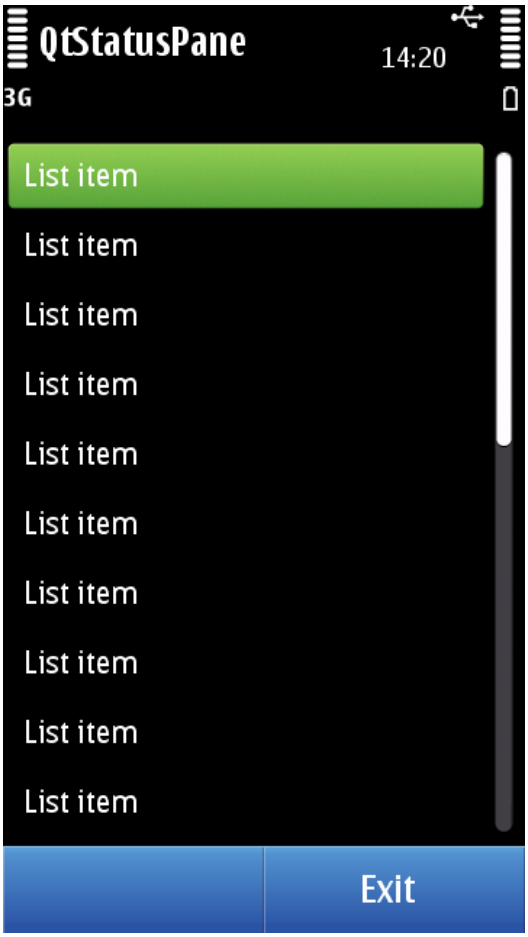
Archived:Hiding status and control panes in Qt for Symbian

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

Qt Quick should be used for all UI development on mobile devices. The approach described in this article (based on QWidget) is deprecated.

This code snippets shows how to hide Symbian status and control (CBA buttons) panes.

Control and status pane visible



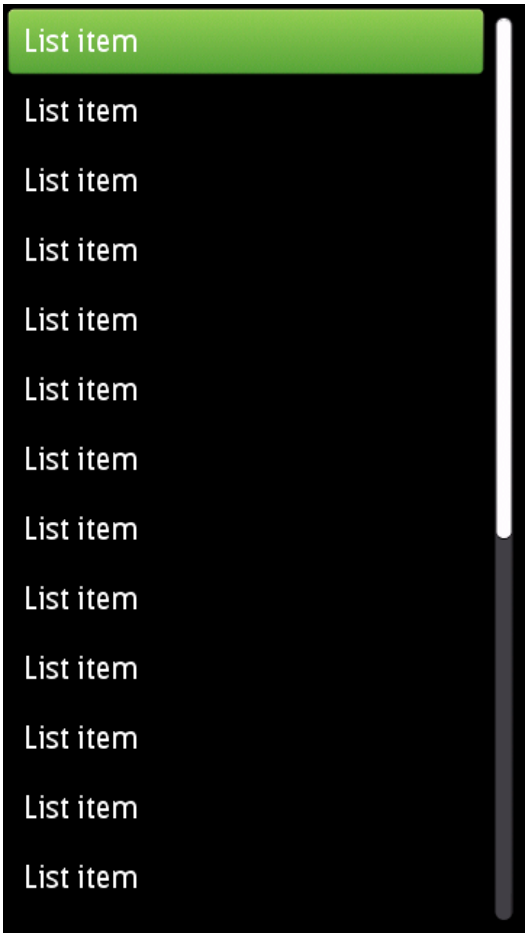
Execute QMainWindow as show maximized mode, call `QMainWindow::showMaximized()`

```
int main(int argc, char *argv[]) {
    QApplication app(argc, argv);
    MainWindow w;
    w.showMaximized();
    return app.exec();
}
```

Hide control and status pane

No control or status pane visible.

(C) Copyright Nokia 2013. All rights reserved.



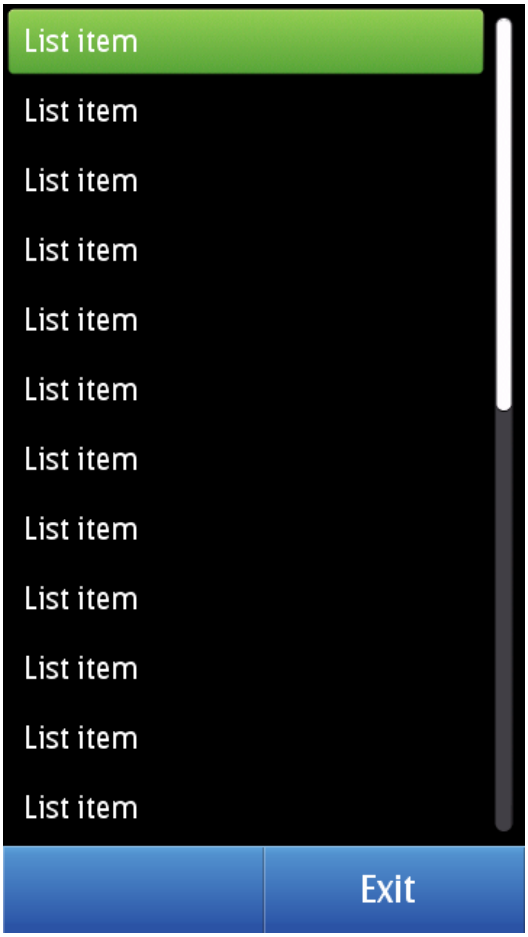
Execute QMainWindow as fullscreen mode, call `QMainWindow::showFullScreen()`

```
int main(int argc, char *argv[]) {
    QApplication app(argc, argv);
    MainWindow w;
    w.showFullScreen();
    return app.exec();
}
```

Hide status pane

Only control pane is visible.

(C) Copyright Nokia 2013. All rights reserved.



Execute QMainWindow as fullscreen mode, call `QMainWindow::showFullScreen()`

```
int main(int argc, char *argv[]) {
    QApplication app(argc, argv);
    MainWindow w;
    w.showFullScreen();
    return app.exec();
}
```

Define control pane as visible in fullscreen mode, set `Qt::WindowSoftkeysVisibleHint` window flag into `QMainWindow`

```
// Enable Qt::WindowSoftkeysVisibleHint flag in QMainWindow constructor
QWidget::setWindowFlags(windowFlags() | Qt::WindowSoftkeysVisibleHint);
```

See also

- WindowType enum
- QWidget
- Archived:Hiding status and control panes using Symbian C++

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.