MiscAbstraction

Inherits From: none   *(MiscAbstraction is a root class)*

Declared In: misckit/MiscAbstraction.h

Class Description

Not all classes need instances. A MiscAbstraction subclass can be used to group together an area of pervasive yet abstract functionality, such as Comparison or Measurement. Alternatively a MiscAbstraction can be employed as a "virtual factory object" for a cluster of closely related classes which do not share a common abstract superclass.

Instance Variables

None declared.

Method Types

Initializing the class	+ initialize
Identifying classes	+ name
	+ class
	+ superclass

Testing for protocol conformance   + conformsTo:

Posing	+ poseAs:
Dynamic loading	+ finishLoading:
	+ startUnloading

Class Methods

For a fuller description of the following methods please refer to the documentation for Object found in:
/NextLibrary/Documentation/NextDev/GeneralRef/01_RootClass/Classes/Object.rtf.

class
   + **class**

Returns **self**.

See also: + **name, superclass**

conformsTo:
   + (BOOL)**conformsTo:** (Protocol *)*aProtocol*

Returns YES if the receiving class conforms to *aProtocol*, and NO if it does not.

finishLoading:
   + **finishLoading:** (struct mach_header *)*header*

Declared but not implemented.

See also: + **startUnloading**

initialize
   + **initialize**

Implemented in subclasses to perform class-specific initialization.

name
   + (const char *)**name**

Returns a null-terminated string containing the name of the class.

See also: + **class, + superclass**

poseAs:
    + **poseAs:** *aClassObject*

Causes the receiving class to ªpose asº its superclass, the *aClassObject* class.

startUnloading
    + **startUnloading**

Declared but not implemented.

See also: + **finishUnloading**

superclass
    + **superclass**

Returns the class object for the receiver's superclass.

See also: + **class, + name**