

ooRexxTry.rex
Lee Peedin – November 29th, 2007

ooRexxTry.rex is an ooRexx script that can be used very much like *rexstry.rex* that is distributed with Open Object Rexx. The primary difference is that code entered in ooRexxTry is executed as a *whole*, not line by line. You may also supply arguments to your code by specifying them in the arguments section of the dialog.

This version (1.0) of ooRexxTry requires the user have at least ooRexx Version 3.2 installed.

It should run on any version of Windows that supports **ooRexx 3.2** or higher.

NOTE: ooRexxTry is *not* meant to be an IDE nor does it have all the bells and whistles of an editor. It's simply a means to test code that you may not be ready to save as a file. ooRexxTry does *not* need to be run from a command line, **UNLESS** you have *pull* statements in your code.

All "say" statements within your code are redirected to a specific area of the dialog. The same is true for any values returned by your code. Errors, Information, & Returned values are also redirected to their specific area of the dialog.

The opening size of the dialog is dependent upon the user's screen resolution. Minimum supported screen resolution is 800 x 600. Information on the location and size of the dialog is saved in a .ini file and used all subsequent times the program is executed. The only exceptions are that the size/location are not saved in the .ini file if the program is closed while it is either minimized or maximized. If the application is closed while running on a secondary monitor and upon the next execution the secondary monitor is not present, the dialog will be positioned at the far right side of the primary screen and be on the same y axis.

Arguments **MUST** be entered on separate lines in the appropriate dialog entry field. Arguments with spaces **MUST BE QUOTED**. Expressions can be used as arguments.

The code dialog entry field can contain any code that can be valid ooRexx code and may include directives.

Use your keyboard's **Enter** key within the arguments area of the dialog and the code area of the dialog. Use your keyboard's **Tab** key to move between the dialog areas.

You may edit your arguments and/or code as much as you like and re-run the code.

You may run a **selected portion** of your code by simply selecting it and using the menu option or button for Run. The selected area will remain selected after execution.

Note: Size and position of the dialog is **never** saved if the dialog is Minimized or Maximized.

Menu Options

<u>F</u> ile	<u>E</u> dit	<u>T</u> ools	<u>H</u> elp
<u>R</u> un	Font <u>N</u> ame	<u>C</u> opy	Current <u>S</u> ettings
<u>S</u> aveAs	<u>L</u> ucida Console	<u>A</u> rgs	<u>A</u> bout
<u>O</u> pen	<u>C</u> ourier New	<u>C</u> ode	
<u>E</u> xit	Font <u>S</u> ize	<u>S</u> ays	
	<u>8</u>	<u>R</u> eturns	
	<u>10</u>	<u>E</u> rrors	
	<u>12</u>	<u>A</u> ll	
	<u>14</u>	<u>C</u> lear	
	<u>16</u>	<u>A</u> rgs	
	<u>18</u>	<u>C</u> ode	
		<u>S</u> ays	
		<u>R</u> eturns	
		<u>E</u> rrors	
		<u>A</u> ll	
		<u>S</u> ilent	
		<u>N</u> o	
		<u>Y</u> es	
		<u>S</u> ave Settings	
		<u>S</u> ave	

Focus will be returned to the code area of the dialog after menu selections. The cursor is placed at the end of the code unless the user is executing only a selected portion of the code.

Selection of the Run option will clear any previous data in these areas of the dialog: "Say", "Returns", and "Errors / Information".

Version Control

Started	21 Aug 2007
Version 0.4	23 Aug 2007
Version 0.5	24 Aug 2007
Version 0.5.1	24 Aug 2007
Version 0.5.2	24 Aug 2007
	Removed the opening dialog and added a font menu Added several choices to the font menu Each dialog area gets font menu choice applied Fixed problem with getScreenSize (wasn't available without the RxFuncAdd or having executed another dialog) Added a ClearAll button to clear all areas of the dialog
Version 0.6	25 Aug 2007
	Expressions can now be used for arguments - this means that arguments that are strings must NOW be in "quotes" All buttons have been replaced with menu selections Dialog width is set to approx. 65% of the user's screen width User may now execute only a portion of the code section by highlighting the desired code and selecting Alt F -> R (run) All local variables that deal with the dialog sizing are now done in a routine that is called at the start of the program. This done to clean up the dialog Init code Combined all the font name/size changes into the UnKnown method Combined all clipboard copies into the ClipBoard method
Version 0.6.1	25 Aug 2007
	Say statements are now placed in the dialog at the time they are executed NOT at the end of execution
Version 0.7	27 Aug 2007
	Dialog size is based on user's screen resolution AND the dialog is now resizable Re-Arranged all menu options and added several more options Added Help/About
Version 0.7.1	27 Aug 2007
	Added information headers to "Copy > All" output Added Run & Cancel buttons back Changed output of syntax errors Trapped syntax error in interpretation of arguments
Version 0.7.2	05 Sep 2007
	Made "About" documentation a button to execute pdf
Version 0.8	11 Sep 2007
	Replaced all instances of ~title to retrieve dialog content with getText method. This was done since ~title only returns the first 255 characters of an editControl. This meant that the entire size of the code could not exceed 255 characters. With this modification there is no limit on the size of the code or arguments.

- Version 0.8.1 11 Sep 2007 Found another instance of ~title used in the Code2File method
- Version 0.8.2 13 Sep 2007 Dialog retains size, position, fontname, & fontsize for next run by using values stored in oorextry.ini
- Version 0.8.3 13 Sep 2007 Code execution and code completion indicator now displays in the Errors/Information dialog.
Beeps upon code completion
Add option to menu for "silent" which turns off all beeps
The value of .silent is stored in the ini file on exit and used at the next execution.
Added "Current Setting" to Help menu - shows current setting for fontname, fontsize, silent.
- Version 0.8.4 14 Sep 2007 Mouse pointer is placed in the center of the code_input area and the Cursor_Wait is displayed while code is being executed.
Cursor is restored to its original shape after execution is completed and mouse pointer is returned to its prior execution position.
Added testing for a single runtime argument. If the argument is "default", all data stored in the oorextry.ini is ignored and default values are used for font name, font size, silent, dialog position, and dialog size. Values at exit "are" stored in the ini file. This will allow a user to "recover" the dialog if it was left on a secondary monitor upon exit and the secondary monitor is not available the next time the application is needed.
- Version 0.8.5 14 Sep 2007 Use the WindowObject class to determine if the dialog is Minimized or Maximized – if either, do not save the size/position in oorextry.ini.
- Version 0.8.6 14 Sep 2007 Fixed bug with mouse pointer and placement if an error was trapped in the arguments.
Added menu option to Tools to Save current settings.
- Version 0.8.7 19 Sep 2007 Added handle to filenamedialog so that the dialog is linked (position wise) to our main dialog.

