How to close correctly a Qt program?

When I try to close my Qt program, it just keeps running in the background though there's no window anymore.

Basically, I would like to know what I should do so it closes properly when I click the red cross on my main window (which has no parent).

Following this link, I tried a few things like :

```
QApplication app(argc, argv);
//...
app.connect(&app, SIGNAL(lastWindowClosed()), &app, SLOT(quit()));
return app.exec();
```

or

```
QApplication app(argc, argv);
//...
app.setQuitOnLastWindowClosed(true);
return app.exec();
```

but neither work, the process still stays in memory, after the cross is clicked.

Then, how can I close correctly my program ?

c++   qt

edited Jan 15 '14 at 15:54
Kara
3,182   7   31   48

asked Jun 12 '13 at 9:02
JBL
7,855   3   25   57

1 Answer

You can close your application manually using `QApplication::quit()` .

By default the execution is terminated when the last top level window with the `Qt::WA_QuitOnClose` attribute has been closed. You don't need to connect `lastWindowClosed` to `quit` because it repeats the default `setQuitOnLastWindowClosed` behavior. You don't need to do `setQuitOnLastWindowClosed(true)` either because it's true by default. The code you've posted does nothing (if nothing is changed somewhere else).

Possible solutions:

- Check your main window attributes. Maybe you have removed `Qt::WA_QuitOnClose` attribute.
- If you have reimplemented `closeEvent` in your top level window, check that close event is being accepted.
- Check if there are some other top level widgets that may be hidden but not closed. Use `QApplication::topLevelWidgets()` to list them.

edited Jun 12 '13 at 21:24

answered Jun 12 '13 at 9:09
Pavel Strakhov
25.7k   3   31   79

Thanks, I'll try this. I did nothing with the `Qt::WA_QuitOnClose` attribute, though, but i'll try to reimplement `closeEvent()` , I didn't read anything about it while searching. – JBL Jun 12 '13 at 9:14

No, if you don't need to reimplement closeEvent, you should not do that. It must work fine by default. I meant that if you reimplemented closeEvent, your implementation may be incorrect. – Pavel Strakhov Jun 12 '13 at 9:18

Oh, indeed. Well, I haven't reimplemented it at all... So, what should I do then ? I must absolutely rely on `QApplication::quit()` ? – JBL Jun 12 '13 at 9:20

Check `QApplication::topLevelWidgets()` . – Pavel Strakhov Jun 12 '13 at 9:23

Just saw there is actually 8 widgets in the list it returns. So that means that when my window exits, there's still 7 widget that are still present "at top level" and keep the program running (if I understand correctly) ? I really don't know which these are. – JBL Jun 12 '13 at 9:29