

AUD6204 - Programming Environments
Lesson 1.2

Basic PD Objects



Figure 1: The **'print'** object prints any input into the PD Console window. We will use print to debug and analyse our code.

Open the PD Window by pressing **'cmd+r'**, clear it by pressing **'cmd+shift+l.'**



Figure 2: The **'message'** object displays and outputs any given message. It also has the ability to handle specific arguments.

We can create a message box quickly by simply pressing **'cmd+2'** on our keyboard.



Figure 3: The **'comment'** object displays any given message. We use comments to annotate our code - this is useful for labelling interface objects. It also helps other people to understand your code, and you to remember what your code does.

We can create a comment box quickly by simply pressing **'cmd+5'** on our keyboard.



Figure 4: The **'bang'** object blinks and outputs a 'bang' (i.e. a trigger) when it receives any message or is clicked. We will use 'bang' objects to trigger events (e.g. samples).

We can create a button quickly by simply pressing **'cmd+shift+b'** on our keyboard.

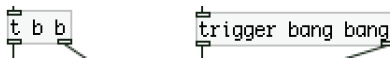


Figure 5: The **'trigger'** object outputs any input received in order from right to left and formatted according to the arguments specified. We can simply use 't' as a shorthand for trigger (right side of image).

NB **Always** use a trigger when connecting a single output to multiple different inputs!... Always...



Figure 6: The ‘**number**’ object displays, inputs, and outputs numbers (both integers and floating point).

We can create a number box quickly by simply pressing ‘**cmd+3**’ on our keyboard.

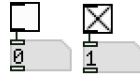


Figure 7: The ‘**toggle**’ object acts as a switch. It outputs a ‘1’ when on and a ‘0’ when off.

We can create a toggle quickly by simply pressing ‘**cmd+shift+t**’ on our keyboard.

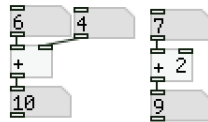


Figure 8: The ‘**+**’ object adds two numbers and outputs the result.

NB output will only be triggered when input is received to the left inlet!

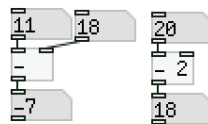


Figure 9: The ‘**-**’ object subtracts the number received in the right inlet from the number received in the left inlet, and outputs the result.

NB output will only be triggered when input is received to the left inlet!

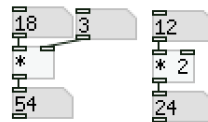


Figure 10: The ‘*****’ object multiplies two numbers and outputs the result.

NB output will only be triggered when input is received to the left inlet!

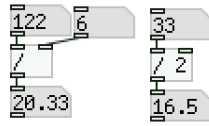


Figure 11: The ‘/’ object divides the number received in the left inlet by the number received in the right inlet, and outputs the result.

NB output will only be triggered when input is received to the left inlet!

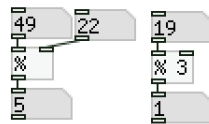


Figure 12: The ‘%’ (modulo) object takes two numbers. Divides the right by the left and outputs the **remainder**.

NB output will only be triggered when input is received to the left inlet!

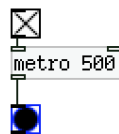


Figure 13: The ‘**metro**’ object acts as a metronome outputting a ‘bang’ at specified intervals. The interval is specified by the argument.

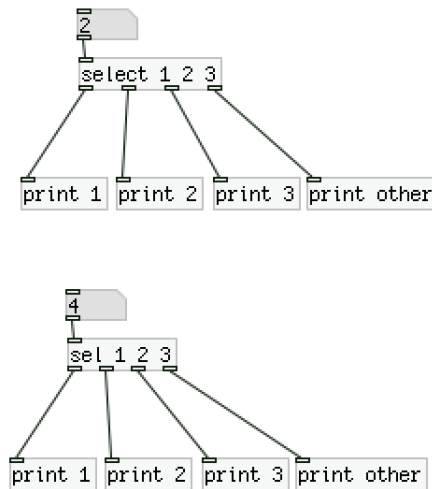


Figure 14: The **select** object selectively outputs a bang in response to any input which matches its argument. Non-matching input is output from the right-most outlet. We can simply use **sel** as a shorthand for trigger (right side of image).

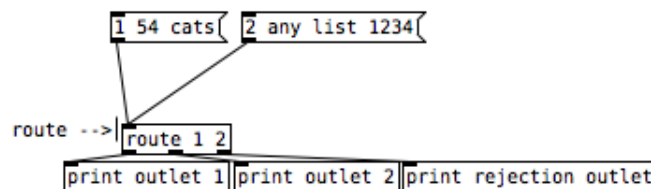


Figure 15: The **route** object sends messages out different outlets according to their first element. Route checks the first element of a message against each of its arguments, which may be numbers or symbols (but not a mixture of the two.). If a match is found, the rest of the message appears on the corresponding outlet. If no match, the message is repeated to the last "rejection" outlet. The number of outlets is the number of arguments plus one.