
| * + [NLGElement](http://docs.google.com/simplenlg/framework/NLGElement.html) | * + NLGFactory.[**createNLGElement**](http://docs.google.com/simplenlg/framework/NLGFactory.html#createNLGElement(java.lang.Object,%20simplenlg.framework.LexicalCategory))(java.lang.Object element, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) this method creates an NLGElement from an object If object is null, return null If the object is already an NLGElement, it is returned unchanged Exception: if it is an InflectedWordElement, return underlying WordElement If it is a String which matches a lexicon entry or pronoun, the relevant WordElement is returned If it is a different String, a wordElement is created if the string is a single word Otherwise a StringElement is returned Otherwise throw an exception |
| * + [NLGElement](http://docs.google.com/simplenlg/framework/NLGElement.html) | * + NLGFactory.[**createWord**](http://docs.google.com/simplenlg/framework/NLGFactory.html#createWord(java.lang.Object,%20simplenlg.framework.LexicalCategory))(java.lang.Object word, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) Creates a new element representing a word. |

* + Constructors in [simplenlg.framework](http://docs.google.com/simplenlg/framework/package-summary.html) with parameters of type [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)

|  |
| --- |
| * + Constructor and Description |
| * + [**InflectedWordElement**](http://docs.google.com/simplenlg/framework/InflectedWordElement.html#InflectedWordElement(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String word, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) Constructs a new inflected word using the giving word as the base form. |
| * + [**WordElement**](http://docs.google.com/simplenlg/framework/WordElement.html#WordElement(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) create a WordElement with the specified baseForm and category |
| * + [**WordElement**](http://docs.google.com/simplenlg/framework/WordElement.html#WordElement(java.lang.String,%20simplenlg.framework.LexicalCategory,%20java.lang.String))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category, java.lang.String id) create a WordElement with the specified baseForm, category, ID |

### Uses of [**LexicalCategory**](http://docs.google.com/simplenlg/framework/LexicalCategory.html) in [**simplenlg.lexicon**](http://docs.google.com/simplenlg/lexicon/package-summary.html)Methods in [simplenlg.lexicon](http://docs.google.com/simplenlg/lexicon/package-summary.html) with parameters of type [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html)

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + [WordElement](http://docs.google.com/simplenlg/framework/WordElement.html) | * + Lexicon.[**getWord**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#getWord(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) get a WordElement which has the specified base form and category |
| * + [WordElement](http://docs.google.com/simplenlg/framework/WordElement.html) | * + Lexicon.[**getWordFromVariant**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#getWordFromVariant(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String variant, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) returns a WordElement which has the specified inflected form and/or spelling variant that matches the specified variant, of the specified category |
| * + java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + XMLLexicon.[**getWords**](http://docs.google.com/simplenlg/lexicon/XMLLexicon.html#getWords(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) |
| * + java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + NIHDBLexicon.[**getWords**](http://docs.google.com/simplenlg/lexicon/NIHDBLexicon.html#getWords(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) |
| * + java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + MultipleLexicon.[**getWords**](http://docs.google.com/simplenlg/lexicon/MultipleLexicon.html#getWords(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) |
| * + abstract java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + Lexicon.[**getWords**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#getWords(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) returns all Words which have the specified base form and category |
| * + java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + XMLLexicon.[**getWordsFromVariant**](http://docs.google.com/simplenlg/lexicon/XMLLexicon.html#getWordsFromVariant(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String variant, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) |
| * + java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + NIHDBLexicon.[**getWordsFromVariant**](http://docs.google.com/simplenlg/lexicon/NIHDBLexicon.html#getWordsFromVariant(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String variant, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) |
| * + java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + MultipleLexicon.[**getWordsFromVariant**](http://docs.google.com/simplenlg/lexicon/MultipleLexicon.html#getWordsFromVariant(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String variant, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) |
| * + abstract java.util.List<[WordElement](http://docs.google.com/simplenlg/framework/WordElement.html)> | * + Lexicon.[**getWordsFromVariant**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#getWordsFromVariant(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String variant, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) returns Words which have an inflected form and/or spelling variant that matches the specified variant, and are in the specified category. |
| * + boolean | * + Lexicon.[**hasWord**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#hasWord(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) return true if the lexicon contains a WordElement which has the specified base form and category |
| * + boolean | * + Lexicon.[**hasWordFromVariant**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#hasWordFromVariant(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String variant, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) return true if the lexicon contains a WordElement which matches the specified variant form and category |
| * + [WordElement](http://docs.google.com/simplenlg/framework/WordElement.html) | * + Lexicon.[**lookupWord**](http://docs.google.com/simplenlg/lexicon/Lexicon.html#lookupWord(java.lang.String,%20simplenlg.framework.LexicalCategory))(java.lang.String baseForm, [LexicalCategory](http://docs.google.com/simplenlg/framework/LexicalCategory.html) category) General word lookup method, tries base form, variant, ID (in this order) Creates new word if can't find existing word |

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* [Class](http://docs.google.com/simplenlg/framework/LexicalCategory.html)
* Use
* [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
* Next
* [Frames](http://docs.google.com/index.html?simplenlg/framework/class-use/LexicalCategory.html)
* [No Frames](http://docs.google.com/LexicalCategory.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)