

MiscDictionary

Inherits From: HashTable
Declared In: <misckit/MiscDictionary.h>

Class Description

This class is a specialized HashTable which associates objects with MiscStrings as keys. Internally, the MiscStrings are actually stored as NXAtoms, but this is hidden from you by overriding all the HashTable methods that deal with the keys. Simply use the id of a MiscString wherever a pointer to a key is required and things will work great. Refer to the HashTable class' documentation for a more complete description of the methods supported by this class.

Instance Variables

None added by this class.

Method Types

- init
- initCapacity:
- insertKey:value:
- isKey:
- nextState:key:value:
- removeKey:
- valueForKey:

Instance Methods

init

- **init**

Initializes the MiscDictionary with a default capacity of 500. Returns *self*.

See also: **±initCapacity:**

initCapacity:

- **initCapacity:**(int)*aCap*

Initializes the MiscDictionary with a capacity of *aCap*. Returns *self*.

See also: **± init**

insertKey:value:

- (void *)**insertKey:**(const void *)*aKey*
value:(void *)*aValue*

This method makes an association in the MiscDictionary. *aKey* should be a MiscString and *aValue* should be an **id**.

isKey:

- (BOOL)**isKey:**(const void *)*aKey*

Returns yes if the MiscString *aKey* is a key in the MiscDictionary.

nextState:key:value:

- (BOOL)**nextState:**(NXHashState *)*aState*
 key:(const void **)*aKey*
 value:(void **)*aValue*

See the documentation for the HashTable class for a description of this method. The only caveats are that *aKey* should be a pointer to the id of a MiscString and *aValue* should be a pointer to an **id**.

removeKey:

- (void *)**removeKey:**(const void *)*aKey*

Removes the MiscString key *aKey* from the MiscDictionary.

valueForKey:

- (void *)**valueForKey:**(const void *)*aKey*

Returns the **id** of the object that is associated with the MiscString key *aKey*.