How to create and destroy Qt stackable windows

This article shows how to dynamically create and destroy stackable windows in Qt.

Note that this example will work on every platform Qt is ported to but has only been tested on Maemo5.
We start with main.cpp:

```
#include <QApplication>

#include "child.h"

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);

    Child child;
    child.show();

    return a.exec();
}
```

Next is child.h:

```
#ifndef CHILD_H
#define CHILD_H

#include <QMainWindow>
#include <QPushButton>
#include <QEvent>

class Child : public QMainWindow
{
    Q_OBJECT

public:
    Child(QWidget *parent=0);

protected:
    bool eventFilter(QObject *obj, QEvent *event);

private slots:
    void buttonClicked(bool checked = false);

private:
    QPushButton *button;
    QMainWindow *window;
};

#endif //CHILD_H
```

This derivative of QMainWindow, which contains one button, slot for button's clicked signal and event filter. When button is clicked we create a window and set event filter on it to catch CloseEvent of the window. When some event is delivered to window, eventFilter will be activated first. In case of CloseEvent we destroy window and execute some code matching to the occasion (we output debug message).

(C) Copyright Nokia 2014. All rights reserved.

Implementation is in child.cpp:

```
#include <QDebug>

#include "child.h"

Child::Child(QWidget *parent)
    : QMainWindow(parent), window(0)
{
    button = new QPushButton();
    button->setText("This is a button");
    button->setFont(QFont("Nokia Sans", 35, QFont::Bold));
    setWindowTitle("This is a main window");
    setCentralWidget(button);
    #ifdef Q_WS_MAEMO_5
        setAttribute(Qt::WA_Maemo5StackedWindow);
    #endif
    connect(button, SIGNAL(clicked(bool)), this, SLOT(buttonClicked(bool)));
}

bool Child::eventFilter(QObject *obj, QEvent *event)
{
    if (event->type() == QEvent::Close) {
        delete window;
        window = 0;
        qDebug("No window");
        return true;
    } else
        // standard event processing
        return QObject::eventFilter(obj, event);
}


void Child::buttonClicked(bool checked)
{
    Q_UNUSED(checked);

    window = new QMainWindow(this);
    window->setWindowTitle("This is second window");
    window->installEventFilter(this);
    #ifdef Q_WS_MAEMO_5
        window->setAttribute(Qt::WA_Maemo5StackedWindow);
    #endif
    window->show();

    qDebug("Window");
}
```


(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.