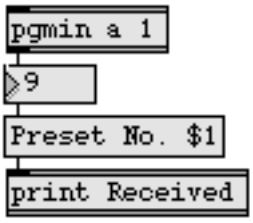
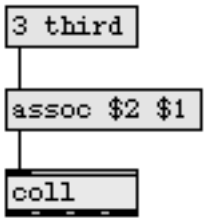
Arguments: $ and #, Changeable Arguments to Objects

$ in a message box

The dollar sign ($) is a special character which can be used in a **message** box to indicate a changeable argument. When the **message** box contains a $ and a number in the range 1-9 (such as $2) as one of its arguments, that argument will be replaced by the corresponding argument in the incoming message before the **message** box sends out its own message.

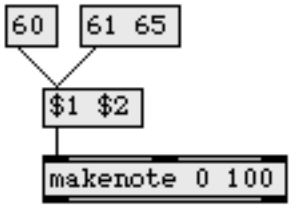


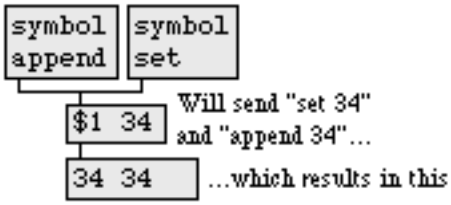


In the left example above, the $1 argument in the **message** box is replaced by the number received in the inlet (in this case 9) before the message is sent out. The message printed in the Max window will read Received: Preset No. 9.

The right example shows that both symbols and numbers can replace changeable arguments. It also shows that changeable arguments can be arranged in any order in the **message** box, making it a powerful tool for rearranging messages. In the example, the message assoc third 3 is sent to the **coll** object.

When a **message** box is triggered without receiving values for all of its changeable arguments (for instance, when it is triggered by a bang), it uses the most recently received values. The initial value of all changeable arguments is 0.





In the left example above, a message of 60 will initially send 60 0 to the **makenote** object. After the 61 65 message has been received, however, the number 65 will be stored in the $2 argument, so a message of 60 will send 60 65 to **makenote**.

A **message** box will not be triggered by a word received in its inlet (except for bang), unless the word is preceded by the word symbol. In such a case, the $1 argument will be replaced

by the word, and not by symbol. In the right example, the $1 argument is replaced by either set or append, and the message set 34 or append 34 is sent to the next **message** box.

To include a special character such as a dollar sign in a message without it having a special meaning, precede the character with a backslash (\).

$ in an object box

A changeable $ argument can also be used in some object boxes, such as the **expr** and **if** objects. In these objects, the $ must be followed immediately by the letter i, f, or s, indicating whether the argument is to be replaced by an int, a float, or a symbol.

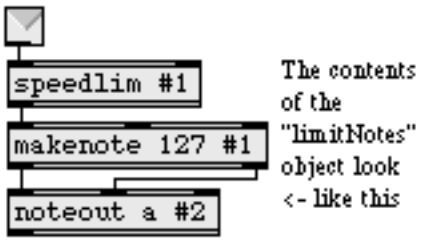
If the message received in the inlet does not match the type of the changeable argument (for example, if an int is received to replace a $f argument), the object will try to convert the input to the proper type. The object cannot convert symbols to numbers, however, so an error message will be printed if a symbol is received to replace a $i or $f argument. Other objects in which a $ argument is appropriate include **sxformat** and **vexpr**.

in object and message boxes

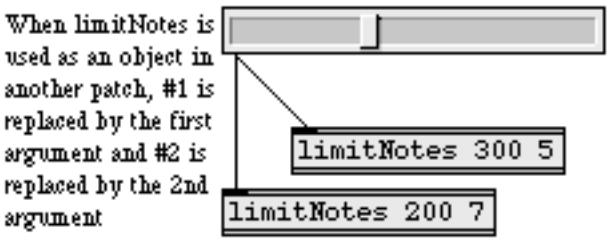
When you are editing a patcher which will be used as a subpatch within another Patcher, message boxes and most object boxes in the subpatch can be given a changeable argument by typing in a pound sign and a number (for example, #1) as an argument. Then, when the subpatch is used inside another Patcher, an argument typed into the object box in the Patcher replaces the # argument inside the subpatch.

In this way, **patcher** objects and your own objects can require typed in arguments to supply them with information, just as many Max objects do. A symbol such as #1 is a changeable argument, and is replaced by whatever number or symbol you type in as the corresponding argument when you use the patch as an object inside another patch. A changeable argument cannot be used to supply the name of an object itself, but can be used as an argument anywhere inside your object.

In the following example, arguments typed into the **limitNotes** object boxes supply values to the objects inside **limitNotes**. When the **hslider** is moved, one **limitNotes** object plays a note every 300 milliseconds on MIDI channel 5, and the other plays a note every 200ms on MIDI channel 7.

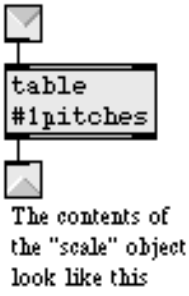


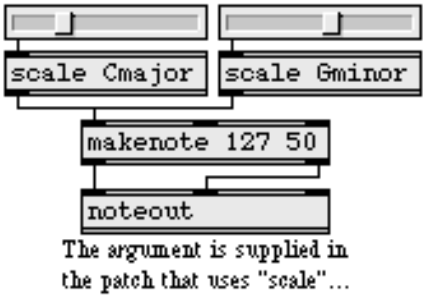
These are Max objects

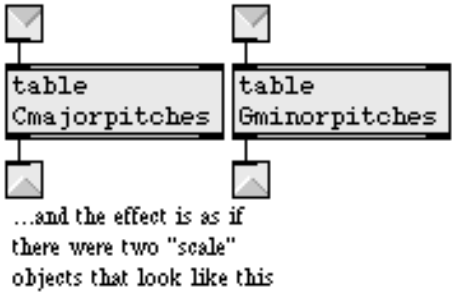


limitNotes is a patch saved as a document

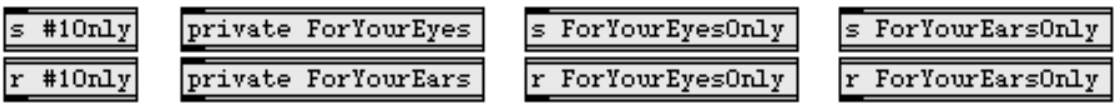
A pound sign and a number can even be part of a symbol argument, providing variations on a name, provided that the changeable argument is the first part of the symbol. In the example below, the #1 part of the changeable argument inside **scale** is replaced by the argument in the patch that uses **scale**. The **scale** objects will each use a different pre-saved **table** file, producing different results.







The same technique can be used to give unique names to **send** and **receive** objects in a subpatch, making the exchange of messages between them private (local to that one instance of the subpatch).

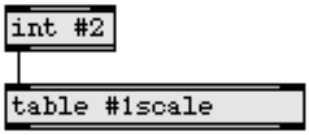


If these objects exist in a patch named private, *and the patch is used for two subpatches like this,* *the objects appear with this name in one patch,* *and with a unique name in the other.*

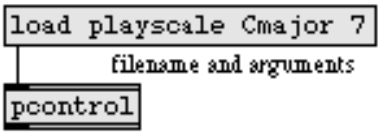
Arguments

$ and #, Changeable Arguments to Objects

When opening a patcher file automatically with the **load** message to a **pcontrol** object, changeable **#** arguments inside the patch being loaded can be replaced by values that are provided as additional arguments in the **load** message, as in the example below.



If these objects exist in a patch



and this message is sent to a pcontrol object,

the patch will open with objects looking like this.

#0 has a special meaning. It can be put at the beginning of a symbol argument, transforming that argument into an identifier unique to each patcher (and its subpatchers) when the patcher is loaded. This allows you to open several copies of a patcher containing objects such as **send** and **receive** without having the copies interfere with each other.

See Also

expr	Evaluate a mathematical expression
message	Send any message
pcontrol	Open and close subwindows within a patcher
Punctuation	Special characters in objects and messages