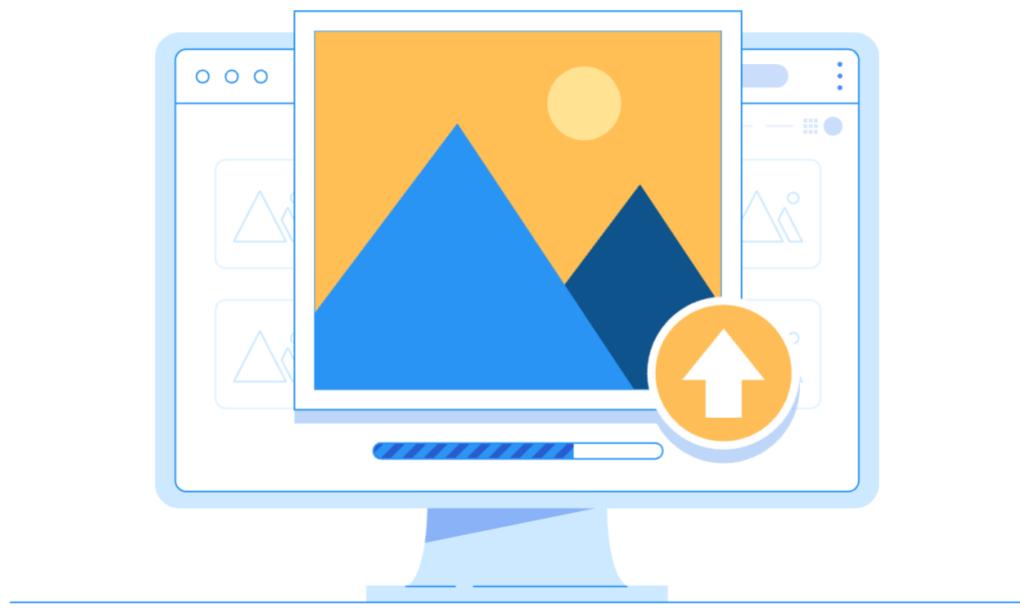


BAB 11

PyQt Designer



11.1 Tujuan

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

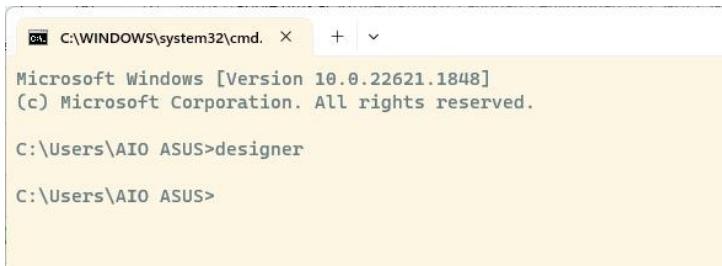
11.2 Pengantar

11.2.1 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.

11.2.2 Instalasi Qt Designer

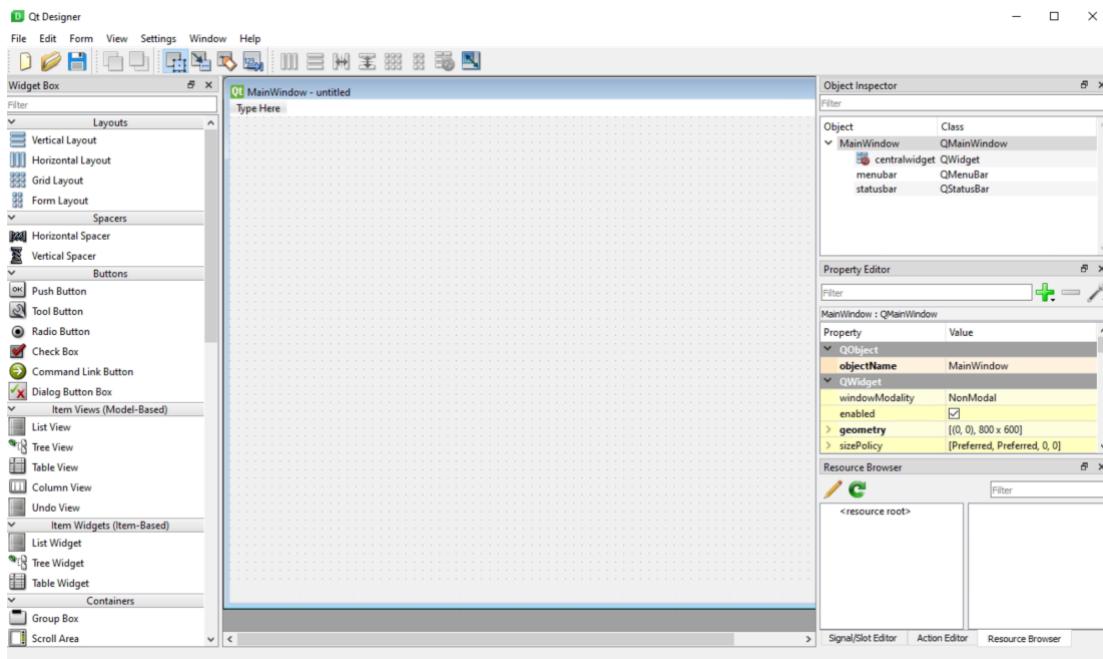
Qt Designer secara otomatis akan terinstall ketika kita melakukan instalasi PyQt6. Jika telah menginstal PyQt6 melalui pip, maka seharusnya Qt Designer juga sudah terinstall pada sistem. Untuk menjalankan Qt Designer, ketikkan perintah designer pada Command Prompt.



```
C:\WINDOWS\system32\cmd. x + ^
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.

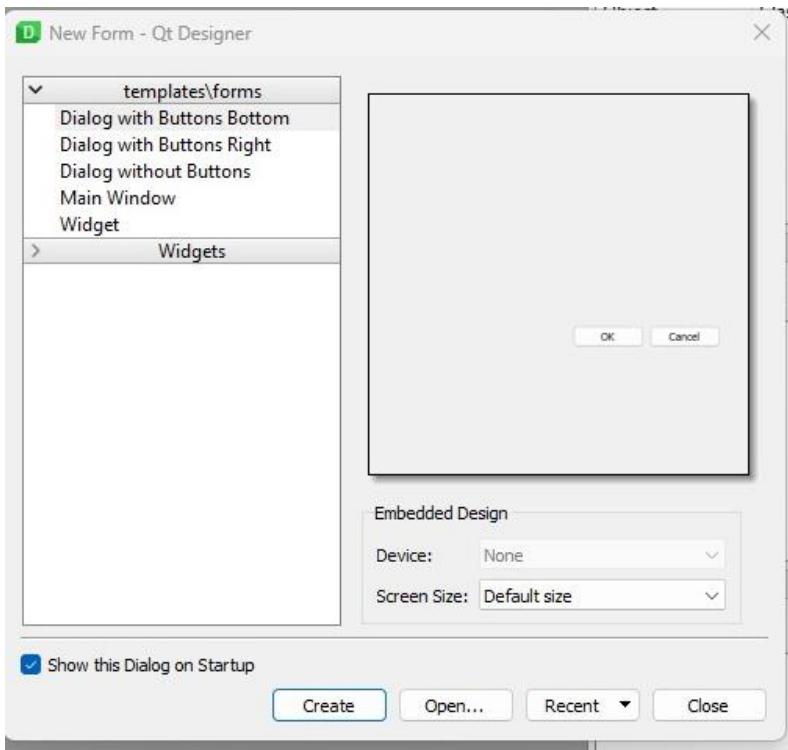
Jika muncul error, maka kita harus melakukan instalasi Qt Designer dengan menggunakan perintah

> pip install PyQt6-tools

```
C:\Users\AIO ASUS>
C:\Users\AIO ASUS>pip install PyQt6-tools
Collecting PyQt6-tools
  Downloading PyQt6_tools-6.4.2.3.3-py3-none-any.whl (29 kB)
Requirement already satisfied: click in c:\users\ aio asus\appdata\local\programs\python\python311\lib\site-packages (from PyQt6-tools) (8.1.3)
Collecting PyQt6==6.4.2
  Downloading PyQt6-6.4.2-cp37-abi3-win_amd64.whl (6.4 MB)
  6.4/6.4 MB 20.5 MB/s eta 0:00:00
Collecting PyQt6-plugins<6.4.2.3,>=6.4.2.2.2
  Downloading PyQt6_plugins-6.4.2.2.3-cp311-cp311-win_amd64.whl (72 kB)
  72.6/72.6 kB ? eta 0:00:00
Requirement already satisfied: python-dotenv in c:\users\ aio asus\appdata\local\programs\python\python311\lib\site-packages (from PyQt6-tools) (1.0.0)
```

11.2.3 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.

11.2.4 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 PyQt6.

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

```
1. pyuic6 -x mainwindow.ui -o mainwindow.py
```

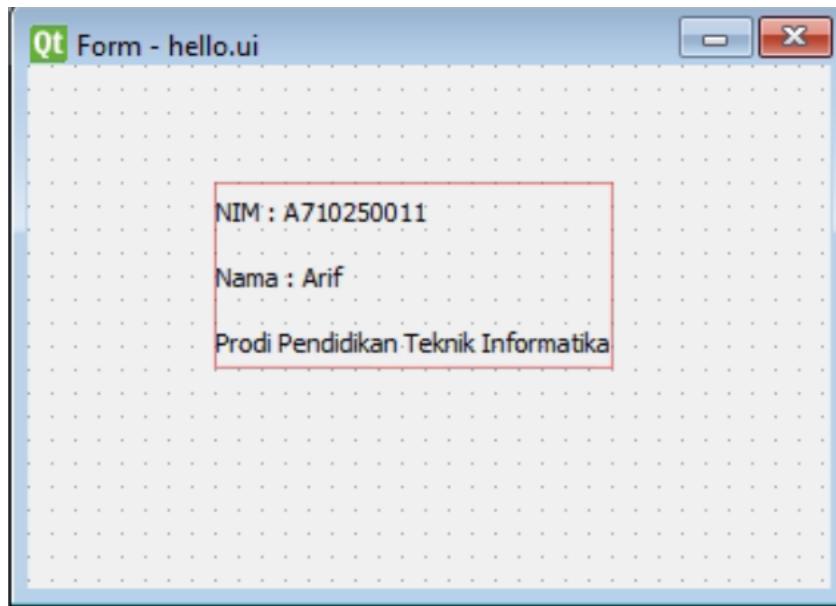
Setelah kita menjalankan perintah ini, kita akan mendapatkan file Python dengan nama mainwindow.py yang berisi kode untuk form. Untuk menjalankan program tersebut ketik perintah berikut pada command line

```
1. python .\coba.py
```

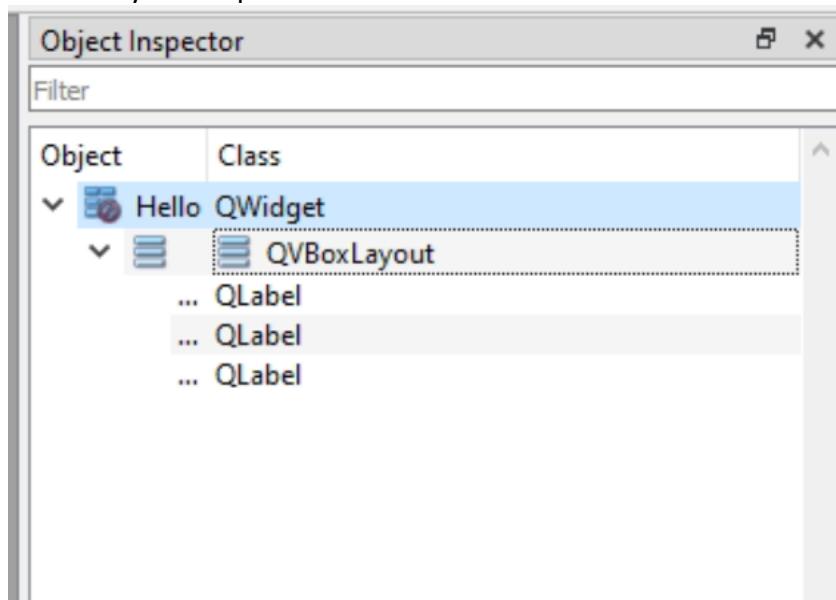
11.3 Kegiatan Praktikum

11.3.1 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



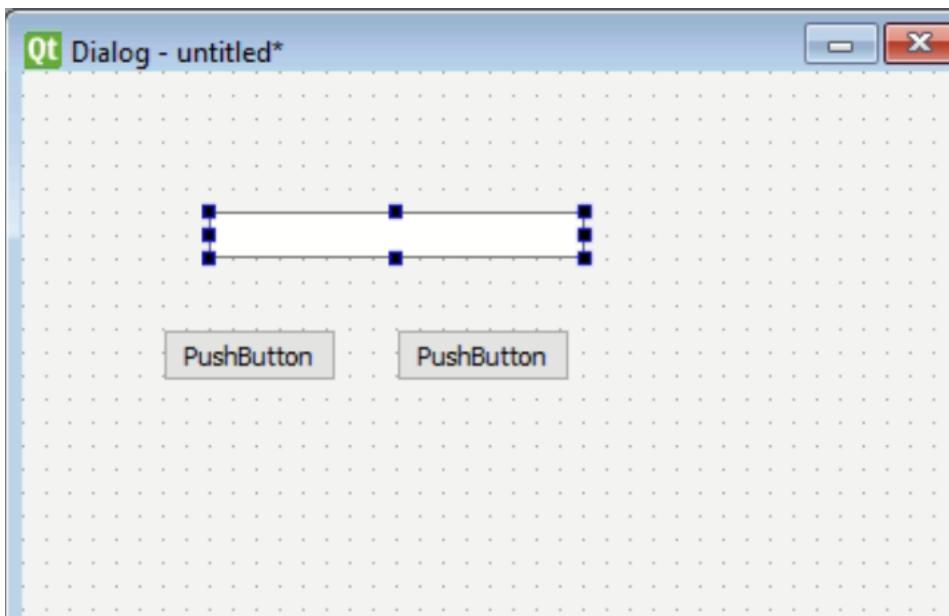
3. Buka command prompt pada folder yang sama dengan file hello.ui kemudian jalankan perintah berikut

```
1. pyuic6 -x hello.ui -o hello.py
```

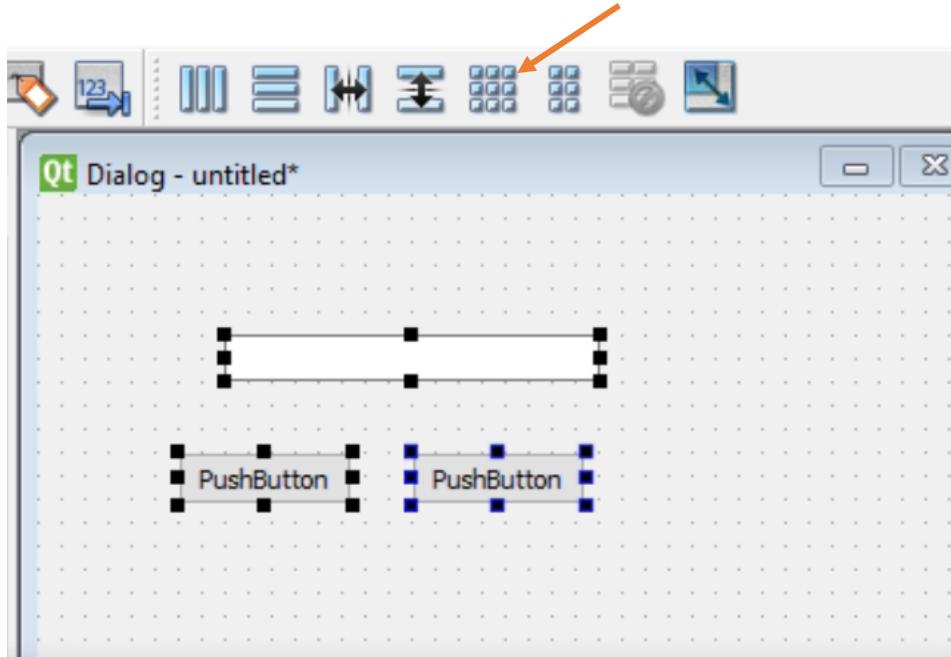
4. Jalankan aplikasi dengan mengetikkan perintah “ python hello.py ” pada command prompt

11.3.2 Kegiatan 2 : Signal dan Slot pada PyQt Designer

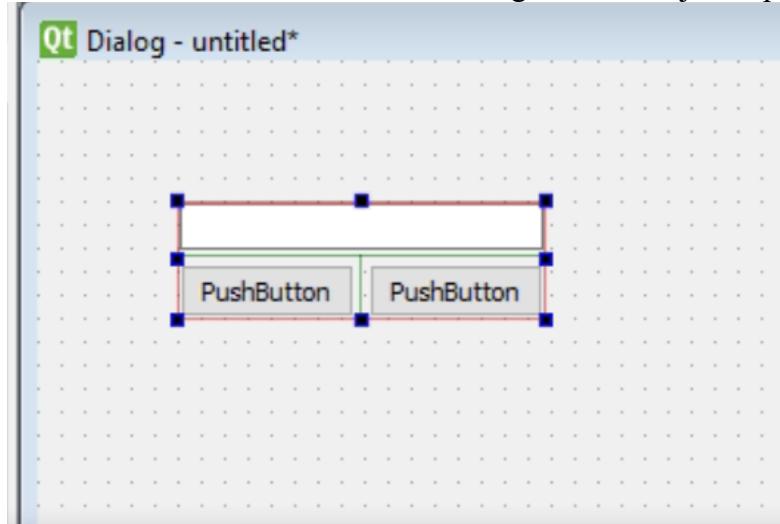
1. Buat rancangan layout berikut menggunakan Qt Designer, Pilih Form dengan tipe **Dialog Without Button**. simpan dengan nama signalslot.ui. Perhatikan bahwa peletakan QLineEdit dan QPushButton masih berantakan.



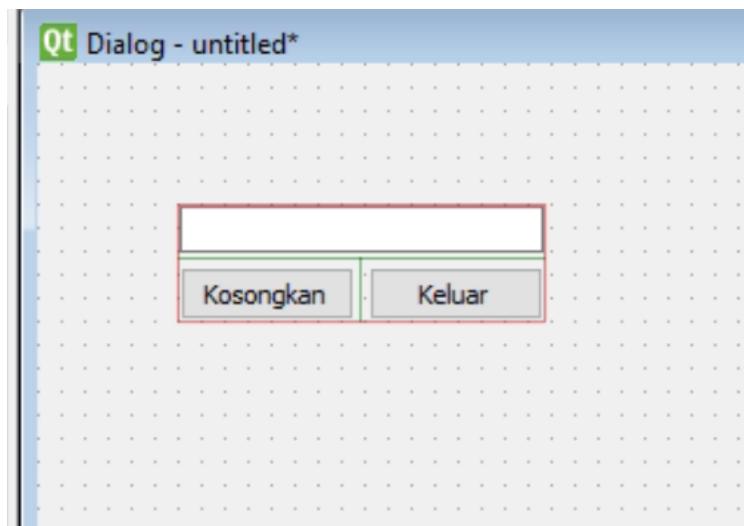
2. Seleksi semua komponen dengan cara tahan Ctrl + Klik QPushButton dan QLineEdit, kemudian pilih logo Layout in Grid



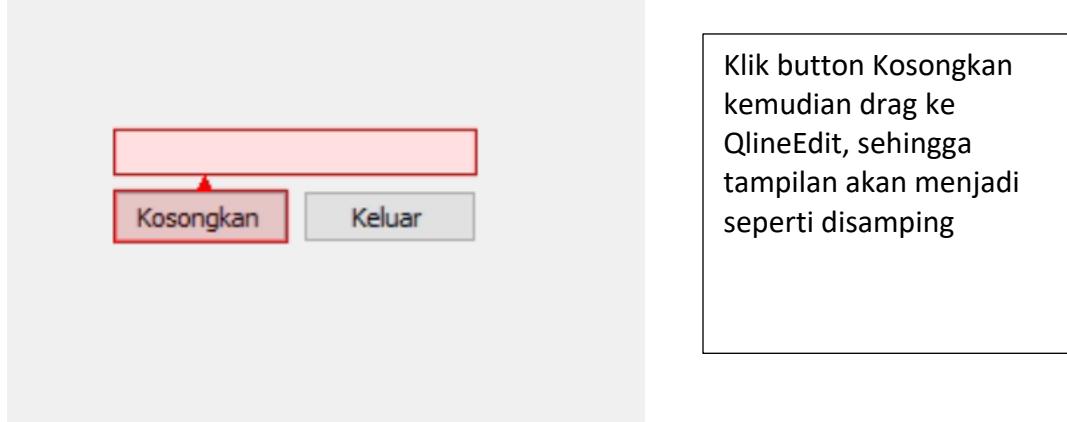
3. Secara otomatis maka susunan widget akan menjadi rapi menyesuaikan layout

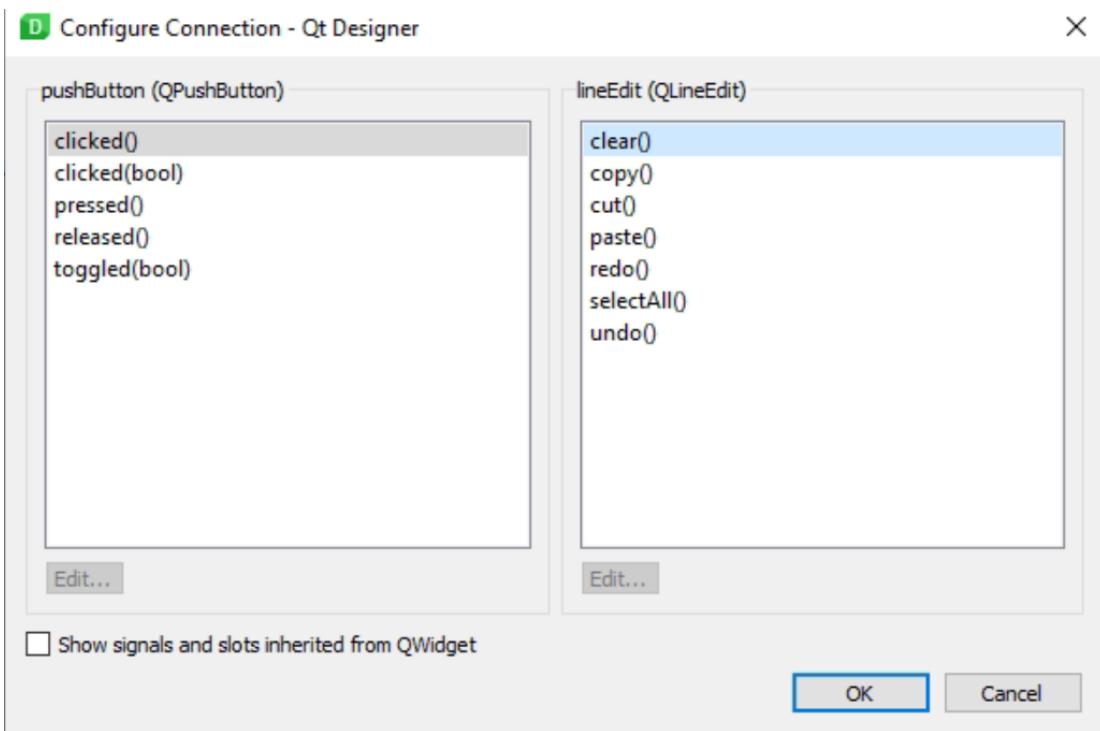


4. Ubah nama PushButton 1 dan 2 menjadi **Kosongkan** dan **Keluar**

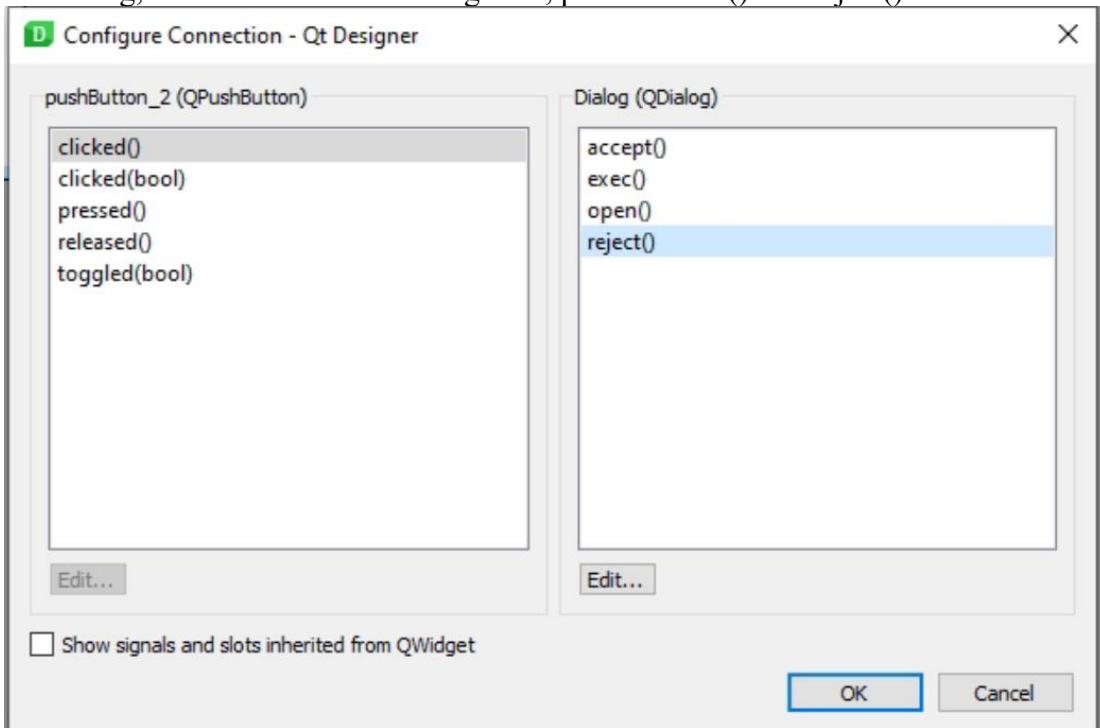


5. Masuk ke Mode Edit Signal/Slot dengan klik logo  pada toolbar. Kemudian klik button **Kosongkan** dan drag ke QLineEdit, maka akan muncul dialog baru, pilih Clicked() dan Clear()

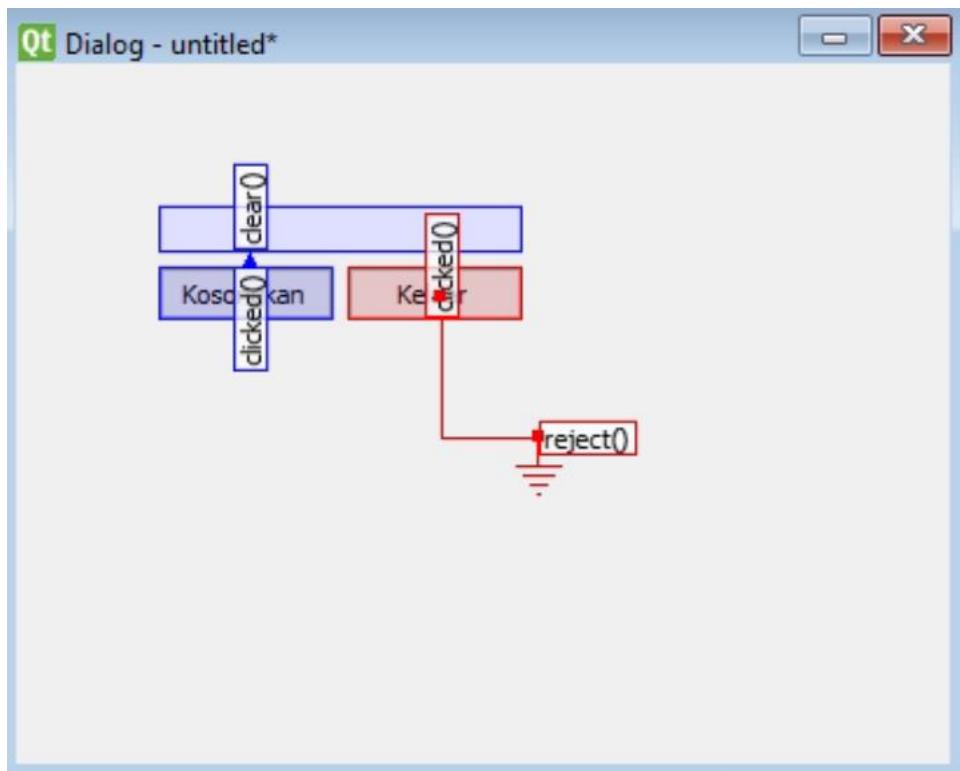




6. Lakukan hal yang sama untuk klik button **Keluar** dan namun drag ke form kosong, maka akan muncul dialog baru, pilih Clicked() dan reject()



7. Tampilan Form akan menjadi seperti berikut



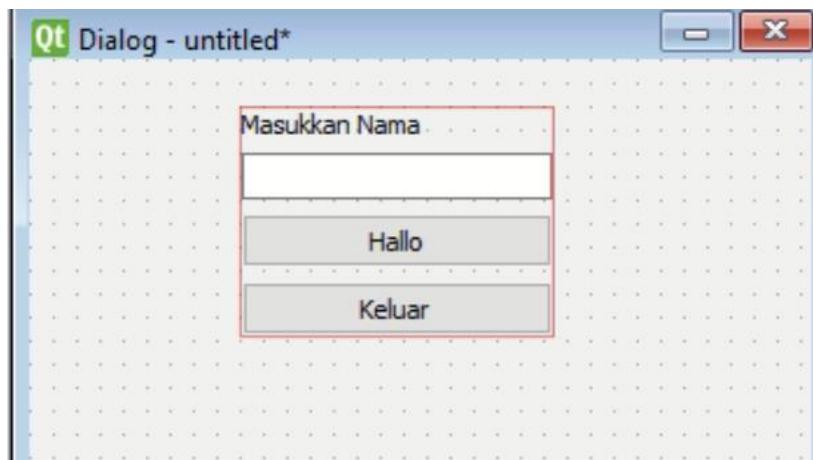
5. Buka command prompt pada folder yang sama dengan file signalslot.ui kemudian jalankan perintah berikut

```
|pyuic6 -x signalslot.ui -o signalslot.py
```

6. Jalankan aplikasi dengan mengetikkan perintah “ python signalslot.py ” pada command prompt

11.3.3 Kegiatan 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. Pada Property Editor, Beri **object name** tombol halo dengan tmbHalo

Property	Value
QObject	
objectName	tmbHalo
QWidget	
enabled	<input checked="" type="checkbox"/>

3. Pada Property Editor, Beri **object name** tombol Keluar dengan tmbkeluar

Property	Value
QObject	
objectName	tmbKeluar
QWidget	
enabled	<input checked="" type="checkbox"/>
geometry	[(1, 56), 133 x 23]

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

```
pyuic6 -x form.ui -o form.py
```

5. Buka file form.py menggunakan visual studio kemudian tambahkan kode berikut (warna biru)

```
1. # Form implementation generated from reading ui file 'form.ui'
2. #
3. # Created by: PyQt6 UI code generator 6.4.2
4. #
5. # WARNING: Any manual changes made to this file will be lost when
6. # pyuic6 is
7. # run again. Do not edit this file unless you know what you are
8. # doing.
9. from PyQt6 import QtCore, QtGui, QtWidgets
10.
11.
12. class Ui_Dialog(object):
13.     def setupUi(self, Dialog):
14.         Dialog.setObjectName("Dialog")
15.         Dialog.resize(181, 155)
16.         self.widget = QtWidgets.QWidget(parent=Dialog)
17.         self.widget.setGeometry(QtCore.QRect(21, 41, 135, 80))
18.         self.widget.setObjectName("widget")
19.         self.gridLayout = QtWidgets.QGridLayout(self.widget)
20.         self.gridLayout.setContentsMargins(0, 0, 0, 0)
21.         self.gridLayout.setObjectName("gridLayout")
22.         self.lineEdit = QtWidgets.QLineEdit(parent=self.widget)
23.         self.lineEdit.setObjectName("lineEdit")
24.         self.gridLayout.addWidget(self.lineEdit, 0, 0, 1, 1)
25.         self.tmbHalo = QtWidgets.QPushButton(parent=self.widget)
26.         self.tmbHalo.setObjectName("tmbHalo")
27.         self.gridLayout.addWidget(self.tmbHalo, 1, 0, 1, 1)
28.         self.tmbKeluar = QtWidgets.QPushButton(parent=self.widget
    )
29.         self.tmbKeluar.setObjectName("tmbKeluar")
30.         self.gridLayout.addWidget(self.tmbKeluar, 2, 0, 1, 1)
31.
32.         self.retranslateUi(Dialog)
33.         self.tmbKeluar.clicked.connect(Dialog.reject) # type: ign
    ore
34.         self.tmbHalo.clicked.connect(self.tampilanPesan)
35.         QtCore.QMetaObject.connectSlotsByName(Dialog)
36.
37.         self.dialog = Dialog
38.
```

```

39.     def retranslateUi(self, Dialog):
40.         _translate = QtCore.QCoreApplication.translate
41.         Dialog.setWindowTitle(_translate("Dialog", "Dialog"))
42.         self.tmbHalo.setText(_translate("Dialog", "Halo"))
43.         self.tmbKeluar.setText(_translate("Dialog", "Keluar"))
44.
45.     def tampilanPesan(self):
46.         isi = self.lineEdit.text()
47.         QtWidgets.QMessageBox.information(self.dialog, "Pesan And
48. a", f"Halo, {isi}!")
49. if __name__ == "__main__":
50.     import sys
51.     app = QtWidgets.QApplication(sys.argv)
52.     Dialog = QtWidgets.QDialog()
53.     ui = Ui_Dialog()
54.     ui.setupUi(Dialog)
55.     Dialog.show()
56.     sys.exit(app.exec())

```

6. Jalankan aplikasi dengan mengetikkan perintah “ python form.py ” pada command prompt

11.4 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

