domain	
	host
	name
	note
	type
	isReallyAPrinter
Retrieving specific information	acceptsBinary
	imageRectForPaper:
	pageSizeForPaper:
	isColor
	isFontAvailable:
	isValid
	languageLevel
	isOutputStackInReverseOrder
Querying the NXPrinter tables	booleanForKey:inTable:
	dataForKey:inTable:length:
	floatForKey:inTable:
	intForKey:inTable:
	rectForKey:inTable:

(BOOL)acceptsBinary

stringForKey:inTable:

(void *)dataForKey:(const char *)key
inTable:(const char *)table
length:(int *)bytes

Returns a pointer to untyped data for the given key in the given table. The length of the data, in by
reference in bytes.   key should be formed as described in the class description, above.

stringForKey:inTable:

(const char *)domain

Returns the name of the domain in which the NXPrinter's printer entry resides. If the object doesn
printer, this returns a pointer to NULL.

(float)floatForKey:(const char *)key inTable:(const char *)table

Returns a floating-point value for the given key in the given table returns 0.0 if the value, which is
can't be coerced to float. key should be formed as described in the class description, above.

stringForKey:inTable:

(const char *)host

Returns the name of the host to which the printer is connected. If the object doesn't represent an a
returns a pointer to NULL.

(NXRect)imageRectForPaper:(const char *)paperType

Returns the printing rectangleÐthe area of the page that's available for printingÐfor the named pa
of paper type names depends on the NXPrinter's type typical names include ªLegalº, ªLetterº, ªA4

pageSizeForPaper:

(BOOL)isColor

Returns YES if the NXPrinter can print in color. Otherwise returns NO.

(BOOL)isFontAvailable:(const char *)fontName

Returns YES if the named font is available to the NXPrinter otherwise returns NO.   Font names ar invocation of Font's useFont: method examples include ªHelvetica- Boldº, ªTimes- Romanº, and ªC

(BOOL)isKey:(const char *)key inTable:(const char *)table

Returns YES if key is a key to table (which must name one of the NXPrinter tables listed in the cla

statusForTable:

(BOOL)isOutputStackInReverseOrder

Returns YES if the printer outputs pages in reverse page order, otherwise returns NO. By being pr the pages in the resulting output stack will be in the correct (first-to-last) order (assuming that the p face-up).

(BOOL)isReallyAPrinter

Returns YES if the NXPrinter corresponds to an actual printer device. Otherwise returns NO.

(BOOL)isValid

Returns YES if the NXPrinter is validÐif its internal state matches physical reality. Otherwise ret important only for NXPrinters that correspond to actual printers.

statusForTable:

(int)languageLevel

Returns 1 or 2 as the NXPrinter recognizes the PostScript Language Level I or Level II.

(const char *)name

Returns the name of the NXPrinter. If the object doesn't represent an actual printer, this returns a p

(NXSize)pageSizeForPaper:(const char *)paperType

Returns the size of the page for the named paper type. The selection of paper type names depends
type typical names include ªLegalº, ªLetterº, ªA4º, and ªB5º.

imageRectForPaper:

(NXRect)rectForKey:(const char *)key inTable:(const char *)table

Returns an NXRect for the given key in the given table. The individual fields are set to 0.0 if the v
ASCII text, can't be fit into an NXRect structure. key should be formed as described in the class d

stringForKey:inTable:

(NXSize)sizeForKey:(const char *)key inTable:(const char *)table

Returns an NXSize for the given key in the given table. The individual fields are set to 0.0 if the v
ASCII text, can't be fit into an NXSize structure. key should be formed as described in the class d

stringForKey:inTable:

(int)statusForTable:(const char *)table

Returns one of the following constants to indicate the status of the given table:

isValid

(const char *)stringForKey:(const char *)key inTable:(const char *)table

Returns a pointer to the ASCII text that corresponds to key in the given table. If the table contains
with this key, the value of the first entry is returned. A pointer to NULL is returned if the table do
precisely matches key. See the class description, above, for more information on this method.

stringListForKey:inTable:

(const char **)stringListForKey:(const char *)key inTable:(const char *)table

Returns a pointer to an array of strings each string gives the ASCII text that corresponds to an entr
in the given table. If key names a main keyword for which there are (in the table) option keyword
contains the option keywords. See the class description, above, for more information on this meth

Note that it's the invoker's responsibility to free the array that's returned by this method, but not th

stringForKey:inTable: