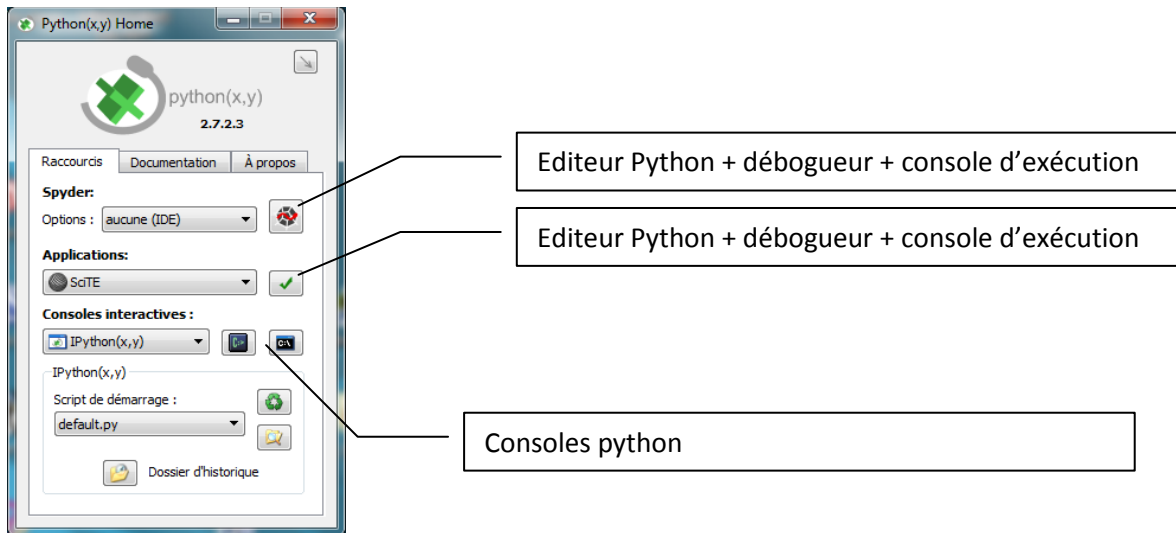


Tutoriel Qt Designer + Python

Etape 1 : Python(x,y)

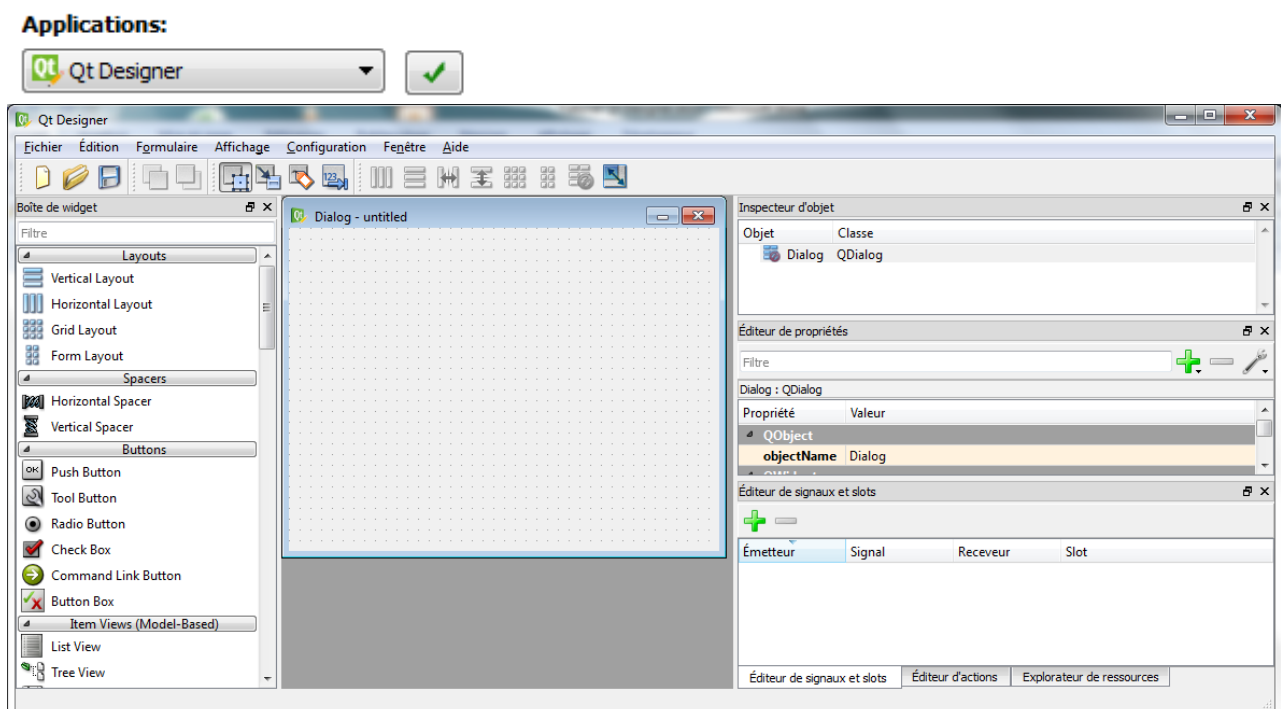
Plusieurs solutions s'offrent à nous pour développer une application fenêtrée en python. J'ai choisi la suite Python(x,y) car elle dispose de tous les outils nécessaires regroupés dans une seule application. L'installation des outils se réduit donc à l'installation du logiciel Python(x,y).



Etape 2 : Interface utilisateur (UI)

Qt Designer permet de créer graphiquement l'interface utilisateur ainsi que la gestion des événements (signaux et slots).

Dans la fenêtre Home de Python(x,y), choisir Qt Designer dans la liste des applications disponibles et cliquez sur le bouton de validation.



Utilisation de Qt Designer :

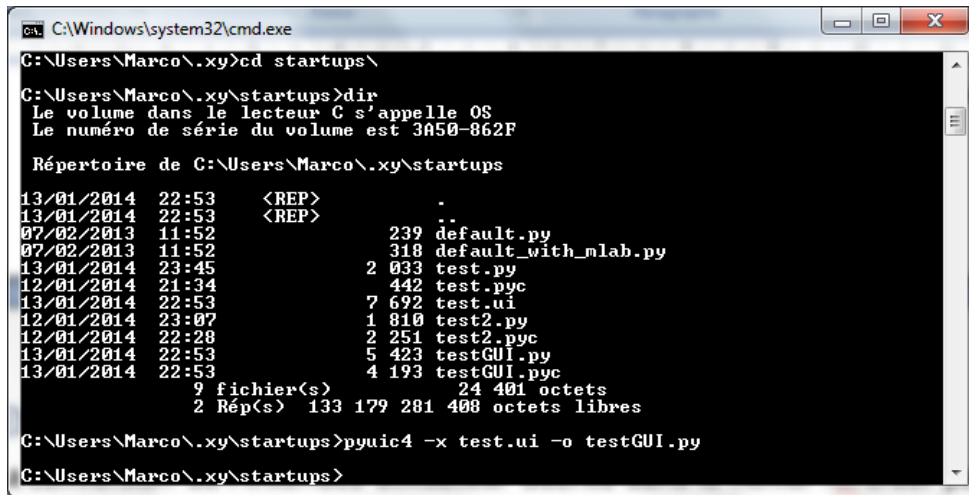
<http://fr.openclassrooms.com/informatique/cours/programmez-avec-le-langage-c/modeliser-ses-fenetres-avec-qt-designer>

Qt Designer créer un fichier .ui qui contient la description de la fenêtre au format xml. Cette description peut être convertie dans différents langages de programmation.

Etape 2 : Conversion .ui en .py

La conversion de l'interface utilisateur décrite dans le fichier .ui créer par Qt Designer doit être traduite en un script python.

On utilise pour cela la commande : **pyuic4 -x fichier.ui -o fichierGUI.py**

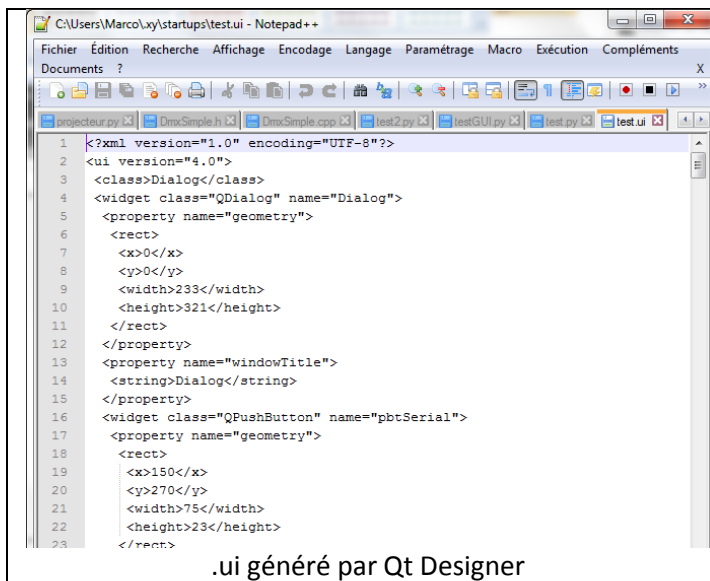


```
C:\Windows\system32\cmd.exe
C:\Users\Marco\xy>cd startups\
C:\Users\Marco\xy\startups>dir
Le volume dans le lecteur C s'appelle OS
Le numéro de série du volume est 3A50-862F

Répertoire de C:\Users\Marco\xy\startups

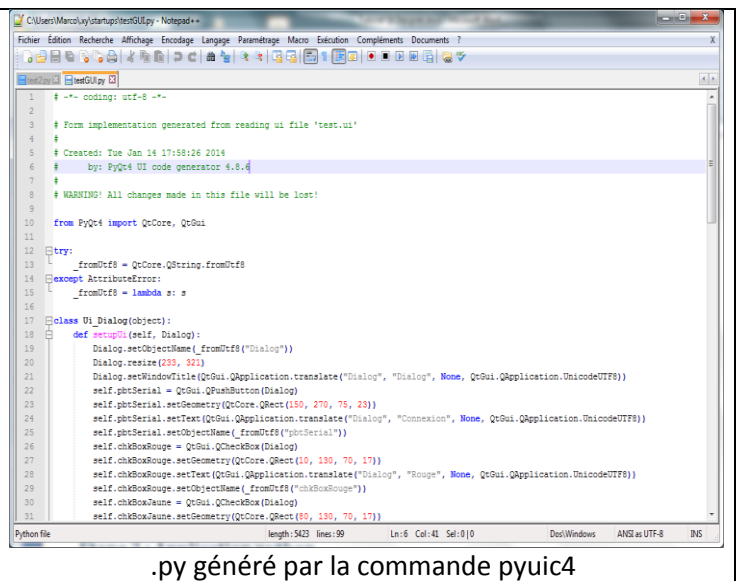
13/01/2014  22:53    <REP>          .
13/01/2014  22:53    <REP>          ..
07/02/2013  11:52             239 default.py
07/02/2013  11:52             318 default_with_mlab.py
13/01/2014  23:45              2 033 test.py
12/01/2014  21:34             442 test.pyc
13/01/2014  22:53              7 692 test.ui
12/01/2014  23:07             1 810 test2.py
12/01/2014  22:28              2 251 test2.pyc
13/01/2014  22:53              5 423 testGUI.py
13/01/2014  22:53              4 193 testGUI.pyc
                9 fichier(s)          24 401 octets
                2 Rép(s)    133 179 281 400 octets libres

C:\Users\Marco\xy\startups>pyuic4 -x test.ui -o testGUI.py
C:\Users\Marco\xy\startups>
```



```
<?xml version="1.0" encoding="UTF-8"?>
<ui version="4.0">
  <class>Dialog</class>
  <widget class="QDialog" name="Dialog">
    <property name="geometry">
      <rect>
        <x>0</x>
        <y>0</y>
        <width>233</width>
        <height>321</height>
      </rect>
    </property>
    <property name="windowTitle">
      <string>Dialog</string>
    </property>
    <widget class="QPushButton" name="pbtSerial">
      <property name="geometry">
        <rect>
          <x>150</x>
          <y>270</y>
          <width>75</width>
          <height>23</height>
        </rect>
      </property>
    </widget>
  </widget>
</ui>
```

.ui généré par Qt Designer



```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

# Form implementation generated from reading ui file 'test.ui'
#
# Created: Tue Jan 14 17:58:26 2014
# by: PyQt4 UI code generator 4.8.6
#
# WARNING! All changes made in this file will be lost!

from PyQt4 import QtCore, QtGui

try:
    _fromUtf8 = QtCore.QString.fromUtf8
except AttributeError:
    _fromUtf8 = lambda s: s

class Ui_Dialog(object):
    def setupUi(self, Dialog):
        Dialog.setObjectName(_fromUtf8("Dialog"))
        Dialog.resize(233, 321)
        Dialog.setWindowTitle(QtGui.QApplication.translate("Dialog", "Dialog", None, QtGui.QApplication.UnicodeUTF8))
        self.pbtSerial = QtGui.QPushButton(Dialog)
        self.pbtSerial.setGeometry(QtCore.QRect(150, 270, 75, 23))
        self.pbtSerial.setText(QtGui.QApplication.translate("Dialog", "Connexion", None, QtGui.QApplication.UnicodeUTF8))
        self.pbtSerial.setObjectName(_fromUtf8("pbtSerial"))
        self.chkBoxRouge = QtGui.QCheckBox(Dialog)
        self.chkBoxRouge.setGeometry(QtCore.QRect(10, 130, 70, 17))
        self.chkBoxRouge.setText(QtGui.QApplication.translate("Dialog", "Rouge", None, QtGui.QApplication.UnicodeUTF8))
        self.chkBoxJaune = QtGui.QCheckBox(Dialog)
        self.chkBoxJaune.setObjectName(_fromUtf8("chkBoxJaune"))
        self.chkBoxJaune.setGeometry(QtCore.QRect(80, 130, 70, 17))

        self.retranslateUi(Dialog)
        QtCore.QMetaObject.connectSlotsByName(Dialog)

    def retranslateUi(self, Dialog):
        Dialog.setWindowTitle(QtGui.QApplication.translate("Dialog", "Dialog", None, QtGui.QApplication.UnicodeUTF8))
        self.pbtSerial.setText(QtGui.QApplication.translate("Dialog", "Connexion", None, QtGui.QApplication.UnicodeUTF8))
        self.chkBoxRouge.setText(QtGui.QApplication.translate("Dialog", "Rouge", None, QtGui.QApplication.UnicodeUTF8))
        self.chkBoxJaune.setText(QtGui.QApplication.translate("Dialog", "Jaune", None, QtGui.QApplication.UnicodeUTF8))

if __name__ == '__main__':
    import sys
    app = QtGui.QApplication(sys.argv)
    dialog = Ui_Dialog()
    dialog.setupUi(dialog)
    sys.exit(app.exec_())
```

.py généré par la commande pyuic4

Etape 3 : Application python

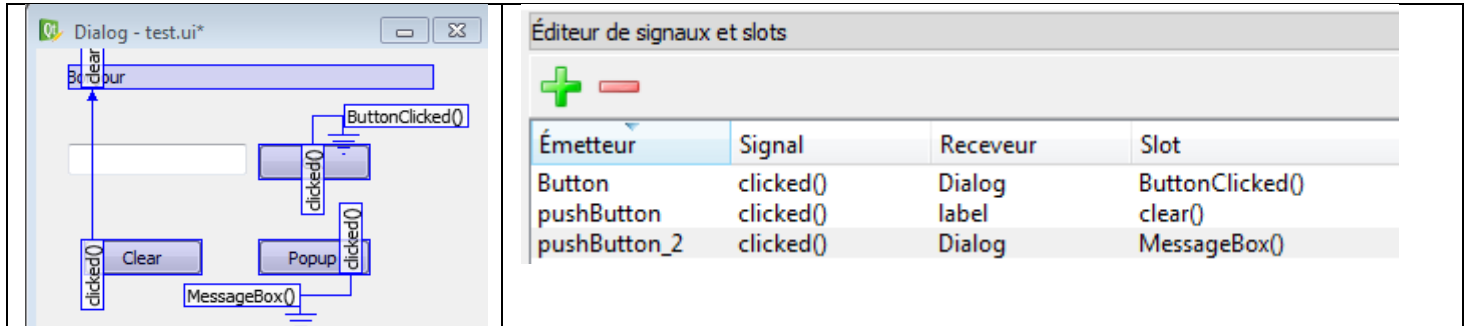
On crée une nouvelle application python dans laquelle on importe les librairies suivantes :

```
import sys
from PyQt4.QtGui import *
import fichierGUI
# fichierGUI est le fichier .py généré par pyuic4
```

On crée ensuite la classe **MainDialog** correspondant à la fenêtre de l'application :

```
class MainDialog(QDialog, testGUI.Ui_Dialog):
    def __init__(self, parent=None):
        super(MainDialog, self).__init__(parent)
        self.setupUi(self)
```

Dans cette classe, on décrit aussi les méthodes correspondantes aux slots de connexion aux signaux



```
def ButtonClicked(self):
    self.label.setText("Bonjour "+self.Edit.text())

def MessageBox(self):
    QMessageBox.information(self, "Popup", "Ceci est un message de test")
```

On exécute ensuite l'application :

```
app=QApplication(sys.argv)
form=MainDialog()
form.show()
app.exec_()
```