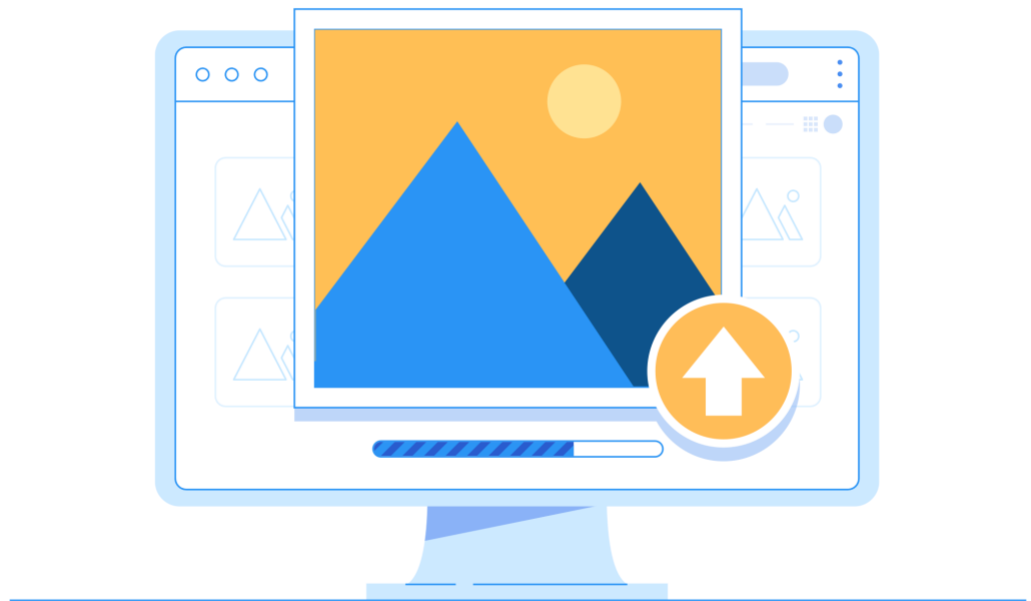


BAB 10

PyQt Designer



10.1 Tujuan

1. Dapat memahami pembuatan aplikasi GUI menggunakan PyQt Designer
2. Dapat melakukan pemaketan aplikasi PyQt Designer dengan Python

10.2 Pengantar

10.2.1 Pengenalan Qt Designer

Qt Designer adalah tool GUI yang digunakan untuk merancang dan membangun antarmuka pengguna dari aplikasi Qt. Dengan Qt Designer kita dapat merancang tata letak widget secara visual dan juga membuat skrip untuk widget tersebut. Qt Designer sangat berguna bagi mereka yang ingin merancang antarmuka pengguna tanpa harus menulis kode untuk itu.

10.2.2 Instalasi Qt Designer

Qt Designer secara otomatis akan terinstall ketika kita melakukan instalasi PyQt5. Jika telah menginstal PyQt5 melalui pip, maka seharusnya Qt Designer juga sudah terinstall pada sistem. Untuk menjalankan Qt Designer, ketikkan perintah designer pada Command Prompt.

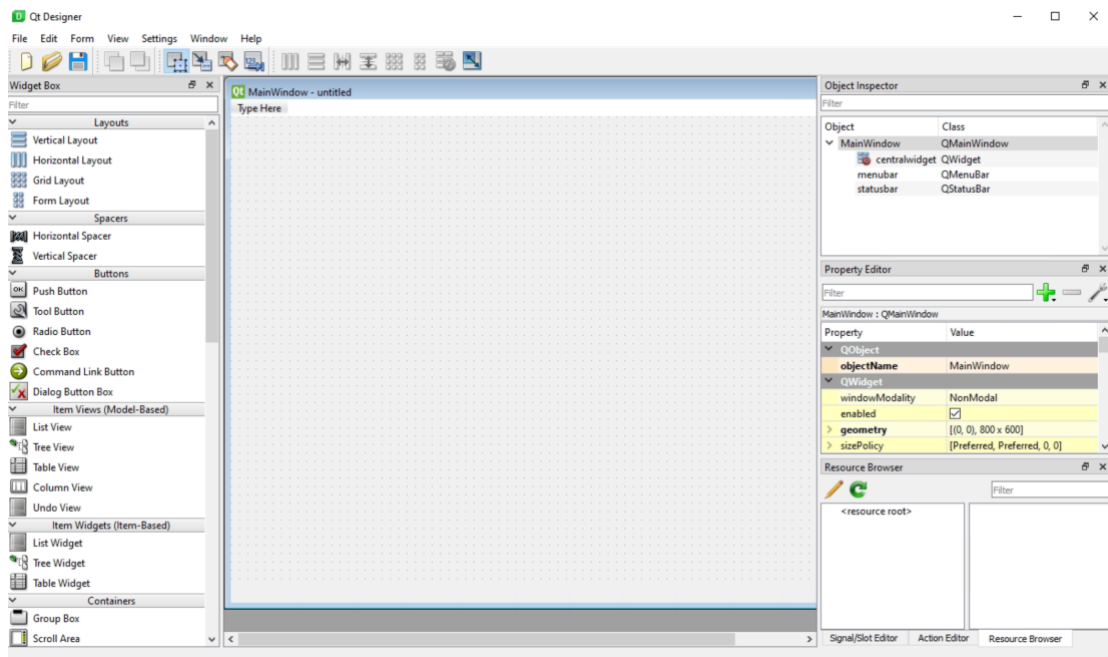


```
C:\WINDOWS\system32\cmd. X + v
Microsoft Windows [Version 10.0.22621.1848]
(c) Microsoft Corporation. All rights reserved.

C:\Users\AIO ASUS>designer

C:\Users\AIO ASUS>
```

Tampilan dari PyQt Designer dapat dilihat pada gambar berikut



- Widget Box: Daftar semua widget yang dapat kita gunakan dalam desain Anda.
- Object Inspector: Menampilkan hierarki widget dalam desain.
- Property Editor: Menampilkan properti dari widget yang dipilih.
- Signal & Slot Editor: Memungkinkan kita untuk mengedit sinyal dan slot dari widget.
- Form Editor: Area kerja utama di mana kita merancang tampilan aplikasi.

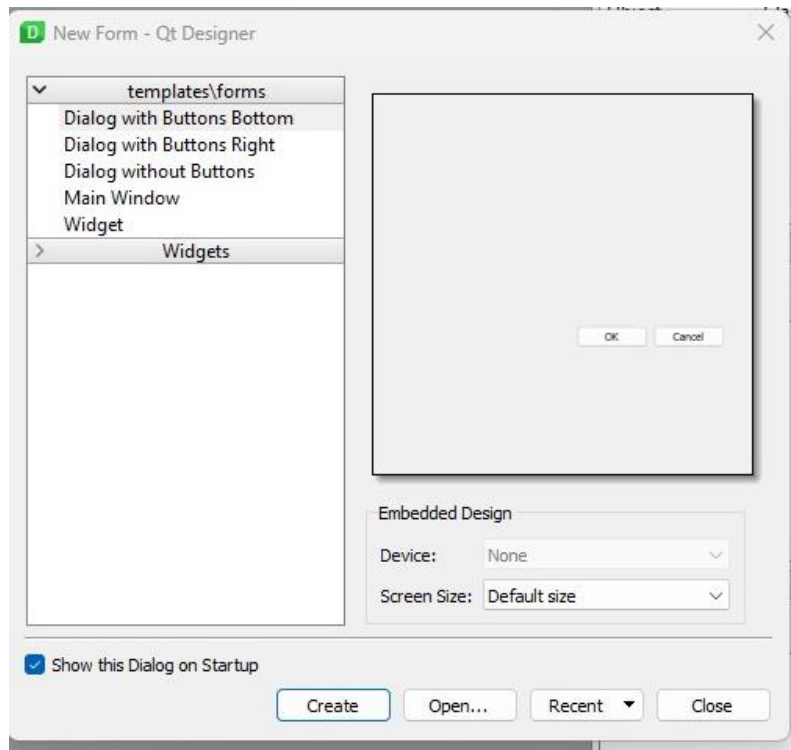
Jika muncul error, maka kita harus melakukan instalasi Qt Designer dengan menggunakan perintah

> pip install PyQt5Designer

```
C:\Users\AIO ASUS>pip install PyQt5Designer
Collecting PyQt5Designer
  Downloading PyQt5Designer-5.14.1-py3-none-win_amd64.whl (40.8 MB)
    40.8/40.8 MB 9.5 MB/s eta 0:00:00
Installing collected packages: PyQt5Designer
Successfully installed PyQt5Designer-5.14.1
```

10.2.3 Membuat Form dengan Qt Designer

Untuk membuat form baru, pilih File > New. kita akan diminta untuk memilih jenis form yang ingin dibuat.



Template form yang tersedia antara lain :

- Dialog with Buttons Bottom: template untuk dialog yang memiliki baris tombol di bagian bawah. Tombol ini biasanya digunakan untuk menerima atau membatalkan operasi.
- Dialog with Buttons Right: template untuk dialog yang memiliki kolom tombol di sisi kanan.
- Dialog without Buttons: template untuk dialog yang tidak memiliki tombol. Kita bebas menambahkan widget lain ke dalam dialog ini.
- Main Window: template untuk jendela utama aplikasi. Jendela utama biasanya memiliki menu, toolbar, dan area konten di mana widget lain dapat ditempatkan.

- Widget: template untuk widget kustom. Widget adalah elemen dasar antarmuka pengguna dalam Qt dan dapat digabungkan untuk membuat antarmuka yang lebih kompleks.

Setelah kita memilih jenis form, kita akan dibawa ke Form Editor. Di sini, kita dapat mulai menambahkan widget ke form dari Widget Box.

10.2.4 Mengubah UI menjadi Kode Python

Jika kita menyimpan file dari PyQt Designer, maka format yang diberikan adalah .ui . Untuk itu kita perlu mengubah format tersebut ke dalam kode python. Kita dapat melakukan ini dengan menggunakan tool pyuic yang disertakan dengan PyQt5.

Sebagai contoh jika memiliki file ui dengan nama mainwindow.ui. Aktifkan command prompt, Jalankan perintah berikut di terminal Anda:

```
1. pyuic5 -x mainwindow.ui -o mainwindow.py
```

Setelah kita menjalankan perintah ini, kita akan mendapatkan file Python dengan nama mainwindow.py yang berisi kode untuk form. Namun kode tersebut belum dapat langsung kita jalankan, kita perlu membuat kode file baru yang berfungsi mengimpor file tersebut. Buat file baru beri nama dengan demo.py

```
1. from PyQt5.QtWidgets import QApplication
2. from mainwindow import Ui_MainWindow
3.
4. if __name__ == "__main__":
5.     app = QApplication([])
6.     window = Ui_MainWindow()
7.     window.show()
8.     app.exec_()
```

Dalam kode di atas, Ui_MainWindow adalah kelas yang dihasilkan oleh pyuic5. Perhatikan bahwa nama mainwindow pada baris 2 harus sesuai dengan nama file

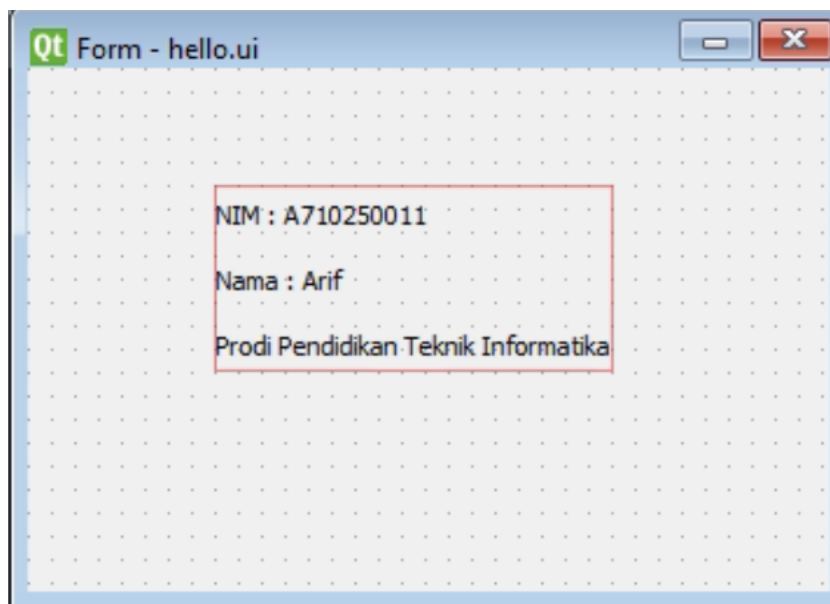
Python yang dihasilkan oleh pyuic5. Untuk menjalankannya ketik perintah berikut pada command prompt

```
1. python demo.py
```

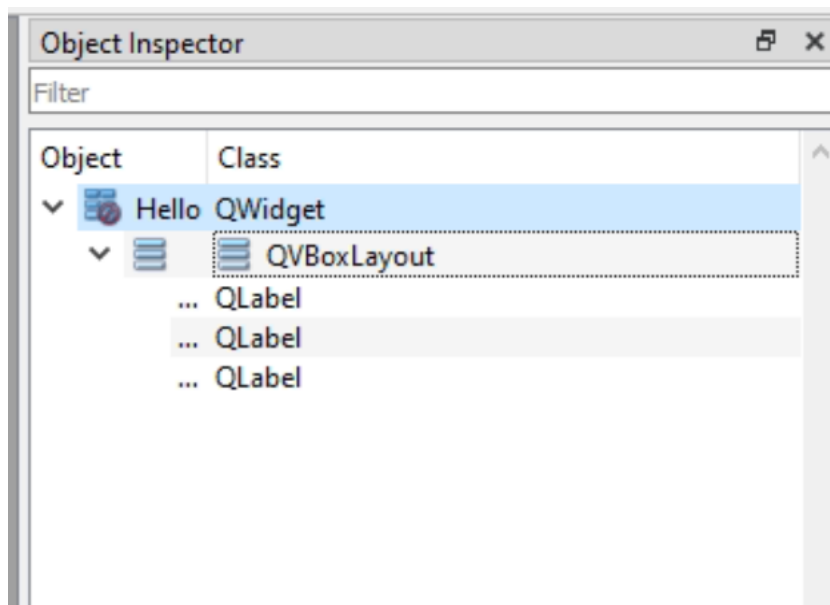
10.3 Kegiatan Praktikum

10.3.1 Kegiatan 1 : Membuat Aplikasi Pertama

1. Buat rancangan layout berikut menggunakan Qt Designer, Pilih Form dengan tipe **Widget**. simpan dengan nama hello.ui.



2. Perhatikan bahwa rancangan UI tersebut menggunakan Vertical Layout yang didalamnya terdapat 3 QLabel



3. Buka command prompt pada folder yang sama dengan file hello.ui kemudian jalankan perintah berikut

```
1. pyuic5 hello.ui -o hello.py
```

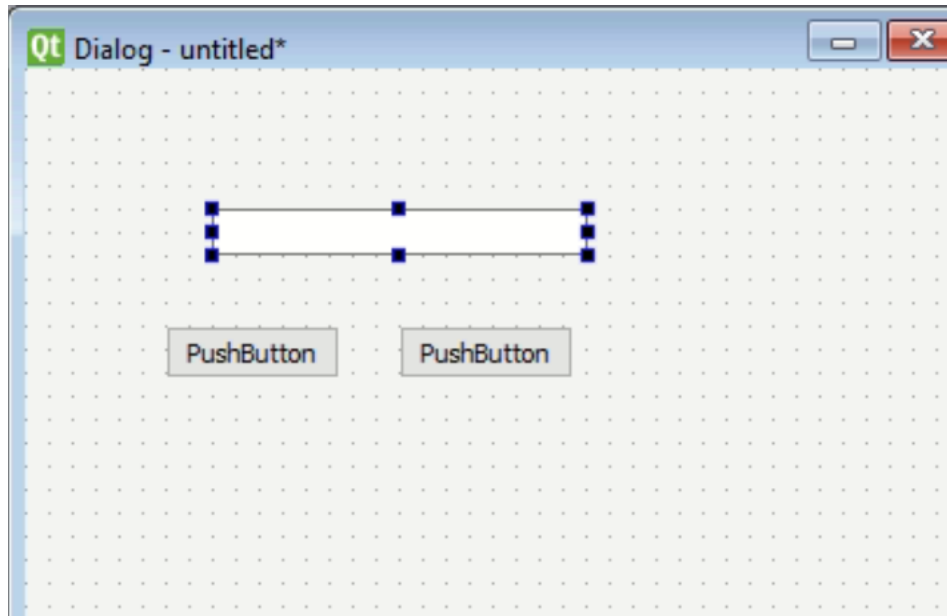
4. Buat file baru dengan nama mainhello.py

```
1. from PyQt5.QtWidgets import QApplication, QMainWindow
2. from hello import Ui_Hello
3.
4. class HelloWorld(QMainWindow, Ui_Hello):
5.     def __init__(self, parent=None):
6.         super(HelloWorld, self).__init__(parent)
7.         self.setupUi(self)
8.
9. app = QApplication([])
10. window = HelloWorld()
11. window.show()
12. app.exec_()
```

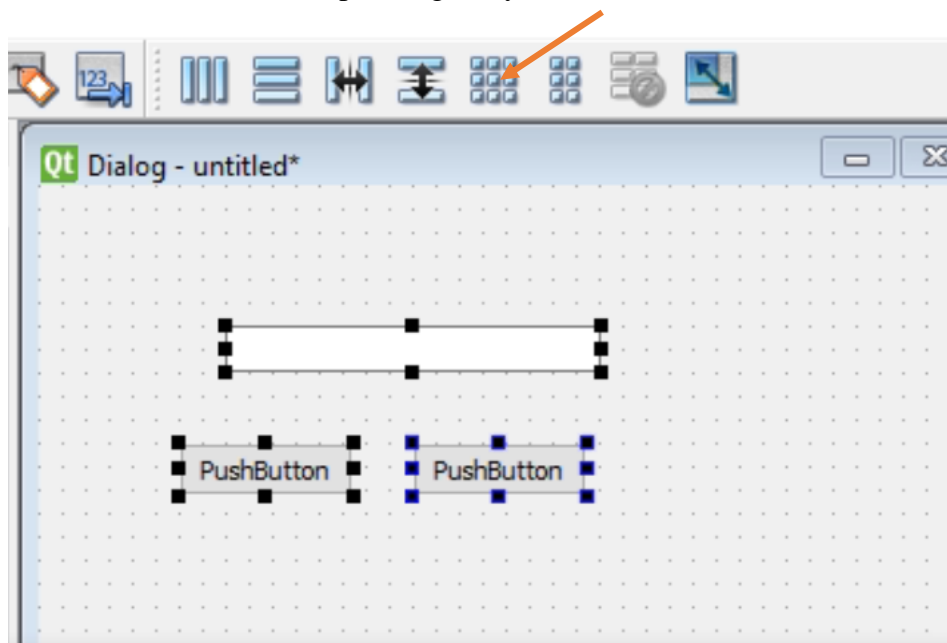
5. Jalankan aplikasi dengan mengetikkan perintah “ python mainhello.py “ pada command prompt

10.3.2 Kegiatan 2 : Signal dan Slot pada PyQt Designer

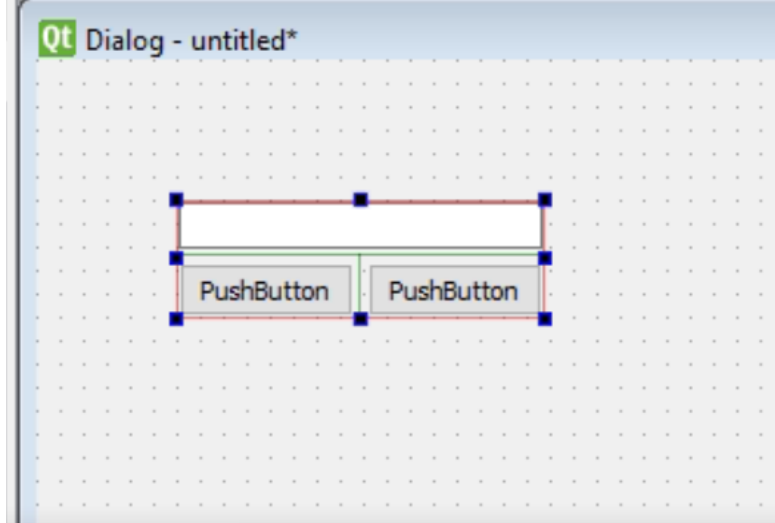
1. Buat rancangan layout berikut menggunakan Qt Designer, Pilih Form dengan tipe **Dialog Without Button**. simpan dengan nama signalslot.ui. Perhatikan bahwa peletakan QLineEdit dan QPushButton masih berantakan.



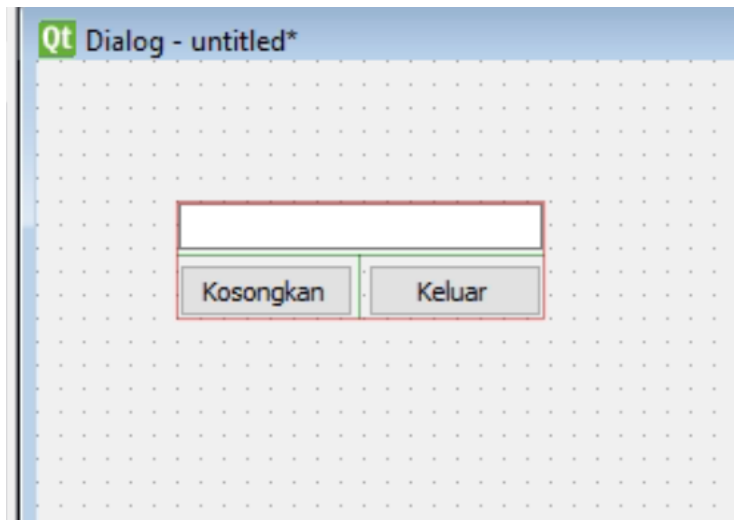
2. Seleksi semua komponen dengan cara tahan Ctrl + Klik QPushButton dan QLineEdit, kemudian pilih logo Layout in Grid




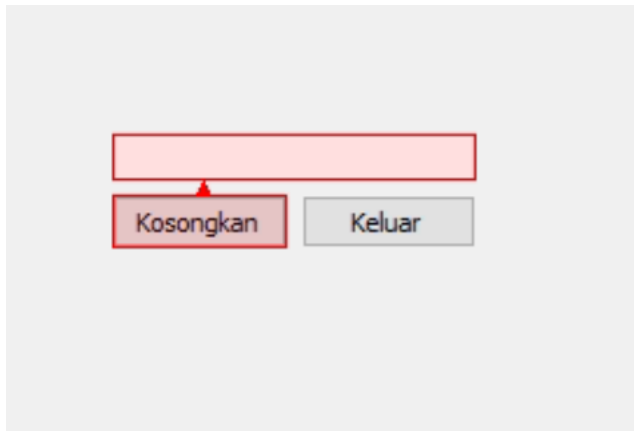
3. Secara otomatis maka susunan widget akan menjadi rapi menyesuaikan layout



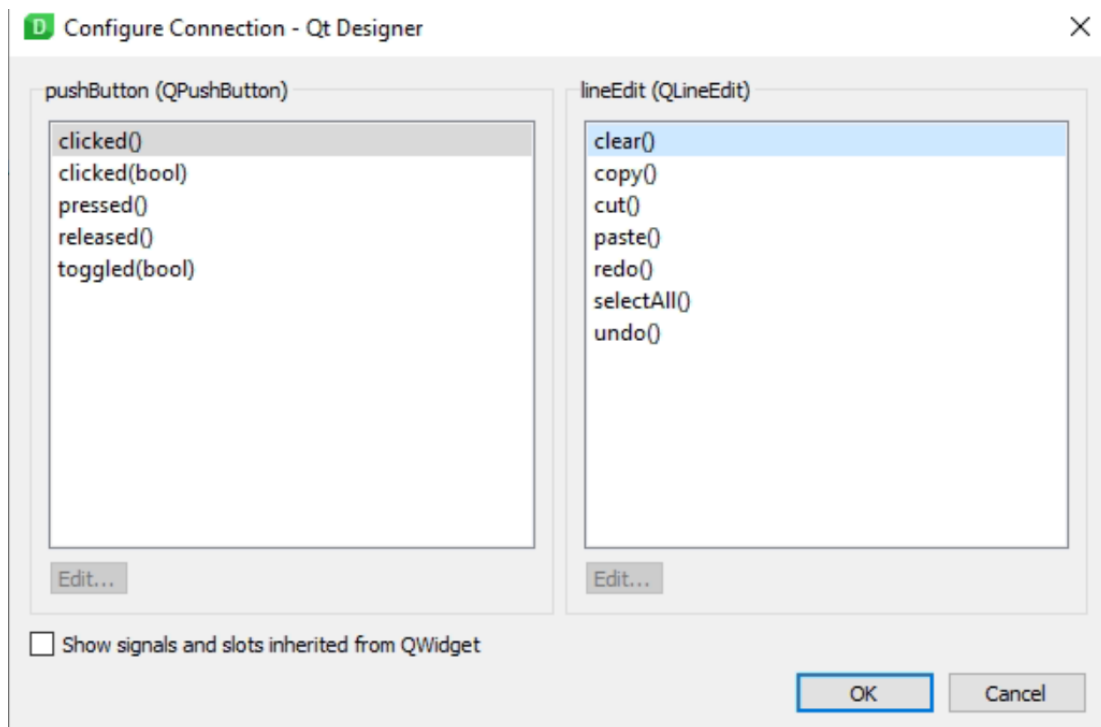
4. Ubah nama PushButton 1 dan 2 menjadi **Kosongkan** dan **Keluar**



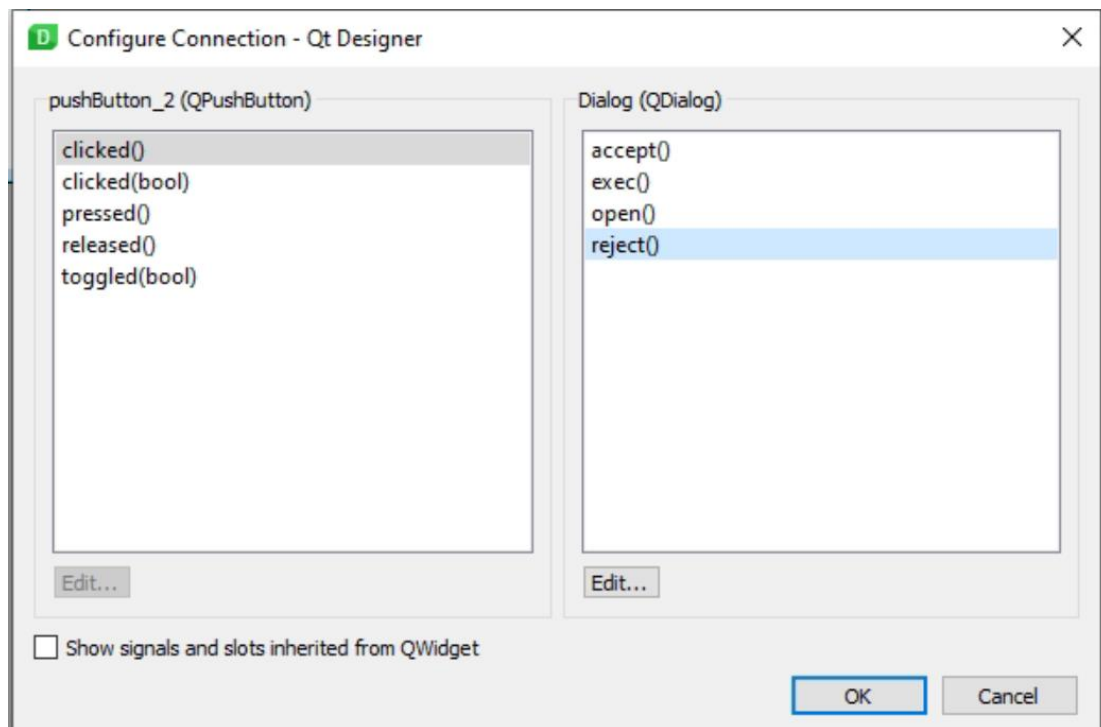
5. Masuk ke Mode Edit Signal/Slot dengan klik logo  pada toolbar. Kemudian klik button **Kosongkan** dan drag ke QlineEdit, maka akan muncul dialog baru, pilih Clicked() dan Clear()



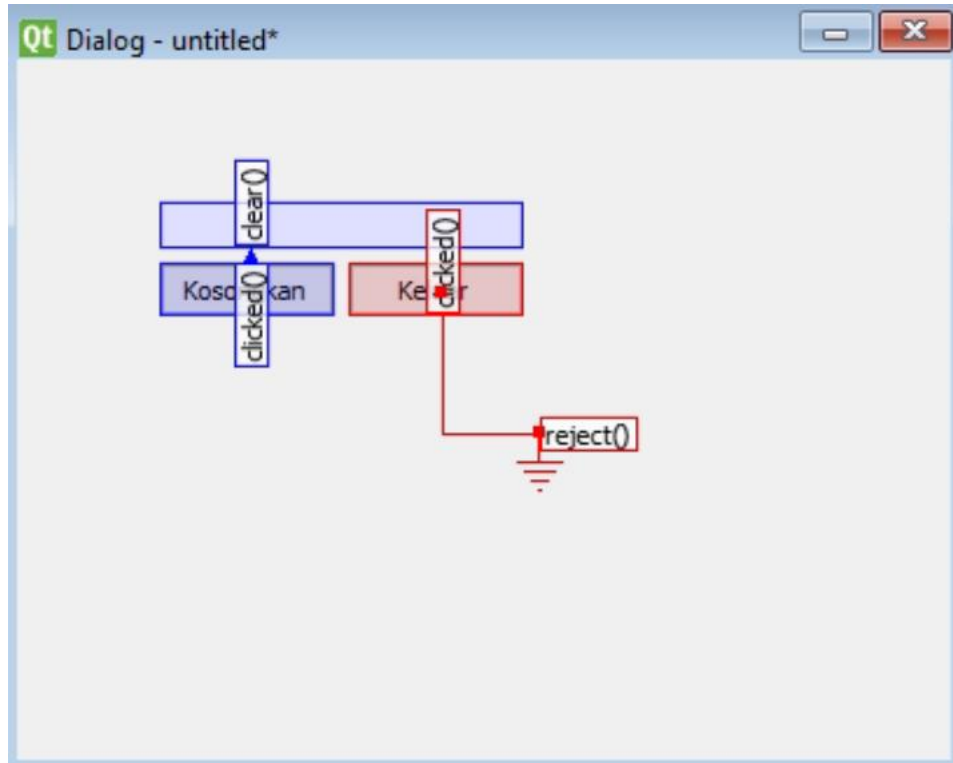
Klik button Kosongkan kemudian drag ke QLineEdit, sehingga tampilan akan menjadi seperti disamping



6. Lakukan hal yang sama untuk klik button **Keluar** dan namun drag ke form kosong, maka akan muncul dialog baru, pilih Clicked() dan reject()



7. Tampilan Form akan menjadi seperti berikut



6. Buka command prompt pada folder yang sama dengan file signalslot.ui kemudian jalankan perintah berikut

```
pyuic5 signalslot.ui -o signalslot.py
```

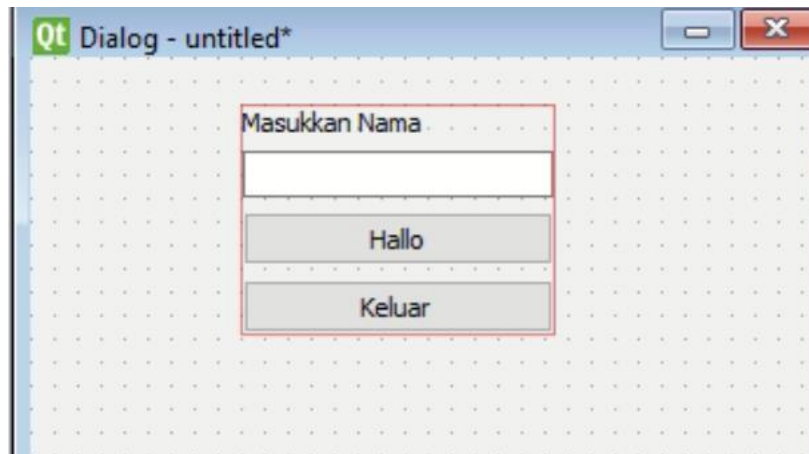
7. Buat file baru dengan nama mainsignalslot.py

```
1. from PyQt5.QtWidgets import QApplication, QDialog
2. from signalslot import Ui_Dialog
3.
4. class DialogWindow(QDialog, Ui_Dialog):
5.     def __init__(self, parent=None):
6.         super(DialogWindow, self).__init__(parent)
7.         self.setupUi(self)
8.
9. app = QApplication([])
10.window = DialogWindow()
11.window.show()
12.app.exec_()
```

8. Jalankan aplikasi dengan mengetikkan perintah “ python mainsignalslot.py “ pada command prompt

10.3.3Kegiatan 3 : Memproses Input

1. Buat rancangan layout berikut menggunakan Qt Designer, Pilih Form dengan tipe **Dialog Without Button**. simpan dengan nama form.ui. Beri nama object QLineEdit menjadi nameLineEdit dan QPushButton “**Halo**” menjadi showMessageButton.



2. Buka command prompt pada folder yang sama dengan file form.ui kemudian jalankan perintah berikut

```
pyuic5 form.ui -o form.py
```

3. Buat file baru dengan nama mainform.py

```
1. from PyQt5.QtWidgets import QApplication, QDialog, QMessageBox
2. from form import Ui_Dialog
3.
4. class Dialog(QDialog, Ui_Dialog):
5.     def __init__(self, parent=None):
6.         super(Dialog, self).__init__(parent)
7.         self.setupUi(self)
8.         self.showMessageButton.clicked.connect(self.show_message)
9.
10.    def show_message(self):
11.        name = self.nameLineEdit.text()
12.        QMessageBox.information(self, "Hello", f"Hello, {name}!")
13.
14. app = QApplication([])
15. dialog = Dialog()
16. dialog.show()
17. app.exec_()
```

4. Jalankan aplikasi dengan mengetikkan perintah “ python mainform.py ” pada command prompt

10.4 Tugas

1. Dengan menggunakan PyQt Designer, rancanglah form dengan tampilan berikut. Terdapat 2 input utama, input harga dengan nominal rupiah dan input pajak dengan nilai 5%, 10% dan 15%. Jika user melakukan klik pada tombol **Hitung Pajak** maka akan menampilkan Total harga beserta pajak yang harus dibayar

