uqtie

Release 0.1.1

Feb 25, 2020

Contents:

1 Intro to UQtie 3

2 Main Application Window 5

3 Stylesheet Variables 7

4 Indices and tables 9

UQtie is a small set of features to make it easier to use PyQt5. It implements a pattern for building throwaway or small Qt applications. Features include Font Selection, Zoom In/Out, and geometry persistence.

CHAPTER 1

Intro to UQtie

The basic point of **UQtie** is to make it easy to create a simple GUI app that supports some useful features that I typically want. So far, these are:

- Window geometry persistence - you resize and reposition the window, then quit the app. Next time you start the app, it starts with the size and position you chose.
- Font selection through a Font Dialog
- Zoom in and Zoom out - you use CTRL-+ and CTRL– to make text, widgets and borders larger or smaller
- Qt Stylesheet (QSS) support that doesn't preclude runtime changes to look-and-feel attributes (e.g. font)

These are obviously not all the features you might want your app to have. I intend to add more features as time goes on.

Most of the features are obtained simply by making your main window a subclass of `UqtWin.MainWindow` instead of `PyQt5.QtWidgets.QMainWindow`. See *Main Application Window*.

These are the modules in the **UQtie** package:

Module Name	Purpose
UqtMav	UqtMavConn allows easy use of pymavlink connections in Qt app
UqtWin	MainWindow class provides an app main window with features
UqtStylesheet	Maintains a QSS file with variable properties - a "poor man's SASS"

UqtMav is not particularly generic, but I have written a handful of little MAVLink tools, so MAVLink is generic for *me*

CHAPTER 2

Main Application Window

Most of the features are obtained simply by making your main window a subclass of `UqtWin.MainWindow` instead of `PyQt5.QtWidgets.QMainWindow`.

For example, note these three steps in the source of `simple_uqtie.py` in the `example/` subdirectory of the repo":

1. Import the `UqtWin` module:

```
from   uqtie import UqtWin
```

2. Use `UqtWin.MainWindow` as the base class of the main window:

```
class TestAppMainWindow(UqtWin.MainWindow):
    def __init__(self, parsedArgs, **kwargs ):
        super(TestAppMainWindow, self).__init__(parsedArgs, **kwargs)
```

3. Instantiate the subclass:

```
# can add optional title='<OptionalTitle>' if it is different from your app name
mainw = TestAppMainWindow(parsedArgs, app=app, organizationName='Craton', appName=
↪'UqtTest')
```

CHAPTER 3

Stylesheet Variables

There are advantages and disadvantages to Qt stylesheets. Qt stylesheets partially overlap other Qt features, such as Qt Style. They aren't mutually exclusive, and they don't all play together either.

UqtStylesheet attempts to make it possible to use a stylesheet while still using the QFontDialog to select fonts, and zoom-in/zoom-out shortcuts such as CTRL-+.

The idea is that we have variables (which come from somewhere, right now a pickled dictionary, but later could be elsewhere, or multiple elsewheres). These variables can be used in the stylesheet, such that the dynamic changes in appearance can be made at runtime without requiring stylesheet changes, and the changes can be persisted also without changing the stylesheet.

Widget sizes (e.g `QScrollBar:vertical { width }`), font, and font size seem like good candidates to be determined dynamically instead of via hardcoded values in a stylesheet. That way, when you have high-DPI monitor and less-than-perfect vision, you can adjust on-the-fly, and your adjustments are saved automatically.

Part of my motivation for doing this is because PyQt running on Windows, Linux, MacOS and Cygwin doesn't behave identically, not even close to identical in some ways. If they would all have identical QScreens given identical monitors and graphics cards, then you could just make reasonable choices for your stylesheet, and rely on the OS GUI settings.

Variables you can use in a QSS file are:

```
$main_font_family
$main_font_weight (not supported yet)
$main_font_size
$scroll_bar_width
```

CHAPTER 4

Indices and tables

- genindex
- modindex
- search