blnuhr Documentation

Release 2.gc33c19b.dirty

Pete R. Jemian

Dec 12, 2018

Contents

blnuhr: Python & Qt rendition of Berlin's quantity didactics clock

docs http://blnuhr.readthedocs.org

git https://github.com/prjemian/blnuhr/

http://www.surveyor.in-berlin.de/berlin/uhr/indexe.html The Berlin quantity didactics clock

CHAPTER 1

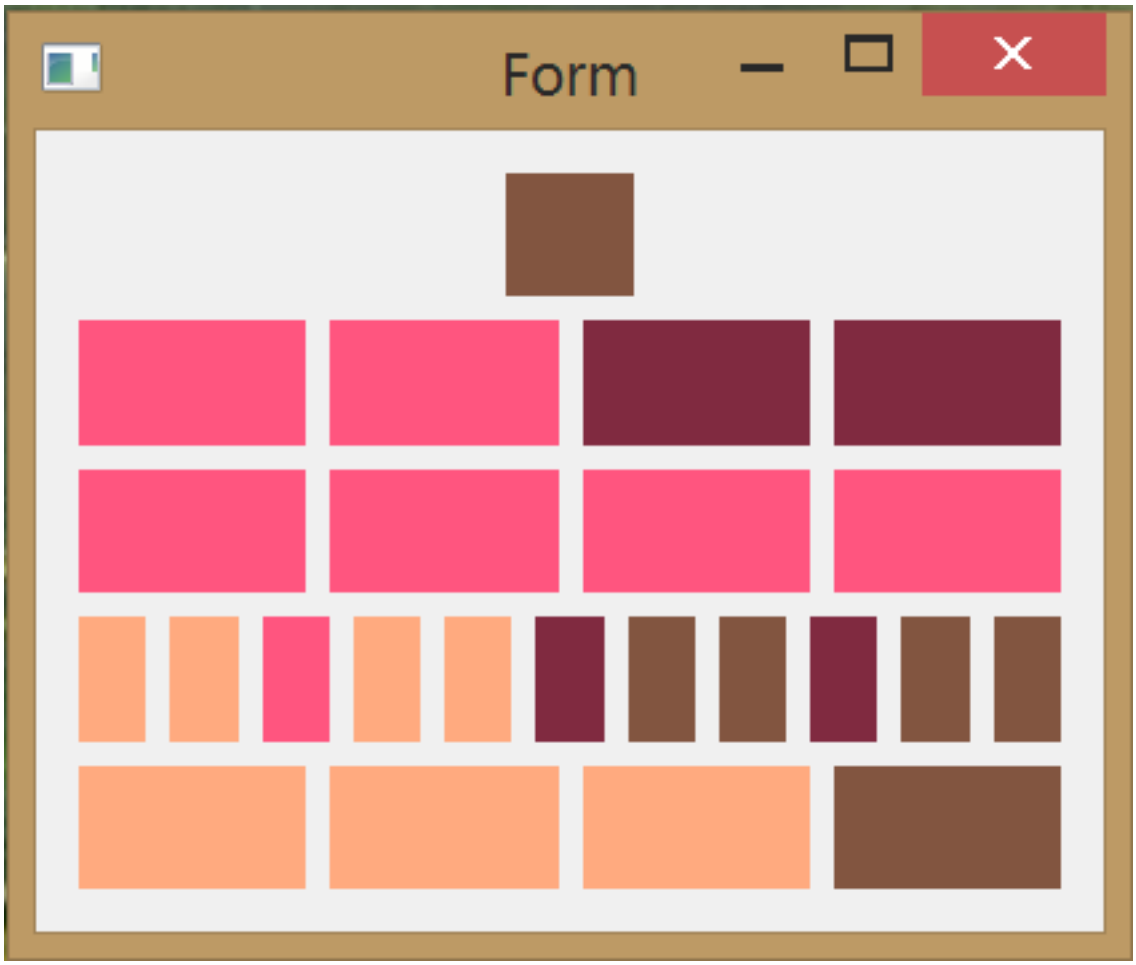
How it shows the time

The time is calculated by adding the lit rectangles. The top rectangle blinks changes every second. In the next row, the each rectangle represents 5 hours. In the third row, every rectangle represents 1 hour. Together, these two rows show the hour of the day. The fourth row rectangles represents 5 minute intervals. (Red rectangles show 15 minute intervals.) In the last row, every rectangle represents 1 minute. Like the hours, these two rows show the minutes after the hour.

CHAPTER 2

Example 14:28 (2:28 pm)

In this example, the time is 14:28.



Here's the explanation:

row 1 seconds are an even number, LED is off

row 2 first two LEDs are on, at least 10 AM

row 3 all LEDs are on, hours = 10 AM + 4 = 14:00

row 4 first 5 LEDs are on, at least 25 after the hour

row 5 first 3 LEDs are on, minutes = 25 + 3 = 14:28

CHAPTER 3

Compare 16:57 (4:57 pm)

Compare with a view of the Berlin Quantity Didactics Clock in 2004 (after it was moved to the Europa center). The time on the clock shown is 16:57 (4:57 pm).


TICKETS

CHAPTER 4

blnuhr Package: Source Code Documentation

Source code documentation for *blnuhr*.

4.1 `main` Module

class blnuhr.main.**Clock_blnuhr**(***_kwargs*)
Bases: PyQt4.QtGui.QWidget

create a widget for the clock and start it running

start()
begin the periodic update of the clock

update(*t=None*)
manage a periodic update of the clock

Show the time as a string on the seconds LED as a tool tip

blnuhr.main.**main**()
entry point to run standalone

4.2 `resources` Module

(internal) support for items in resources folder, such as forms defined in .ui files

blnuhr.resources.**get_forms_path**()
identify our resources directory

blnuhr.resources.**loadUi**(*ui_file*, *baseinstance=None*, ***kw*)
load a .ui file for use in building a GUI

Wraps *uic.loadUi()* with code that finds our program's *resources* directory.

See http://nullege.com/codes/search/PyQt4.uic.loadUi

See http://bitesofcode.blogspot.ca/2011/10/comparison-of-loading-techniques.html

inspired by: http://stackoverflow.com/questions/14892713/how-do-you-load-ui-files-onto-python-classes-with-pyside?lq=1

Basic Procedure

1. Use Qt Designer to create a .ui file.
2. Create a python class of the same type as the widget you created in the .ui file.
3. When initializing the python class, use uic to dynamically load the .ui file onto the class.

Here is an example from this code:

```
from PyQt4 import QtGui
import resources

UI_FILE = 'plainTextEdit.ui'

class TextWindow(QtGui.QDialog, form_class):

    def __init__(self, title, text):
        QtGui.QDialog.__init__(self, parent)
        resources.loadUi(UI_FILE, baseinstance=self)
        self.setWindowTitle(title)
        self.plainTextEdit.setPlainText(text)

import sys
app = QtGui.QApplication(sys.argv)
win = TextWindow('the title', __doc__)
win.show()
sys.exit(app.exec_())
```

blnuhr.resources.**resource_file**(*filename*)
absolute path to file in resources directory

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

version 0.1.12

published Dec 12, 2018

Python Module Index

b

Index