

Création d'une application en Python avec la bibliothèque QT

Introduction

Lors de la création d'une petite application en C++ avec la bibliothèque QT, j'ai découvert l'application QT designer que je trouve vraiment parfaite pour imaginer et construire des interfaces de manière interactive. Utilisant PHP depuis plusieurs années, j'ai cherché en vain un PHP-QT qui permettrait de construire des applications clientes exécutables de manière autonome. Il existe bien un PHP-GTK mais lorsque on a goûté à QT designer c'est un peu dur d'utiliser glade. Par contre il existe un PyQT qui permet de faire cela. Mais comme son nom le laisse supposer c'est avec Python. Bon, encore un nouveau langage informatique à apprendre ;-)
Dans cette série de tutoriels, je vous propose de découvrir avec moi Python et la bibliothèque QT.

Pré-requis

Sous kubuntu, il faut installer les paquets suivants :
qt3 designer
pyqt-tools
(python est déjà installé d'office)

1er programme

(source : <http://www.cs.usfca.edu/~afedosov/qttut/>)

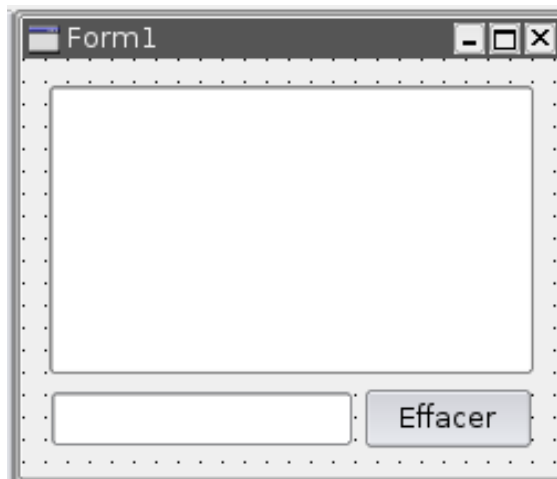
Avec Qt designer

(K/Développement/Qt3 Designer)

Créer un nouveau projet en choisissant Dialog

- Choisir le widget LineEdit pour créer une ligne en bas à gauche de Form1
- Choisir le widget ListBox pour créer une boîte à liste. Celle-ci contient par défaut une entrée que l'on peut supprimer en double cliquant dans la liste.
- Choisir le widget PushButton pour créer un bouton en bas à droite. Un double clic sur celui-ci permet de modifier le texte. Taper « Effacer ».
- Passer la propriété autoDefault à False (A droite dans l'onglet Property vers le bas de la liste)

A ce stade, vous devriez disposer de ceci :



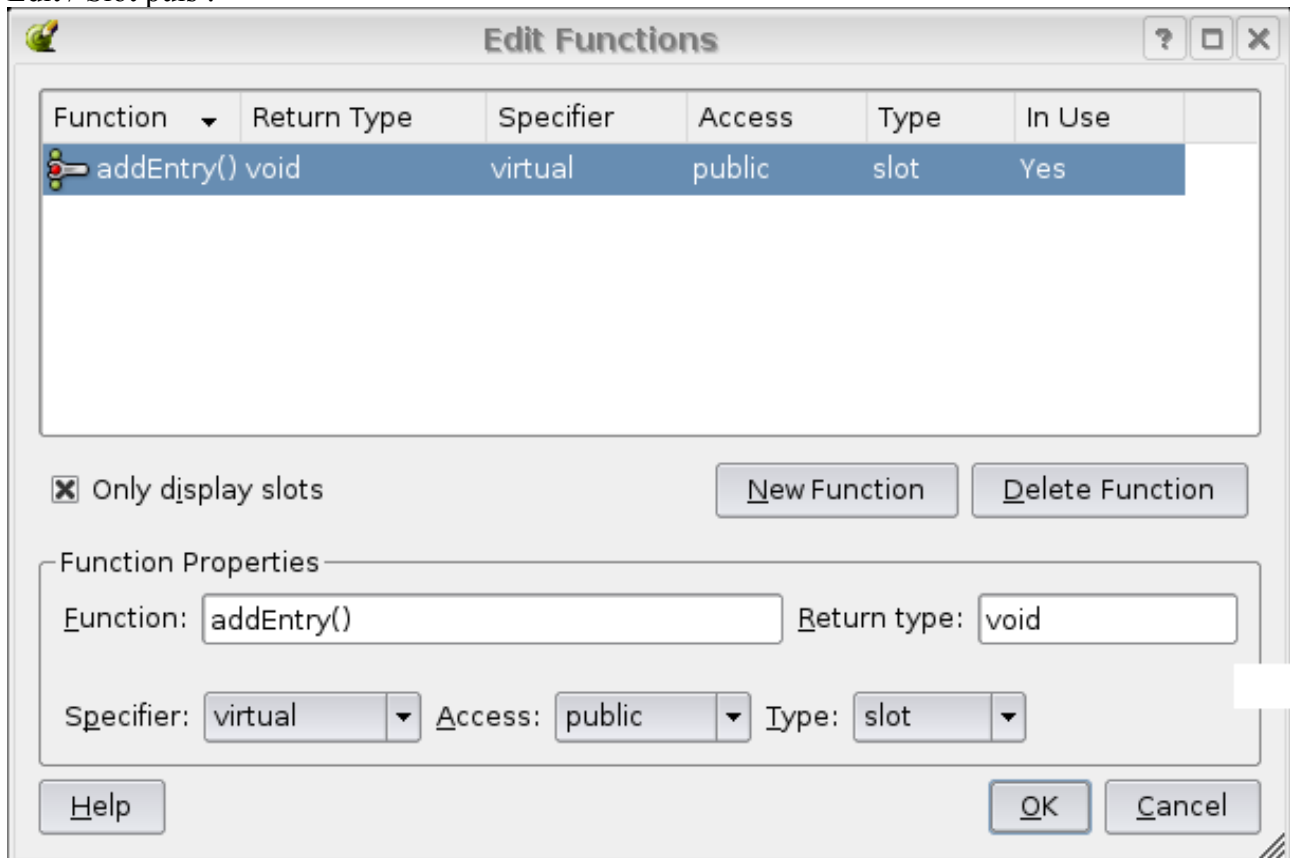
On va maintenant s'occuper de la partie dynamique de notre boîte de dialogue. C'est à dire qu'on va

indiquer ce qu'il faut faire en réponse aux actions de l'utilisateur. Dans la terminologie de QT un signal est généré lorsque l'utilisateur agit sur un widget. Celui-ci sera envoyé à un slot qui correspond à une fonction de traitement.

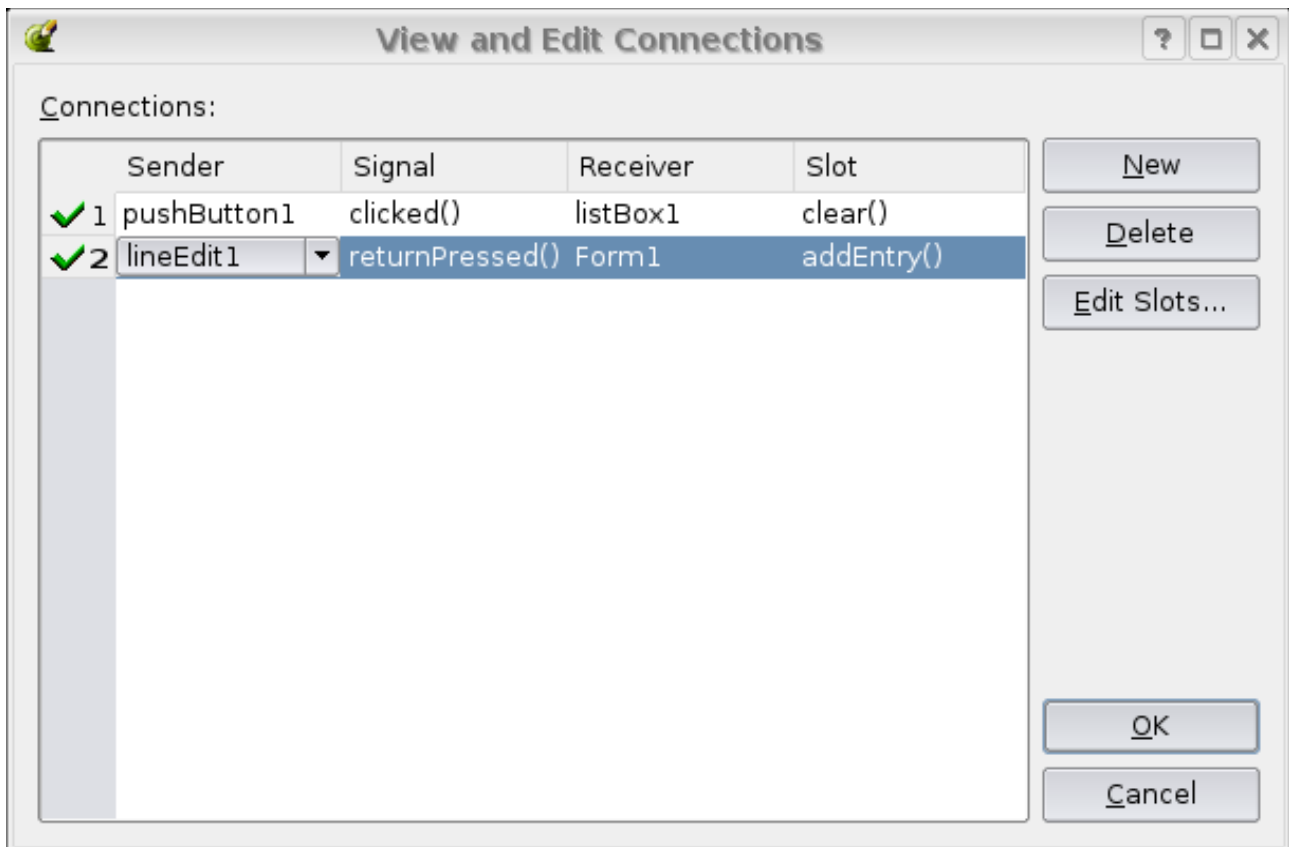
Nous voulons effacer le contenu de la listBox lorsque l'utilisateur clique sur le bouton « Effacer ». Pour cela, on va utiliser le bouton « Connect Signal/Slots » qu'on peut aussi appeler avec la touche F3, puis faire un glisser/déposer du pushBotton1 vers listBox1. La sélection des éléments est signalé par un cadre rouge. La boîte « View and Edit Connections » apparaît. Il faut choisir le signal « clicked » et le slot « clear() ».

On va faire la même opération pour « lineEdit1 », mais nous devons au préalable créer un slot, car il ne s'agit pas d'une fonction prédéfini.

Edit / Slot puis :



F3 puis faire un glisser/déposer du lineEdit1 vers listBox1 et compléter pour obtenir :



Il faut maintenant implémenter la fonction addEntry().

Faire un double clic sur form1.ui.h (En haut à droite sous Form1:/...

Il faut compléter la fonction

```
void Form1::addEntry()
{
    e = self.lineEdit1.text().ascii()
    self.listBox1.insertItem(e)
    self.lineEdit1.clear()
}
```

Voilà, c'est terminé. Il faut enregistrer le fichier.

En ligne de commande, on va créer le programme python avec :

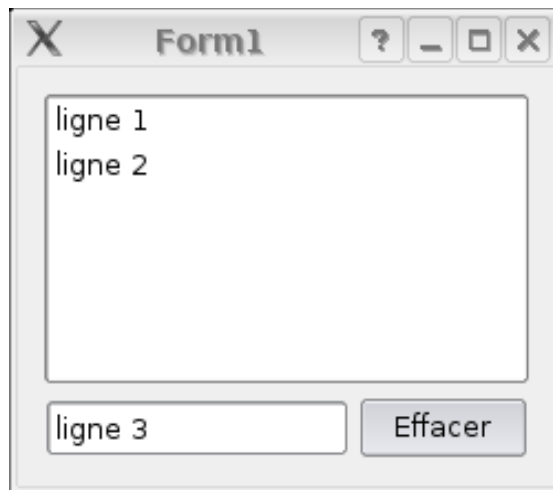
```
pyuic form1.ui > form1.py
```

Il ne reste plus qu'à créer le fichier mygui.py

```
from qt import *
from form1 import *
import sys
if __name__ == "__main__":
    app = QApplication(sys.argv)
    f = Form1()
    f.show()
    app.setMainWidget(f)
    app.exec_loop()
```

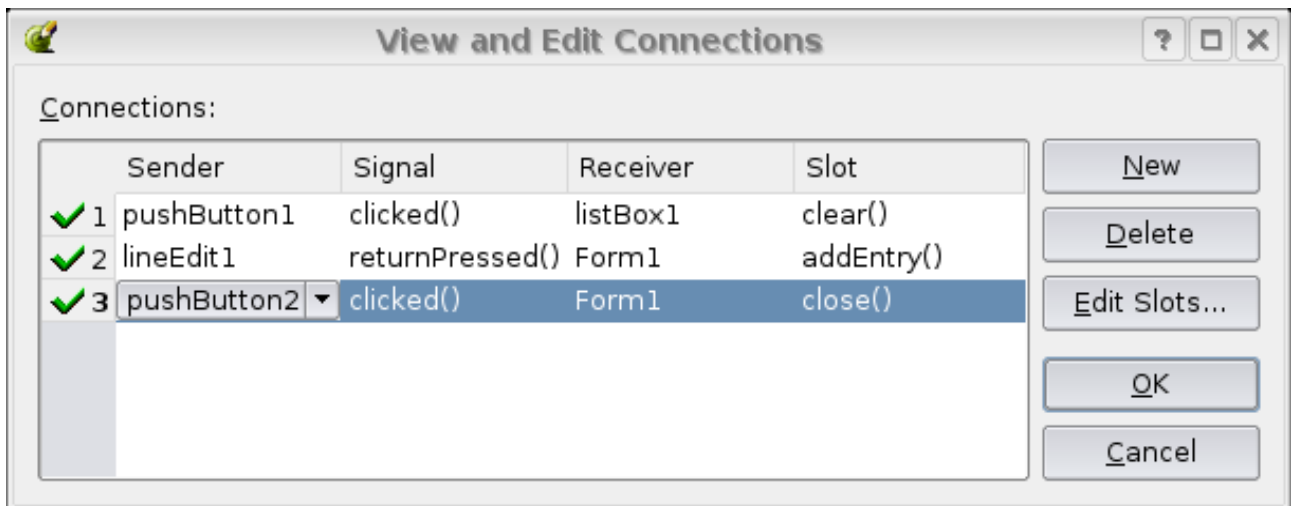
Puis à lancer notre programme avec

```
python mygui.py
```

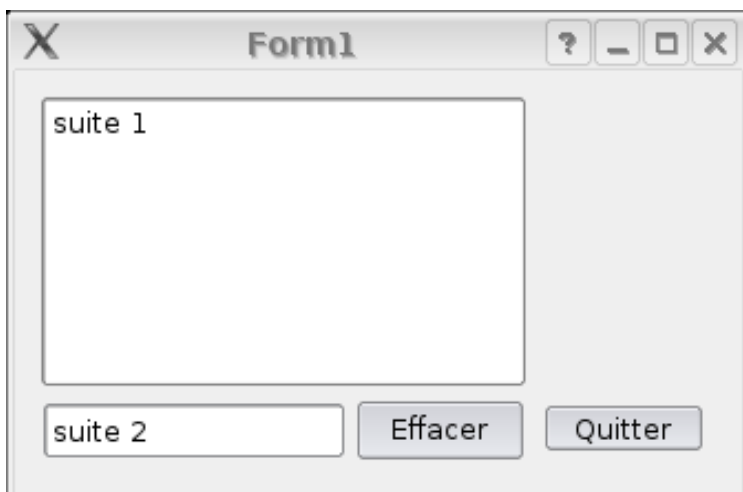


Ajouter le bouton quitter à notre boîte de dialogue

Reprendre les explications précédentes pour obtenir ceci.



Cette fois ci, il suffit de recréer le fichier form1.py et d'exécuter mygui.py pour obtenir



Si votre boîte de dialogue se ferme lorsque vous validez une ligne, c'est que vous avez oublié de mettre l'option `AutoDefault` à `false` pour le bouton « Quitter ».

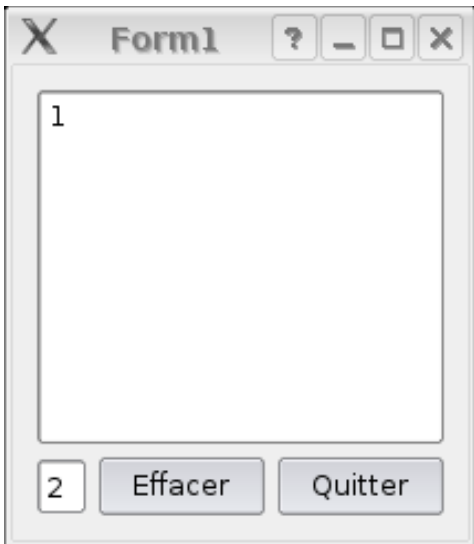
Améliorer l'aspect de votre boîte de dialogue

Vous n'en avez pas marre de ces boîtes de dialogue qu'on ne peut pas redimensionner. Avec QT vous

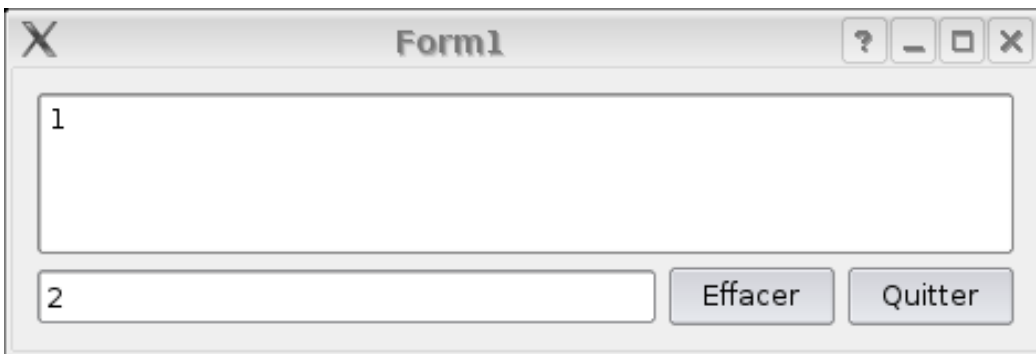
allez pouvoir faire de jolies boites qui s'agrandissent en redimensionnant/déplaçant leurs éléments.

- Sélectionner les 3 éléments situés en bas et cliquer sur le bouton « Lay Out Horizontally » ou Ctrl-H.
- Sélectionner tous les éléments et cliquer sur le bouton « Lay Out Vertically » ou Ctrl-L.
- Cliquer sur le fond de Form1, puis sur Ctrl-L ou Ctrl-H pour que les widgets occupent la totalité le l'espace.

Après compilation de form1.py vous devriez obtenir ceci



ou bien



C'est terminé

Voilà, pour cette première présentation, on a fait plus de QT designer que de Python. Mais c'est mon premier jour de Python, il faut me laisser un peu de temps pour que je puisse découvrir ce langage.

J'ai trouvé ces liens pour approfondir le sujet

<http://wikipython.flibuste.net/moin.py/PyQt>

<http://docs.python.org/> (en anglais)

et encore 2 autres (merci [Nemo])

<http://fr.diveintopython.org/toc.html>

<http://www.ulg.ac.be/cifen/inforef/swi/python.htm>