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/Form.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/features/Feature.html)
* [Next Class](http://docs.google.com/simplenlg/features/Gender.html)
* [Frames](http://docs.google.com/index.html?simplenlg/features/Form.html)
* [No Frames](http://docs.google.com/Form.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.1ksv4uv)

simplenlg.features

## Enum Form

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

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

* public enum Form  
  extends java.lang.Enum<[Form](http://docs.google.com/simplenlg/features/Form.html)>  
  An enumeration representing the different forms a verb and its associated phrase can take. The form is recorded under the Feature.FORM feature and applies to verbs and verb phrases.Version: 4.0 Author: A. Gatt and D. Westwater, University of Aberdeen.

### Enum Constant SummaryEnum Constants

|  |
| --- |
| * + Enum Constant and Description |
| * + [**BARE\_INFINITIVE**](http://docs.google.com/simplenlg/features/Form.html#BARE_INFINITIVE) The bare infinitive is the base form of the verb. |
| * + [**GERUND**](http://docs.google.com/simplenlg/features/Form.html#GERUND) In English, the gerund form refers to the usage of a verb as a noun. |
| * + [**IMPERATIVE**](http://docs.google.com/simplenlg/features/Form.html#IMPERATIVE) The imperative form of a verb is the one used when the grammatical mood is one of expressing a command or giving a direct request. |
| * + [**INFINITIVE**](http://docs.google.com/simplenlg/features/Form.html#INFINITIVE) The infinitive form represents the base form of the verb, with our without the particle *to*. |
| * + [**NORMAL**](http://docs.google.com/simplenlg/features/Form.html#NORMAL) Normal form represents the base verb. |
| * + [**PAST\_PARTICIPLE**](http://docs.google.com/simplenlg/features/Form.html#PAST_PARTICIPLE) Most verbs will have only a single form for the past tense. |
| * + [**PRESENT\_PARTICIPLE**](http://docs.google.com/simplenlg/features/Form.html#PRESENT_PARTICIPLE) The present participle is identical in form to the gerund and is normally used in the active voice. |

### Method SummaryMethods

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + static [Form](http://docs.google.com/simplenlg/features/Form.html) | * + [**valueOf**](http://docs.google.com/simplenlg/features/Form.html#valueOf(java.lang.String))(java.lang.String name) Returns the enum constant of this type with the specified name. |
| * + static [Form](http://docs.google.com/simplenlg/features/Form.html)[] | * + [**values**](http://docs.google.com/simplenlg/features/Form.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

#### BARE\_INFINITIVE public static final [Form](http://docs.google.com/simplenlg/features/Form.html) BARE\_INFINITIVE The bare infinitive is the base form of the verb.

#### GERUND public static final [Form](http://docs.google.com/simplenlg/features/Form.html) GERUND In English, the gerund form refers to the usage of a verb as a noun. For example, *I like* ***swimming***. In more general terms, gerunds are usually formed from the base word with *-ing* added to the end.

#### IMPERATIVE public static final [Form](http://docs.google.com/simplenlg/features/Form.html) IMPERATIVE The imperative form of a verb is the one used when the grammatical mood is one of expressing a command or giving a direct request. For example, ***Close*** *the door.*

#### INFINITIVE public static final [Form](http://docs.google.com/simplenlg/features/Form.html) INFINITIVE The infinitive form represents the base form of the verb, with our without the particle *to*. For example, *do* and *to do* are both infinitive forms of *do*.

#### NORMAL public static final [Form](http://docs.google.com/simplenlg/features/Form.html) NORMAL Normal form represents the base verb. For example, *kiss*, *walk*, *bark*, *eat*.

#### PAST\_PARTICIPLE public static final [Form](http://docs.google.com/simplenlg/features/Form.html) PAST\_PARTICIPLE Most verbs will have only a single form for the past tense. However, some verbs will have two forms, one for the simple past tense and one for the past participle (also knowns as passive participle or perfect participle). The part participle represents the second of these two forms. For example, the verb *eat* has the simple past form of *ate* and also the past participle form of *eaten*. Another example, is *write*, *wrote* and *written*.

#### PRESENT\_PARTICIPLE public static final [Form](http://docs.google.com/simplenlg/features/Form.html) PRESENT\_PARTICIPLE The present participle is identical in form to the gerund and is normally used in the active voice. However, the gerund is meant to highlight a verb being used as a noun. The present participle remains as a verb. For example, *Jim was* ***sleeping***.

### Method Detail

#### values public static [Form](http://docs.google.com/simplenlg/features/Form.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 (Form c : Form.values())   System.out.println(c); Returns:an array containing the constants of this enum type, in the order they are declared

#### valueOf public static [Form](http://docs.google.com/simplenlg/features/Form.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

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/Form.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/features/Feature.html)
* [Next Class](http://docs.google.com/simplenlg/features/Gender.html)
* [Frames](http://docs.google.com/index.html?simplenlg/features/Form.html)
* [No Frames](http://docs.google.com/Form.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.1ksv4uv)