Archived:Dynamic use Qt StyleSheets

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 (using C++ for the Qt app UI) is deprecated.

Overview

This article shows how to use StyleSheets Dynamically. The easiest way to customize the look of Qt applications with CSS without having to do by Painter function which might be relatively hard to do. Style sheets are very light weight and faster way to improve the look and feel of Qt controls. For the real unique user experience, QtQuick and QGraphicsView is a better choice.

When using stylesheets, styles are applied using selectors. For instance, the snippet below gives all QLineEdit widgets a yellow background color. Here, the selector is the word QLineEdit.

```
QLineEdit
{
    background: red;
}
```

There are numerous selectors based on object names, sub-controls, pseudo-states and more. These can be used to change the look of a user interface and also make all QPushButton widgets named **okButton** have green text, and so on.

For dynamic changes in the user interface look, the property value selector can be used in combination with dynamic properties. Dynamic properties was introduced in Qt 4.2 and allows your to assign property values to QObjects for properties that do not exist at compile time. I.e., if you choose to set the a property called urgent to true for a QObject, that property will stick even tough the class does not contain a `Q_PROPERTY` macro for the property urgent.

Creating a stylesheet selector relying on a dynamic property, e.g. urgent, makes it possible to highlight parts of the user interface in a very dynamic manner. For instance, adding the following rule to the stylesheet above will given any QLineEdit with the urgent property set to true red text on a yellow background.

```
[urgent=true]
{
    color: red;
}
```

This does not have to be limited to QLineEdits though. Setting the urgent property to true on a QCheckBox or QPushButton will make their texts red as well. Basically, any stylesheet aware widget can be turned red using this simple trick.

--skumar_rao 15:11, 21 November 2010 (UTC)

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.