**PsychoPy Question Library (Qlib) Manual**

Updated: 2/13/2012

The PsychoPy Question Library (qLib) is a library of functions that simplify the creation of questionnaire items in PsychoPy. Once the qLib has been imported, the functions from this library can be called in any PsychoPy experiment. QLib contains 8 customizable question types:

* Text Dialog: Scrolling text entry item
* Slider: Continuous response item
* Scale: A horizontally-oriented, (single response) multiple-choice item
* Bars: A comparative rating item
* Choice: A vertically-oriented, (single response) multiple-choice item
* MultiChoice: A multiple response multiple choice item
* Text Input: A multi-line text entry item
* Text Field: A single line text entry item (can limit allowed response length)

More information on these functions is provided below. NOTE: The **how to call the function** examples display the default settings and will only run if a parent window is specified.

**Importing and Using qLib**

In order to use qLib, the qLib folder must be saved in the same location as your PsychoPy program. For example, if the program example.py were moved outside of the original folder qLib2012, the program will not run. However, if the subfolder ‘qLib’ were moved to the same location, example.py would run successfully.

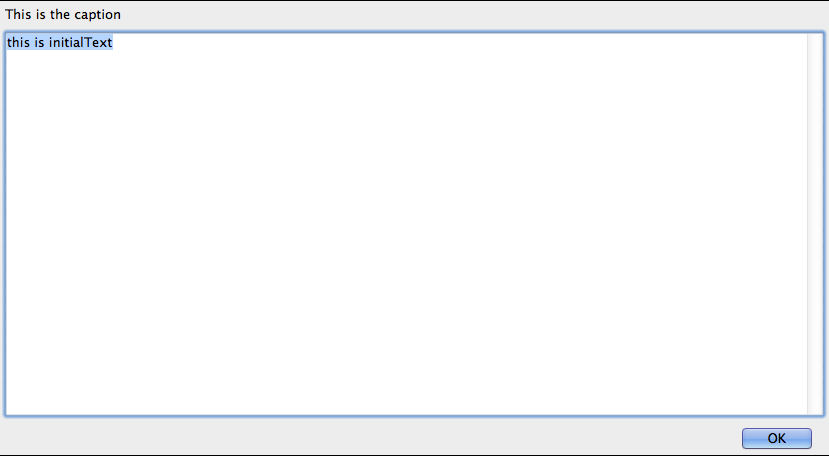
To call qLib functions within your program, you must first import qLib. To do this you must type:

from qLib.qLib import \*

If you do not include this step, PsychoPy will not recognize any of the functions in qLib as available functions.

**Descriptions of qLib Functions**

**TEXT DIALOG:** This function is a dialog box than can be used to record text entry. The box appears in front of the psychoPy window and may be best for recording long (multi-line or multi-paragraph) responses. This function allows for scrolling and has a read-only function that is convenient for presenting large amounts of text on one page. The function returns a tuple containing the text entered and response time.



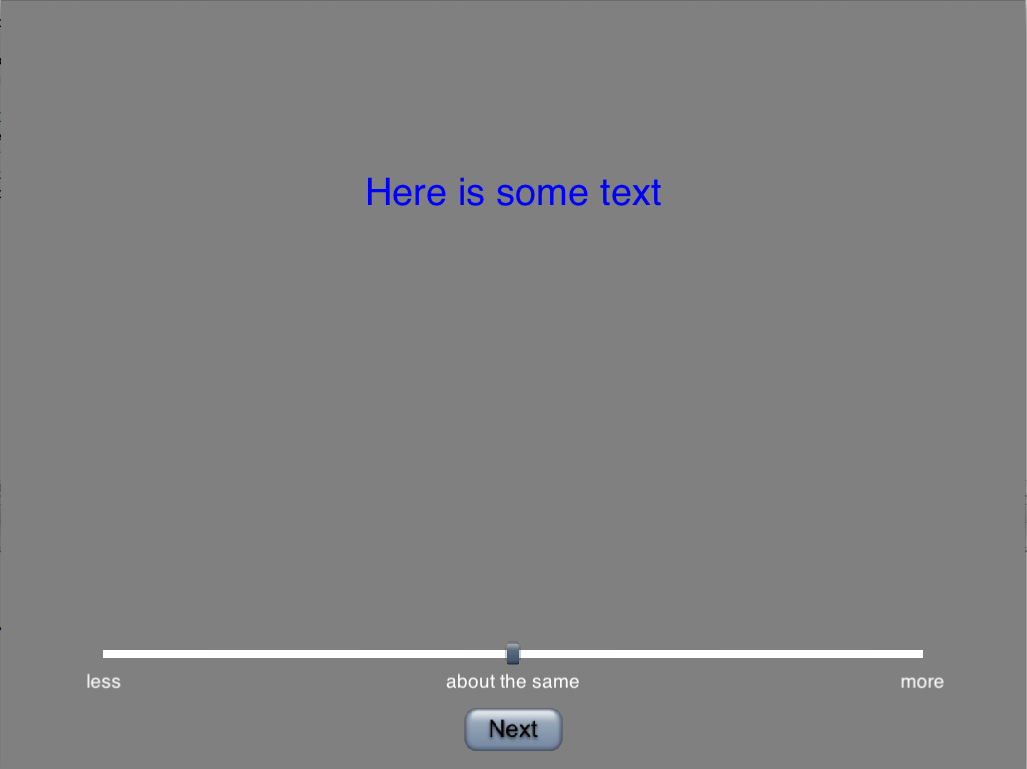
**To call this function:**

textDialog(window, clock=None, wsize=(.8,.5), title='',caption='', initialText='enter text here', select=None,readOnly = False)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *clock:*if provided, the specified clock will be used for timing, if not, a new one will be created.
* *wsize:*tuple indicating the size of the total dialog box (including boarder) in ‘norm’ units (2,2 should be full screen).
* *caption:* string, appears in the border around the actual text entry. Could be used to display a question or prompt.
* *initialText:*Text displayed in the entry window at outset. By default highlighted (so any key press will delete this). If no response is given, this text will be returned as the participant response.
* *select*: Boolean (True/False) variable specifying whether the initialText should be highlighted when the dialog box is presented. If True, text is highlighted. If false, initialText is not highlighted and cursor is set immediately after initialText. By default this is True.
* *readOnly***:** Boolean(True/False) variablespecifying whether text displayed in dialog box can be changed. If True, text can be entered. If false, initialText cannot be deleted and text cannot be entered in the dialog box.

**SLIDER QUESTION:** This is a “slider” question. The slider question presents a trial with a slider for a response. The subject can move the slider back and forth, with or without feedback, and then selects a value by clicking the Next button. The slider returns the value of the slider when the Next button is clicked and the response time.



**To call this function:**

slider(window, drawList = [], clock = None, width = .8, limits = [0,100], start = None, labels = [‘left’,’right’], snap2mouse = False, snap2Labels = False, feedback = False, fedbcakDigits = 2, feedbackColor = ‘lightblue’, labelColor = ‘white’, sliderLoc = -.7, forceChoice = True)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *drawList:* A list of (drawable) objects to be presented with the slider, can include movies.
* *clock:* if provided, the specified clock will be used for timing, if not, a new one will be created.
* *width:* Width (as a proportion of the window) of the slider line. A width of .8 is 80% the width of the screen.
* *limits:* List of 2 numbers that lists the upper and lower bounds for the slider (0-100 means that responses will be recorded on a 0-100 scale)
* *start:* Specifies the start position of the slider bar. By default starts at the middle (specification should be w/in limits set by the limits argument)
* *labels:* Labels to be used on the slider. Can use as many labels as you like, the program will evenly space and display labels below the slider line. (2 labels would label the left and right sides of the scale, three would label the left, middle and right)
* *snap2mouse:* Boolean (True/False) variable that specifies whether a line click will move the slider bar to that position on the slider line. If True, this will happen. If false, the slider bar can only be moved by clicking and dragging the slider bar.
* *Snap2labels:* Boolean (True/False) variable. When True, specifies that the slider can only switch between labels and not rest at any other ‘in-between’ point on the slider line(False by default).
* *Feedback:* Boolean (True/False) variable. When true provides feedback as to the current position of the slider (in units defined by the limit argument)
* *feedbackDigits:* number of digits to which the feedback number is rounded. This also determines how the responses is returned.
* *feedbackColor:* color in which feedback is displayed (if feedback is True)
* *labelColor:* Color of labels
* *sliderLoc:* Vertical position of slider line on the screen (in ‘norm’ units).
* *forceChoice:* When true does not display the ‘Next’ button until a participant has moved the slider from its starting position

**SCALE QUESTION:** The scale question allows for a single response by clicking on one of a specified number of buttons arranged horizontally across the screen. Participants respond by clicking one of the response buttons then the ‘Next’ button. Returns button number and response time.



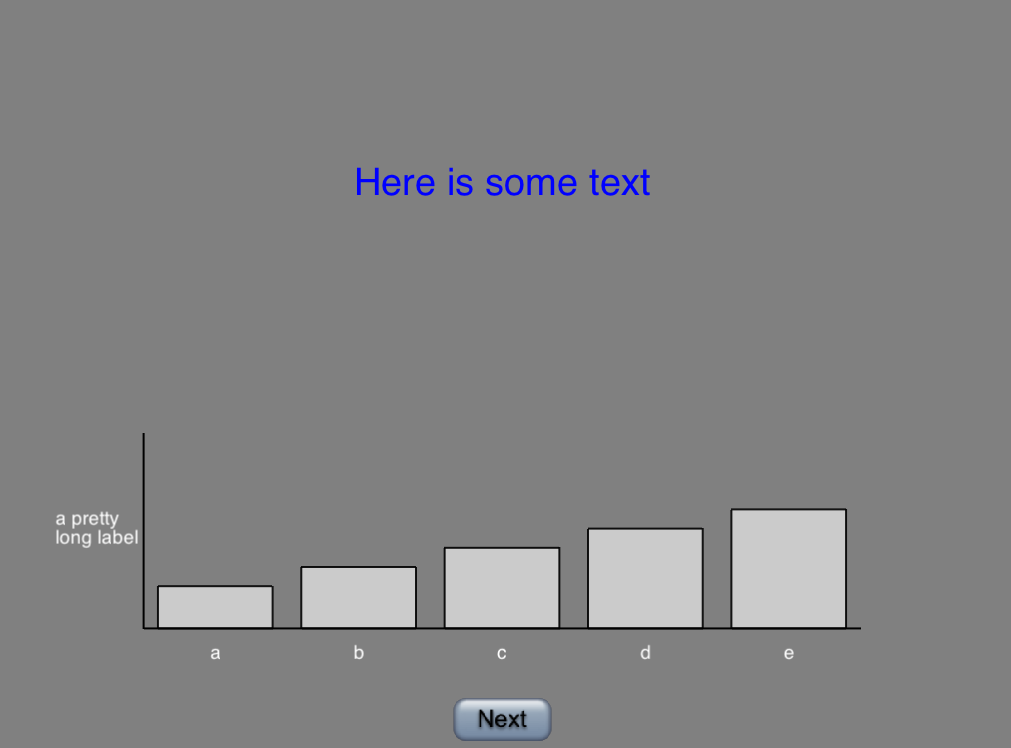
**To call this function:**

scale(window, drawList = [], clock = None, width = .8, labels = [‘left’, ‘right’], labelColor = ‘white’, nButtons = 5, numberButtons = False, forceChoice = True, scaleLoc = -.6)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *drawList:* A list of (drawable) objects to be presented with the scale, can include movies.
* *clock:* if provided, the specified clock will be used for timing, if not, a new one will be created.
* *width:* Width (as a proportion of the window) of the combined response buttons. A width of .8 is 80% the width of the screen.
* *labels:* Labels to be used on the scale items. Can use as many labels as you like, the program will evenly space and display labels below the scale buttons. (2 labels would label the left and right sides of the scale, three would label the left, middle and right)
* *labelColor:* Color of labels
* *nButtons:* number of response buttons to display (they will be evenly spaced across the screen width you specified with the width argument).
* *numberButtons:* Boolean (True/False) variable. If True, numbers the buttons starting with 1 on the far left.
* *forceChoice:* Boolean (True/False) variable. If True, only presents the next button after a selection has been made.
* *ScaleLoc:* Vertical location of the scale on the screen (in norm units).

**BARS QUESTION:** The bars questions presents a trial with a bar chart for response. Participants can click and drag the bars to a desired height and click the next button to end the trial. Returns heights of the bars and response time.



**To call this function:**

bars(window, drawList = [], clock = None, width = .8, height = .8, labels = None, labelColors = None, yLabels= None, yLabelColors = None, barColors = None, nBars = 5, limits = [0.0,100.0], defaultHeight = None, drawAxes = True, forceChoice = True)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *drawList:* A list of (drawable) objects to be presented with the trial, can include movies.
* *clock:* if provided, the specified clock will be used for timing, if not, a new one will be created.
* *width:* Width (as a proportion of the window) of the bar chart relative to the window (default .8 or 80% of the screen).
* *Height:* Height of the barchart in norm units
* *Labels:* A list of the labels to be evenly spaced between ends of the bar chart (x-axis).
* *labelColors:* Color of labels
* *yLabels:* list of the labels to be evenly spaced on the y-axis
* *yLabelColors:* color of yLabels
* *barColors:* a list of colors for the bars (if background is white, default is gray).
* *nBars:* Number of bars displayed
* *limits:* Limits for high and low points of y-axis. Will also be range of possible responses
* *defaultHeight:* Height (w/in limits specified above) at which bars begin (can also be list so different bars have different starting points).
* *drawAxes:* Boolean (True/False) variable. When true draws X and Y axes for the bar chart.
* *forceChoice:* Boolean(True/False) variable. When true requires the subject to move a bar before the ‘Next’ button appears.

**CHOICE QUESTION:** This is a “choice” question. The choice question presents a trial with a set of radio buttons. Only one radio button may be selected. Choice question returns the number of the selected button or None if no buttons are selected.



**To call a choice question:**

choice(window,drawList = [], vPos=0.2,labels = [], labelColor = ‘white’, labelSize = .1, forceChoice = True)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *clock:* If provided, the psychopy clock object will be used for timing. If not, one will be created.
* *drawList:* A list of objects that should be drawn along with the slider
* *vPos:* The vertical position of the first item on the list of radio buttons. The buttons will be spaced evenly between this location ant the Next button at the bottom of the window. (default 0.0)
* *hPos:* The horizontal position of the left edge of the radio button items (default -0.2)
* *labels:* A list with the radio button labels to be placed evenly spaced between vPos and the Next button (default ['a','b','c','d','e'])
* *labelColor:* Color of the label text (default 'white')
* *labelSize:* Size of the text of the labels (default 0.1)
* *forceChoice:* If True, require the subject to select a radio button before displaying the Next button. (default True)

**MULTICHOICE QUESTION:** This is a “multichoice” question. The multichoice question presents a trial with a set of checkboxes. Multiple checkboxes may be selected. Multichoice returns a list of the numbers of boxes selected or an empty list if no boxes are selected.



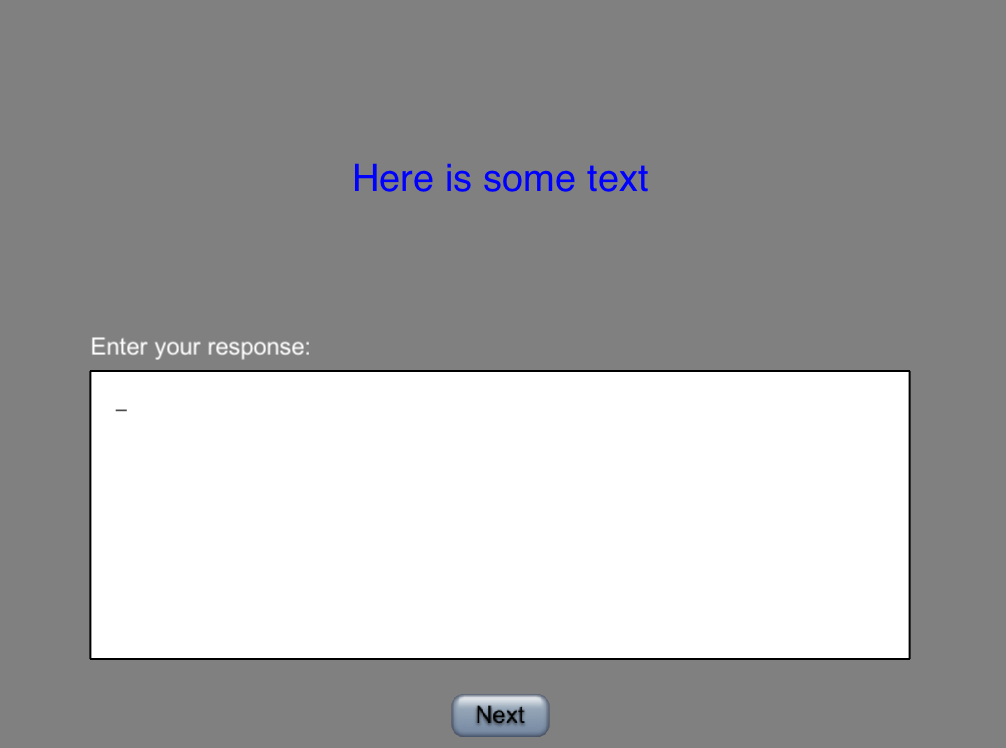
**To call a multichoice question:**

multiChoice(window, drawList = [], vPos=0.2,labels = [], labelColor = ‘white’, labelSize = .1, forceChoice = True)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *clock:* If provided, the psychopy clock object will be used for timing. If not, one will be created.
* *drawList:* A list of objects that should be drawn along with the trial
* *vPos:* The vertical position of the first item on the list of check boxes. The boxes will be spaced evenly between this location ant the Next button at the bottom of the window. (default 0.0)
* *hPos*: The horizontal position of the left edge of the check box items (default -0.2)
* *labels:* A list with the check box labels to be placed evenly spaced between vPos and the Next button (default ['a','b','c','d','e'])
* *labelColor*: Color of the label text (default 'white')
* *labelSize:* Size of the text of the labels (default 0.1)
* *forceChoice:* If True, require the subject to select at least one box before displaying the Next button. (default True)

**TEXT INPUT QUESTION**: Similar to the text dialog function, only this is built into the window and does not allow for scrolling. Allows participants to enter text responses of multiple lines. Allows for drawing of any drawable items (unlike text dialog).



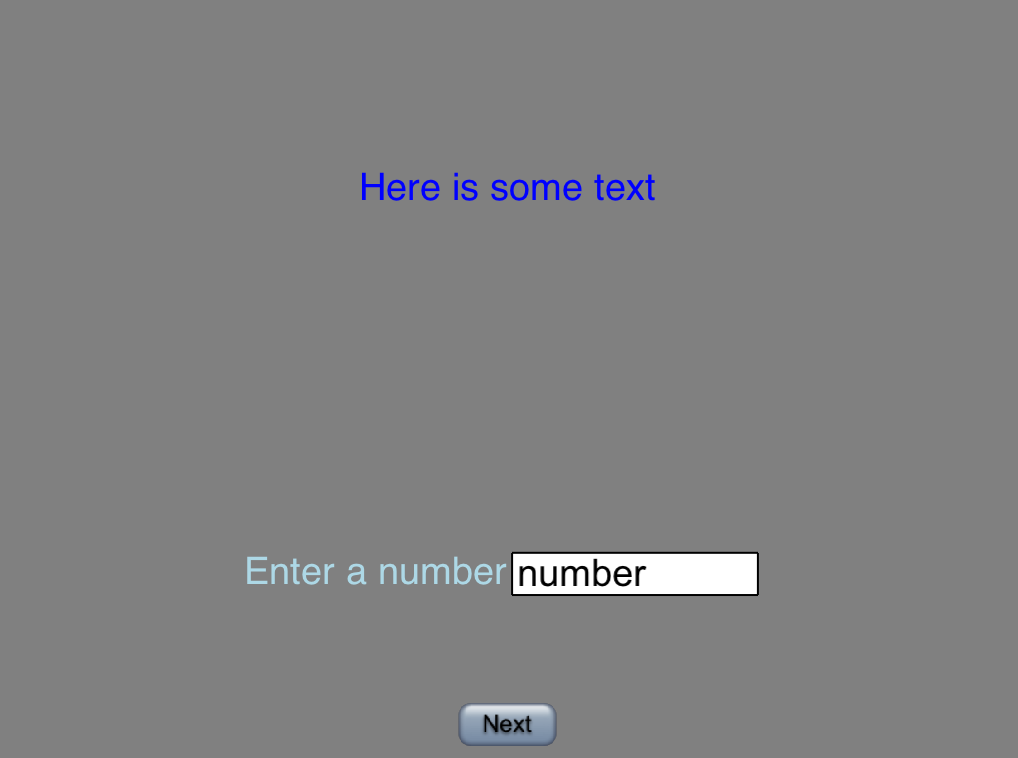
**To call a text input question:**

textInput(window, clock = None, drawList = [], prompt = ‘Enter your response’, promptHeight = .065, promptOffset = .1, promptColor = ‘white’, boxTop = 0.0)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *clock:* If provided, the psychopy clock object will be used for timing. If not, one will be created.
* *drawList:* A list of objects that should be drawn along with the trial
* *prompt:* String that appears directly above the entry window
* *promptHeight:* Height of prompt (in window normed units)
* *promptOffset:* distance prompt appears from entry window
* *promptColor:* Color of prompt text
* *boxTop:* Vertical position of text input box in ‘norm’ units.

**TEXT FIELD QUESTION:** This is a text input question that returns a one-line response of a limited size. This is good for short text response items. Can specify the maximum length of a response.



**How to call a text field question:**

textField(window, clock = None, label = ‘Enter a number’,labelColor = ‘black’, drawList = [], text = ‘number’, maxChars = 8, size = .1, pos = [0,0], type = ‘string’)

**Keyword arguments:**

* *window:* The parent window over which the trial is drawn (must be specified)
* *clock:* If provided, the psychopy clock object will be used for timing. If not, one will be created.
* *labelColor:* Color of the text item (see below)
* *drawList:* A list of objects that should be drawn along with the trial
* *text:* Text to appear initially in the entry window. Will be replaced by any text entered.
* *maxChars:* maximum # of characters for a response
* *size:* size of the entry text
* *pos:* position of the text entry box (in ‘norm’ units)
* *type:* Limit the entered data by limiting input to valid characters: ‘string’, ‘int’, or ‘float’