EBook Gratis

APRENDIZAJE pyqt

Free unaffiliated eBook created from
Stack Overflow contributors.

#pyqt

Tabla de contenido

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: pyqt

It is an unofficial and free pyqt ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official pyqt.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con pyqt

Observaciones

PyQt es un enlace de Python al popular framework de aplicaciones Qt multiplataforma comúnmente utilizado para hacer aplicaciones gráficas. PyQt4 admite Qt4 y PyQt5 admite Qt5. Se ejecuta en todas las plataformas compatibles con Qt (Windows, OS X, Linux, iOS y Android). Los enlaces se implementan como un conjunto de módulos y clases de Python.

Para obtener más información, consulte el sitio web de PyQt .

Examples

Instalación de PyQt4

Método de instalación sugerido

Windows : descargue y ejecute el archivo de configuración binario .

Linux (Debian) : Ejecute este comando en su línea de comandos:

```
$ apt-get install python-qt4 pyqt4-dev-tools qt4-designer
```

OS X : Ejecuta este comando en tu línea de comando:

```
$ brew install pyqt
```

Instale manualmente

También puede descargar el código fuente manualmente desde aquí y luego instalarlo y configurarlo usted mismo.

Prueba tu instalacion

Si pyqt está instalado correctamente, podrá ejecutar el comando `pyuic4` . Si está instalado correctamente, verá el siguiente error:

```
$ pyuic4
Error: one input ui-file must be specified
```

Instalación completa

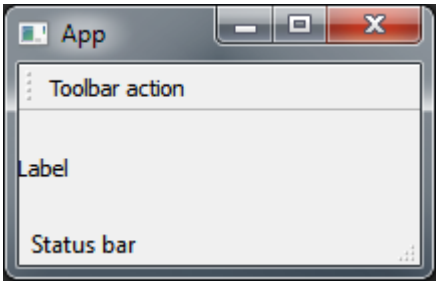
Ahora ha instalado la biblioteca PyQt4. Dos aplicaciones útiles también se han instalado a lo largo del lado PyQt4:

- Qt Designer: una aplicación para el diseño 'arrastrar y soltar' de interfaces gráficas (crea

archivos `.ui`),
- pyuic4: una aplicación de línea de comandos que puede convertir archivos `.ui` en código Python.

Una aplicacion basica

El siguiente ejemplo muestra una ventana de GUI principal básica con un widget de etiqueta, una barra de herramientas y una barra de estado utilizando PyQt4.




```
import sys
from PyQt4 import QtGui


class App(QtGui.QApplication):
    def __init__(self, sys_argv):
        super(App, self).__init__(sys_argv)
        self.build_ui()

    def build_ui(self):
        # build a main GUI window
        self.main_window = QtGui.QMainWindow()
        self.main_window.setWindowTitle('App')
        self.main_window.show()

        # add a label to the main window
        label = QtGui.QLabel('Label')
        self.main_window.setCentralWidget(label)

        # add a toolbar with an action button to the main window
        action = QtGui.QAction('Toolbar action', self)
        toolbar = QtGui.QToolBar()
        toolbar.addAction(action)
        self.main_window.addToolBar(toolbar)

        # add a status bar to the main window
        status_bar = QtGui.QStatusBar()
        status_bar.showMessage('Status bar')
        self.main_window.setStatusBar(status_bar)


if __name__ == '__main__':
    app = App(sys.argv)
    sys.exit(app.exec_())
```

Hola Mundo

Este código básico abrirá una ventana GUI "Hello world" usando PyQt4:

```
import sys
from PyQt4 import QtGui

# create instance of QApplication
app = QtGui.QApplication(sys.argv)

# create QLabel, without parent it will be shown as window
label = QtGui.QLabel('Hello world!')
label.show()

# start the execution loop of the application
sys.exit(app.exec_())
```

Este es el mismo código que usa PyQt5.

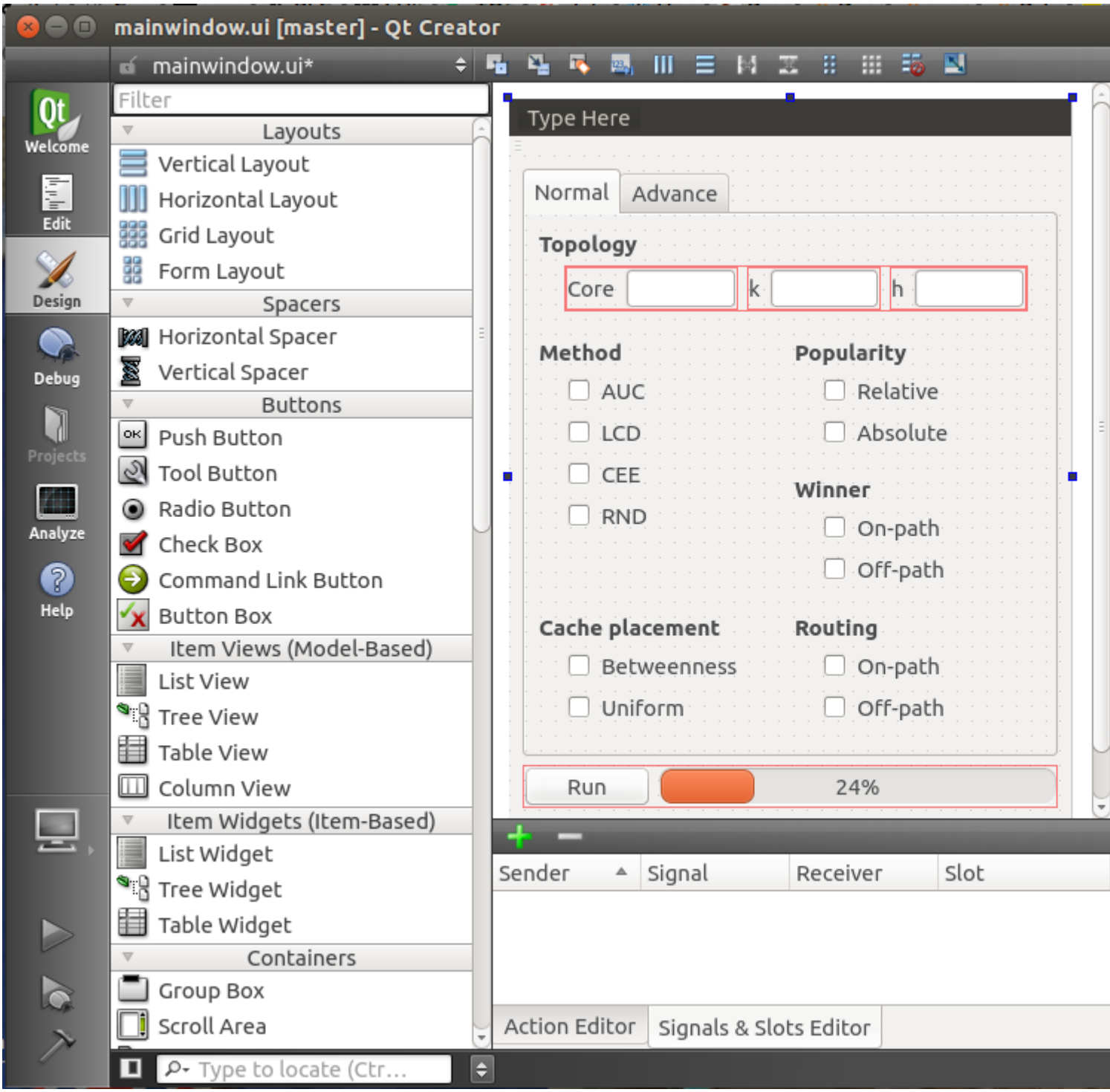
```
import sys
from PyQt5 import QtWidgets

# create instance of QApplication
app = QtWidgets.QApplication(sys.argv)

# create QLabel, without parent it will be shown as window
label = QtWidgets.QLabel('Hello world!')
label.show()

# start the execution loop of the application
sys.exit(app.exec_())
```

Una muestra simple de arrastrar y soltar

Haga una aplicación GUI simple en 3 sencillos pasos.

1. Diseño

Abra `Qt Creator` , cree un nuevo proyecto y haga su diseño. Guarde su resultado como archivo `.ui` (aquí: `mainwindow.ui`).



2. Genera el archivo .py correspondiente

Ahora puede crear un archivo .py a partir de su archivo .ui que generó en el paso anterior. Ingrese lo siguiente en su línea de comando:

```
$ pyuic4 mainwindow.ui -o GUI.py
```

Si la línea anterior se ejecuta con éxito, se `GUI.py` un archivo `GUI.py`

3. Códigos de Python

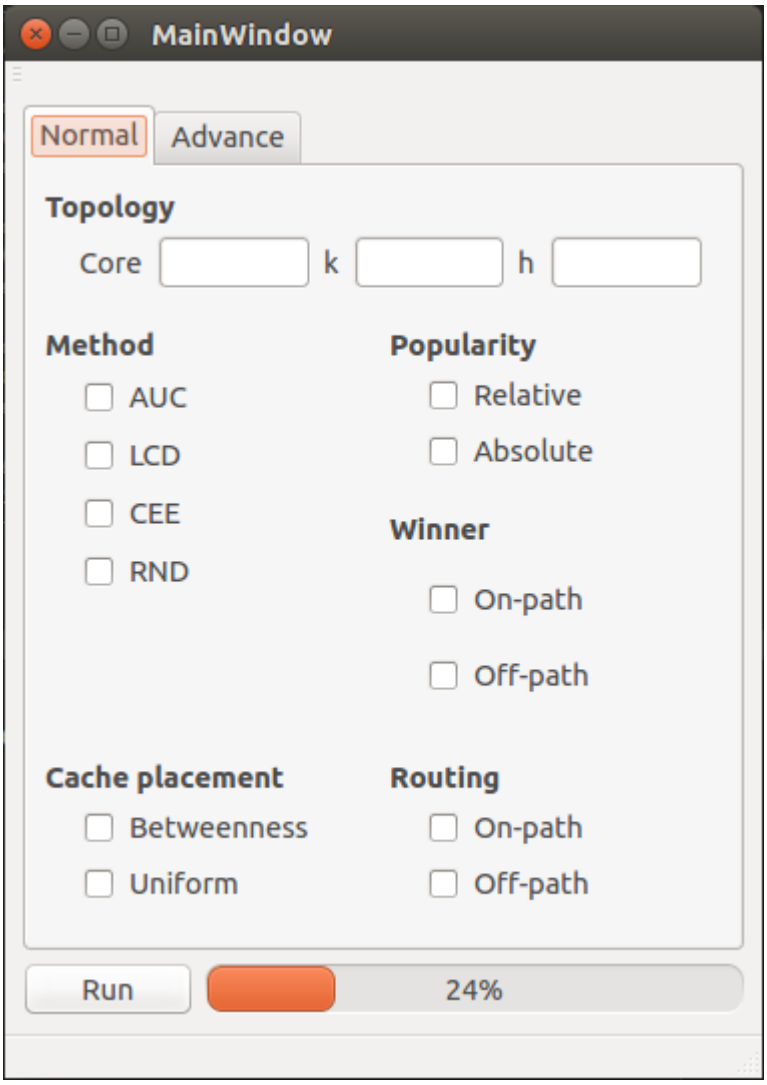
Puede agregar su propio código (por ejemplo, señales y ranuras) en el archivo `GUI.py` , pero es

mejor agregarlos en un archivo nuevo. Si alguna vez desea realizar cambios en su GUI, el archivo `GUI.py` se sobrescribirá. Es por eso que usar otro archivo para agregar funcionalidad es mejor en la mayoría de los casos.

Llamemos al nuevo archivo `main.py`

```
from PyQt4 import QtGui
import sys
import GUI # Your generated .py file


class MyApp(QtGui.QMainWindow, GUI.Ui_MainWindow):
    def __init__(self, parent=None):
        super(ExampleApp, self).__init__(parent)
        self.setupUi(self)

        # Connect a button to a function
        self.btn_run.clicked.connect(self.run)

    def run(self):
        # Write here what happens after the button press
        print("run")


if __name__ == '__main__':
    app = QtGui.QApplication(sys.argv)
    form = ExampleApp()
    form.show()
    app.exec_()
```

Ahora puedes ejecutar `main.py` y ver tu GUI.



Lea Empezando con pyqt en línea: https://riptutorial.com/es/pyqt/topic/1715/empezando-con-pyqt

Capítulo 2: Usando hilos con PyQt

Observaciones

Si bien algunas partes del marco Qt son seguras para subprocesos, muchas de ellas no lo son. La documentación de Qt C ++ proporciona una buena visión general de qué clases son reentrantes (se pueden usar para crear instancias de objetos en varios subprocesos). Las siguientes reglas son las más buscadas:

- No puede crear o acceder a un objeto de la GUI de Qt desde fuera del hilo principal (por ejemplo, cualquier subclase `QWidget` o similar).
- Incluso si la clase Qt es reentrada, no puede compartir el acceso a un objeto Qt entre subprocesos a menos que la documentación de Qt para esa clase indique explícitamente que las instancias son seguras para subprocesos.
- Puede usar `QObject.moveToThread()` si necesita mover un objeto Qt de un hilo a otro (no se aplica a los objetos de la GUI de Qt, que siempre deben permanecer en el hilo principal). Pero tenga en cuenta que el objeto no debe tener un padre.

Según este control de calidad de desbordamiento de pila, no se recomienda utilizar los hilos de Python si su hilo tiene la intención de interactuar con PyQt de alguna manera (incluso si esa parte del marco Qt es seguro para los hilos).

Examples

El modelo trabajador

```
# this method can be anything and anywhere as long as it is accessible for connection
@pyqtSlot()
def run_on_complete():

    pass

# An object containing methods you want to run in a thread
class Worker(QObject):
    complete = pyqtSignal()

    @pyqtSlot()
    def a_method_to_run_in_the_thread(self):
        # your code

        # Emit the complete signal
        self.complete.emit()

# instantiate a QThread
thread = QThread()
# Instantiate the worker object
worker = Worker()
# Relocate the Worker object to the thread
worker.moveToThread(thread)
# Connect the 'started' signal of the QThread to the method you wish to run
```

```
thread.started.connect(worker.a_method_to_run_in_the_thread)
# connect to the 'complete' signal which the code in the Worker object emits at the end of the
method you are running
worker.complete.connect(run_on_complete)
# start the thread (Which will emit the 'started' signal you have previously connected to)
thread.start()
```

Lea Usando hilos con PyQt en línea: https://riptutorial.com/es/pyqt/topic/2775/usando-hilos-con-pyqt

Creditos

S. No	Capítulos	Contributors
1	Empezando con pyqt	101, Achayan, Community, Ian, Makoto, mehD, three_pineapples, Trilarion
2	Usando hilos con PyQt	ederag, ekhumoro, three_pineapples