

## Praat Scripting 10 Demo window

The `Demo window` in Praat is very similar to the *Picture Window*. You can define viewports (but note the different basic grid which is 100 x 100 and not 12 x 12), use `Draw` and `Paint` commands as well as other arguments such as `Axes...`, `Text...`, `Line width...` etc.

The biggest difference between the `Demo` and `Picture` window is the fact that it allows interactions with the user. The `Demo` window can only be created through a script and used for demonstrations, presentations, simulations and adaptive experiments.

Look at the following Praat script. This will generate the string *Hello World!* and will write it in the middle of the `Picture Window` using a red color.

```

1 Erase all
2 Select outer viewport: 0, 10, 0, 10
3 Axes: 0, 100, 0, 100
4 Red
5 Text: 50, "centre", 50, "half", "Hello world!"

```

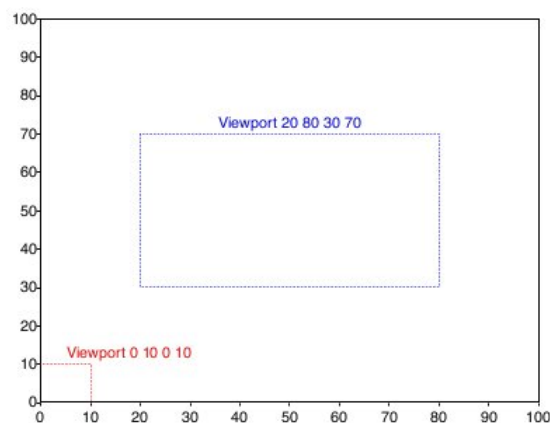
You can also use the above arguments and generate the output in the `Demo` window. To do so, use `demo` in front of all commands.

```

1 demo Erase all
2 demo Select outer viewport: 0, 10, 0, 10
3 demo Axes: 0, 100, 0, 100
4 demo Red
5 demo Text: 50, "centre", 50, "half", "Hello world!"

```

In contrast to the `Picture` window, the output *Hello World!* is now displayed in the the lower left corner. That is because the basic grip is 100 x 100 and the script uses only a small part of it (see Figure 1)<sup>1</sup>.



<sup>1</sup>Mayer, 2012: *Phonetische Analysen mit Praat*, p. 225; [http://praatpfanne.lingphon.net/downloads/praat\\_manual.pdf](http://praatpfanne.lingphon.net/downloads/praat_manual.pdf)

For most applications you will want some user input, e.g. click on things in the Demo window or press any keys. To include these actions in your script, do the following...

```

1 demo Erase all
2 demo Select outer viewport: 0, 10, 0, 10
3 demo Axes: 0, 100, 0, 100
4 demo Red
5 demo Text: 50, "centre", 50, "half", "Hello world!"
6 while demoWaitForInput ()
7     if demoClicked ()
8         goto page2
9     endif
10 endwhile
11 label page2
12 demo Erase all
13 demo Select outer viewport: 90, 100, 90, 100
14 demo Axes: 0, 100, 0, 100
15 demo Blue
16 demo Text: 50, "centre", 50, "half", "Hello world!"

```

The script above shows the red rectangle in the lower left corner. After any mouse click by the user the output changes into a blue rectangle in the upper right corner.

rows 6-11	These rows are responsible for the user input, in fact to recognize mouse clicks within the Demo window
row 6	<code>while</code> starts an infinite loop until an argument causes to leave the loop
row 7	the argument that causes to leave the loop is the mouse click in the Demo window
row 8	if the user clicks in the window the loop will be left and the script continues at a given position, here <code>page2</code> which will be defined later in the script (see row 11)
row 11	label that defines the position in which the script should continue after it waited for an input; therefore a script can also continue at a position earlier than the used while loop

If you want the user to use a key on the keyboard, use the following...

```

1 while demoWaitForInput ()
2     if demoKeyPressed ()
3         goto page2
4     endif
5 endwhile

```

In order to define a specific key or the a specific area that has to be clicked in, you have to specify those. For a particular area you have to define the coordinates of the grid (and therefore create clickable *buttons*).

```

1 demo Erase all
2 demo Select inner viewport: 0, 10, 0, 10
3 demo Axes: 0, 100, 0, 100
4 demo Paint rectangle: "Red", 0, 100, 0, 100
5 while demoWaitForInput ()
6     if demoClicked ()
7         if demoClickedIn (0, 100, 0, 100)
8             goto page2
9         endif
10    endif
11 endwhile
12 label page2
13 demo Erase all
14 demo Select inner viewport: 90, 100, 90, 100
15 demo Axes: 0, 100, 0, 100
16 demo Paint rectangle: "Blue", 0, 100, 0, 100
17 while demoWaitForInput ()
18     if demoClicked ()
19         if demoClickedIn (0, 100, 0, 100)
20             goto page3
21         endif
22     endif
23 endwhile
24 label page3
25 demo Erase all
26 demo Select inner viewport: 0, 100, 0, 100
27 demo Axes: 0, 100, 0, 100
28 demo Black
29 demo 18
30 demo Text: 50, "centre", 50, "half", "End"

```

And for a specific key...

```

1 while demoWaitForInput ()
2     if demoKeyPressed ()
3         if demoKey$ () = " "
4             goto page3
5         endif
6     endif
7 endwhile

```

### Exercise 10

Write a script that creates three tones (New → Sound → Create Sound as pure tone / Create Sound from formula) with

1. 10.000 Hz
2. 15.000 Hz
3. 20.000 Hz.

Create two buttons (red and green). If the user is able to hear the tone the green button should be pressed. If not, the red button. Print the frequency of the tones and the information whether it could be heard in a textfile. Make sure the user knows when the next sound was played.

Tip: `demoWaitForInput` is the place where your drawings will typically be painted on the screen. If you want painting to happen earlier, you can use `demoShow ( )`