

QGIS Application - Feature request #6397

save and restore the command history in the Python console

2012-09-23 09:54 PM - Alister Hood

| | | |
|--|--------|--------------------------------------|
| Status: | Closed | |
| Priority: | Normal | |
| Assignee: | | |
| Category: | | |
| Pull Request or Patch supplied: | No | Resolution: fixed/implemented |
| Easy fix?: | No | Copied to github as #: 15651 |
| Description It would be useful to be able to save and restore the command history in the Python console. But what would be best?: <ol style="list-style-type: none">1. saving a separate command history per project2. saving a separate command history per user3. manually saving and restoring, so the user chooses the file4. an option so you can choose 1, 2 or 3. I guess if 3 was built in then users could implement 1 or 2 with a project action or similar. | | |

History

#1 - 2012-09-24 03:52 PM - Salvatore Larosa

Hi Alister,

Are you using the new python console?

If so, you can save/open history file by "save script file/open script file" button!

Also, I am working on a new feature in order to save history silently!

#2 - 2012-09-24 08:43 PM - Alister Hood

Hi,

Salvatore Larosa wrote:

| *If so, you can save/open history file by "save script file/open script file" button!*

That is quite different from what I'm talking about. That saves the history as a script, and when you load it it runs all the commands in the script - that would be similar to saving the history in the console on linux as a bash script, and running that script.

I'm talking about just remembering the commands in the history, like the history is normally remembered in a bash shell on linux, so when you restart QGIS (or reopen the project), you can scroll back through them.

I guess, we can already do it with just this:

```
history_filename = "C:/test.py"
lines = [l.rstrip() for l in open(history_filename, "U")]
for i in lines:
```

```
qgis.console._console.edit.updateHistory(i)
```

Salvatore Larosa wrote:

Also, I am working on a new feature in order to save history silently!

I guess that is probably what I'm talking about, anyway. Will it just be for option 2 from my list?

#3 - 2012-09-24 09:36 PM - Alister Hood

Alister wrote:

But what would be best?:

1. saving a separate command history per project
2. saving a separate command history per user
3. manually saving and restoring, so the user chooses the file
4. an option so you can choose 1, 2 or 3.

I guess if 3 was built in then users could implement 1 or 2 with a project action or similar.

So this is half an implementation (loading the history) of 1, using a project action:

```
def openProject():
    import qgis
    history_filename = "C:/test.py"
    lines = [l.rstrip() for l in open(history_filename, "U")]
    for i in lines:
        qgis.console._console.edit.updateHistory(i)
    pass
```

The big problem is that it only works if the QGIS Python console is open before the project is opened and QGIS doesn't even remember whether the Python console was open last time QGIS was closed...

#4 - 2012-09-24 09:52 PM - Alister Hood

Alister wrote:

The big problem is that it only works if the QGIS Python console is open before the project is opened and QGIS doesn't even remember whether the Python console was open last time QGIS was closed...

I guess there might be code that could be used to open the console...

#5 - 2012-09-25 01:54 AM - Alister Hood

Oh, this works. The only issue is that it always toggles the visibility of the console. This is not desirable - either it should return the visibility to its initial state, or it should always make it visible.

```
def openProject():
    import qgis.console
    global _console
    try:
        qgis.console.show_console()
    except:
        _console = qgis.PythonConsole(qgis.utils.iface.mainWindow())
    history_filename = "/test_macro.py"
    lines = [l.rstrip() for l in open(history_filename, "U")]
    for i in lines:
        qgis.console._console.edit.updateHistory(i)
    pass
```

#6 - 2012-09-25 03:41 AM - Alister Hood

Oops. That should just be this:

```
def openProject():
    import qgis.console
    qgis.console.show_console()
    history_filename = "/test_macro.py"
    lines = [l.rstrip() for l in open(history_filename, "U")]
    for i in lines:
        qgis.console._console.edit.updateHistory(i)
    pass
```

#7 - 2012-09-25 04:42 AM - Alister Hood

Ah, of course. This is what I was trying to achieve:

```
def openProject():
    import qgis.console
    qgis.console.show_console()
    qgis.console.show_console()
    history_filename = "/test_macro.py"
    lines = [l.rstrip() for l in open(history_filename, "U")]
    for i in lines:
        qgis.console._console.edit.updateHistory(i)
    pass
```

#8 - 2012-09-25 04:52 AM - Alister Hood

edit: replaced code in last comment

#9 - 2015-11-09 06:13 AM - Médéric RIBREUX

- *Resolution set to fixed/implemented*
- *Status changed from Open to Closed*
- *% Done changed from 0 to 100*

Hello, bug triage...

this feature is now implemented into QGIS. There is a console history. You can use it with Up and Down keys to navigate into (bash style). Console history is stored in a file into ~/.qgis2/console_history.txt (so the history is stored on a user level). You can even inspect the whole console history from the console using Ctrl+Shift+Space.