

Praktikum Pemrograman GUI Pertemuan 5

Dasar Materi (Penggunaan PyQt)

1. Penggunaan Kelas QLabel
2. Penggunaan Gambar icon ke dalam kontrol QPushButton
3. Penggunaan Kelas QTextEdit
4. Penggunaan Kelas QRadioButton
5. Penggunaan Kelas QCheckBox
6. Penggunaan Kelas QComboBox

Upload Ke Github masing-masing, materi yang telah diajarkan oleh Dosen Pengampu Praktikum.



<https://github.com/afandi354>

1. Menyisipkan Gambar dengan dengan kelas QLabel

Langkah-langkah :

- Buat sebuah file dengan nama **DemoGambar.py**
- Ketikkan source code berikut pada file **DemoGambar.py**

```
import sys

from PyQt5.QtGui import *
from PyQt5.QtCore import *
from PyQt5.QtWidgets import *

class DemoGambar(QWidget):
    def __init__(self):
        super().__init__()
        self.setupUi()

    def setupUi(self):
        self.resize(500, 300)
        self.move(300, 300)
        self.setWindowTitle('Menyisipkan Gambar')

        self.label1 = QLabel()
        self.label1.setText('<b><font size=11> Demo Menampilkan  
Gambar dengan QLabel</font></b>')

        self.label2 = QLabel()
        self.label2.setText('')

        layout = QVBoxLayout()
        layout.addWidget(self.label1)
        layout.addWidget(self.label2)
        self.setLayout(layout)

if __name__ == '__main__':
    a = QApplication(sys.argv)

    form = DemoGambar()
    form.show()

    a.exec_()
```

- Running di terminal atau CMD dengan perintah : **python DemoGambar.py**
- Screenshot hasil keluaranya.
- Push source code ke github!

Latihan 1

1. Buatlah program menggunakan kelas QLabel untuk menyisipkan gambar.
Dengan ketentuan ukuran form 500x600



2. Menyisipkan icon pada QPushButton

Langkah-langkah :

- a. Buat sebuah file dengan nama **DemoIcon.py**
- b. Ketikkan source code berikut pada file **DemoIcon.py**

```
import sys

from PyQt5.QtGui import *
from PyQt5.QtCore import *
from PyQt5.QtWidgets import *

class DemoIcon(QWidget):
    def __init__(self):
        super().__init__()
        self.setupUi()

    def setupUi(self):
        self.resize(400, 100)
        self.move(300, 300)
        self.setWindowTitle('Demo QPushButton')

        self.label = QLabel()
        self.label.setText('Demo QPushButton dengan gambar icon')

        icon1 = QIcon('add-icon.png')
        self.button1 = QPushButton('\tTambah')
        self.button1.setIcon(icon1)

        icon2 = QIcon('delete-icon.png')
        self.button2 = QPushButton('\tHapus')
        self.button2.setIcon(icon2)

        icon3 = QIcon('refresh-icon.png')
        self.button3 = QPushButton('\tRefresh')
        self.button3.setIcon(icon3)

        hbox = QHBoxLayout()
        hbox.addWidget(self.button1)
        hbox.addWidget(self.button2)
        hbox.addWidget(self.button3)

        layout = QVBoxLayout()
        layout.addWidget(self.label)
        layout.addLayout(hbox)
        self.setLayout(layout)

if __name__ == '__main__':
    a = QApplication(sys.argv)

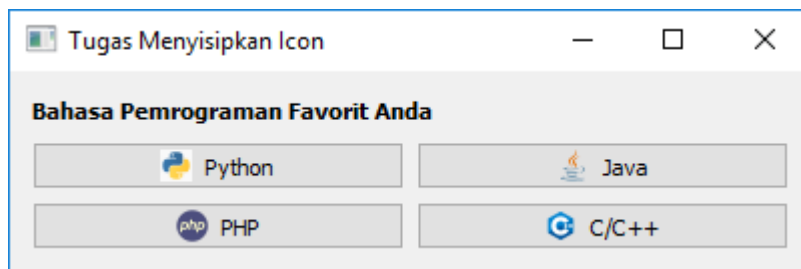
    form = DemoIcon()
    form.show()

    a.exec_()
```

- c. Running di terminal atau CMD dengan perintah : **python DemoIcon.py**
- d. Screenshot hasil keluaranya
- e. Push source code ke github!

Latihan 2

2. Buatlah program yang menampilkan icon pada Push Botton seperti pada gambar dibawah ini, dengan ketentuan ukuran 400x100



3. Membuat program dengan QTextEdit

Langkah-langkah :

- a. Buat sebuah file dengan nama **DemoQTextEdit.py**
- b. Ketikan source code berikut pada file **DemoQTextEdit.py**

```
import sys

from PyQt5.QtGui import *
from PyQt5.QtCore import *
from PyQt5.QtWidgets import *

class MainForm(QWidget):
    def __init__(self):
        super().__init__()
        self.setupUi()

    def setupUi(self):
        self.resize(400, 200)
        self.move(300, 300)
        self.setWindowTitle('Demo QTextEdit')

        self.label1 = QLabel()
        self.label1.setText('No. HP')
        self.phoneEdit = QLineEdit()
        vbox1 = QVBoxLayout()
```

```

vbox1.addWidget(self.label1)
vbox1.addWidget(self.phoneEdit)

self.label2 = QLabel()
self.label2.setText('Pesan')
self.messageEdit = QTextEdit()
vbox2 = QVBoxLayout()
vbox2.addWidget(self.label2)
vbox2.addWidget(self.messageEdit)

vbox3 = QVBoxLayout()
vbox3.addLayout(vbox1)
vbox3.addLayout(vbox2)

self.sendButton = QPushButton('&Kirim SMS')
self.cancelButton = QPushButton('&Batal')
hbox = QHBoxLayout()
hbox.addStretch()
hbox.addWidget(self.sendButton)
hbox.addWidget(self.cancelButton)

layout = QVBoxLayout()
layout.addLayout(vbox3)
horizontalLine = QFrame()
horizontalLine.setFrameShape(QFrame.HLine)
horizontalLine.setFrameShadow(QFrame.Sunken)
layout.addWidget(horizontalLine)
layout.addLayout(hbox)
self.setLayout(layout)

if __name__ == '__main__':
    a = QApplication(sys.argv)

    form = MainForm()
    form.show()

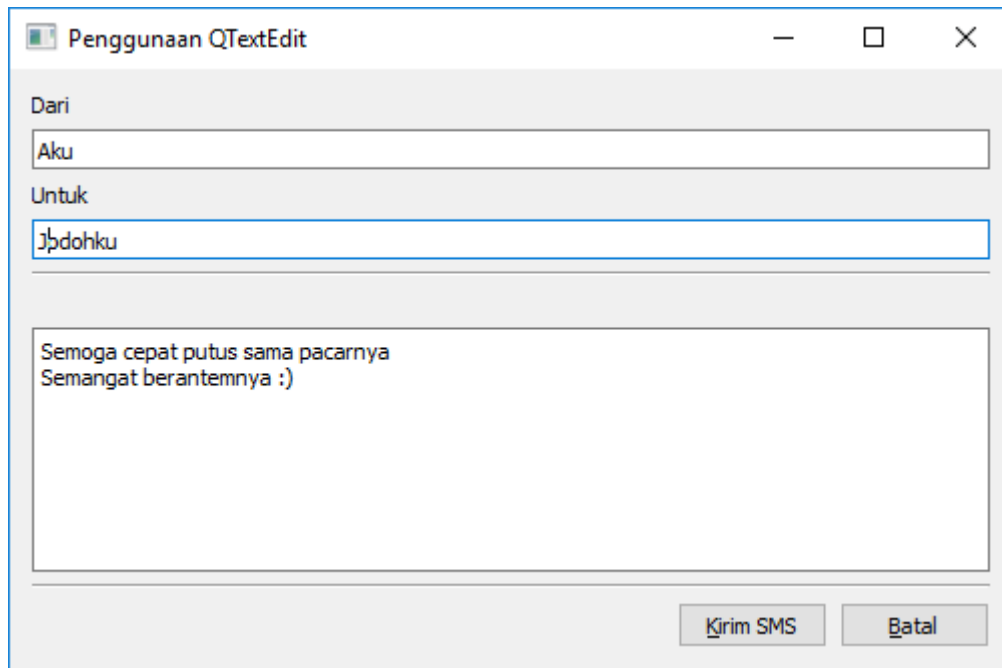
    a.exec_()

```

- c. Running di terminal atau CMD dengan perintah : **python DemoQTextEdit.py**
- d. Screenshot hasil keluaranya
- e. Push source code ke github!

Latihan 3

3. Buatlah program yang menampilkan TextEdit, sehingga tampilan programnya terlihat seperti gambar di bawah ini.



4. Membuat Radio Button dengan PyQt

Langkah-langkah :

- Buat sebuah file dengan nama **DemoQRadioButton.py**
- Ketikkan source code berikut pada file **DemoQRadioButton.py**

```
import sys
```

```
from PyQt5.QtGui import *  
from PyQt5.QtCore import *  
from PyQt5.QtWidgets import *
```

```
class MainForm(QWidget):
```

```
    def __init__(self):  
        super().__init__()  
        self.setupUi()
```

```
    def setupUi(self):  
        self.resize(400, 150)  
        self.move(300, 300)  
        self.setWindowTitle('Demo QRadioButton')
```

```
        self.label1 = QLabel()  
        self.label1.setText('Siapa Dosen Pengampu Mata Kuliah <b>Pemrograman  
GUI</b>')
```

```
        self.cekCondro = QRadioButton()  
        self.cekCondro.setText('&Condro Kartiko')  
        self.cekCondro.setChecked(True)
```

```

self.cekVanda = QRadioButton()
self.cekVanda.setText('&Novanda Alim Setya')
self.cekFandi = QRadioButton()
self.cekFandi.setText('&Afandi Nur Aziz')
self.cekGita = QRadioButton()
self.cekGita.setText('&Gita Fadhila')

vbox = QVBoxLayout()
vbox.addWidget(self.label1)
vbox.addWidget(self.cekCondro)
vbox.addWidget(self.cekVanda)
vbox.addWidget(self.cekFandi)
vbox.addWidget(self.cekGita)

self.resultLabel = QLabel('<b>Jawaban Anda: </b>')
self.checkButton = QPushButton('Cek Jawaban')

layout = QVBoxLayout()
layout.addLayout(vbox)
layout.addWidget(self.checkButton)
layout.addWidget(self.resultLabel)
layout.addStretch()
self.setLayout(layout)

self.checkButton.clicked.connect(self.checkButtonClick)

def checkButtonClick(self):
    if self.cekFandi.isChecked():
        self.resultLabel.setText('<b>Jawaban Anda : Benar</b>')
    else:
        self.resultLabel.setText('<b>Jawaban Anda : Salah</b>')

if __name__ == '__main__':
    a = QApplication(sys.argv)

    form = MainForm()
    form.show()

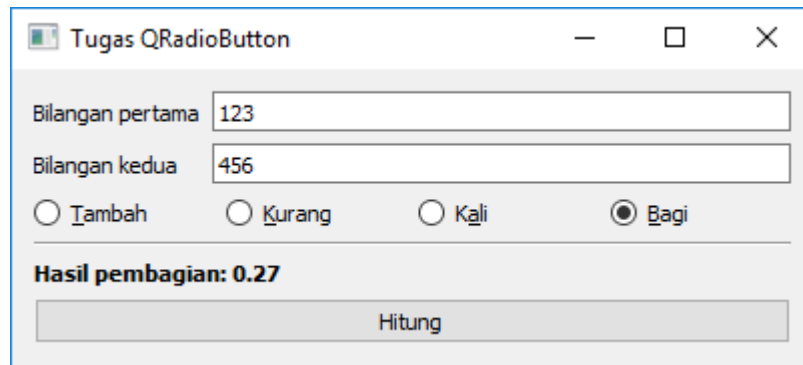
    a.exec_()

```

- c. Running di terminal atau CMD dengan perintah : **python DemoQRadioButton.py**
- d. Screenshot hasil keluaranya
- e. Push *source code* ke github!

Latihan 4

4. Buatlah program untuk membuat kalkulator sederhana menggunakan RadioButton seperti tampilan gambar dibawah ini



5. Membuat CheckBox dengan PyQt

Langkah-langkah :

- a. Buat sebuah file dengan nama **DemoQCheckBox.py**
- b. Ketikkan source code berikut pada file **DemoQCheckBox.py**

```
import sys
```

```
from PyQt5.QtGui import *  
from PyQt5.QtCore import *  
from PyQt5.QtWidgets import *
```

```
class MainForm(QWidget):
```

```
    def __init__(self):  
        super().__init__()  
        self.setupUi()
```

```
    def setupUi(self):  
        self.resize(300, 100)  
        self.move(300, 300)  
        self.setWindowTitle('Demo QCheckBox')
```

```
        self.label = QLabel()  
        self.label.setText('Bahasa Pemrograman Favorit Anda :')
```

```
        self.javaCheck = QCheckBox()  
        self.javaCheck.setText('Java')  
        self.pythonCheck = QCheckBox()  
        self.pythonCheck.setText('Python')  
        self.rubyCheck = QCheckBox()  
        self.rubyCheck.setText('Ruby')  
        self.phpCheck = QCheckBox()  
        self.phpCheck.setText('PHP')
```

```

hbox1 = QHBoxLayout()
hbox1.addWidget(self.javaCheck)
hbox1.addWidget(self.pythonCheck)
hbox1.addWidget(self.rubyCheck)
hbox1.addWidget(self.phpCheck)

self.okButton = QPushButton('&OK')
self.exitButton = QPushButton('Keluar')
hbox2 = QHBoxLayout()
hbox2.addStretch()
hbox2.addWidget(self.okButton)
hbox2.addWidget(self.exitButton)

layout = QVBoxLayout()
layout.addWidget(self.label)
layout.addLayout(hbox1)
horizontalLine = QFrame();
horizontalLine.setFrameShape(QFrame.HLine)
horizontalLine.setFrameShadow(QFrame.Sunken)
layout.addWidget(horizontalLine)
layout.addLayout(hbox2)
layout.addStretch()
self.setLayout(layout)

self.okButton.clicked.connect(self.okButtonClick)
self.exitButton.clicked.connect(self.close)

def okButtonClick(self):
    choices = []
    if self.javaCheck.isChecked(): choices.append('Java')
    if self.pythonCheck.isChecked(): choices.append('Python')
    if self.rubyCheck.isChecked(): choices.append('Ruby')
    if self.phpCheck.isChecked(): choices.append('PHP')
    QMessageBox.information(self, 'Informasi', repr(choices))

if __name__ == '__main__':
    a = QApplication(sys.argv)

    form = MainForm()
    form.show()

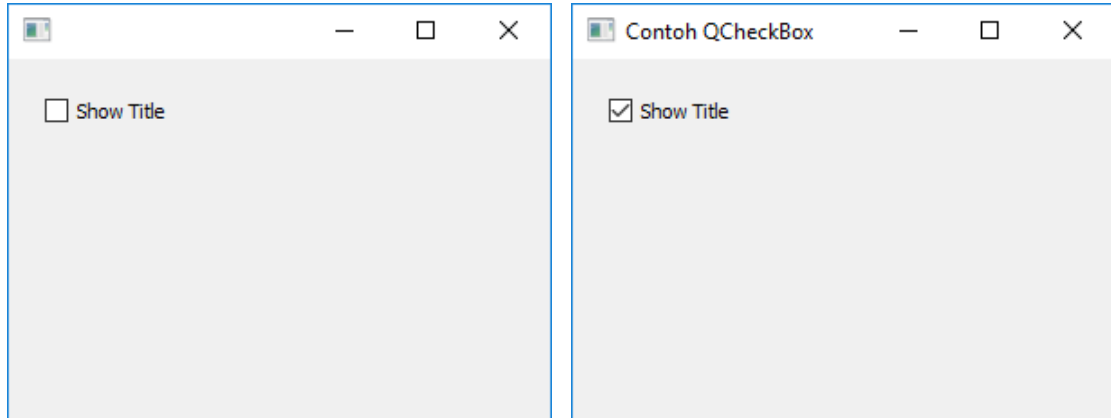
    a.exec_()

```

- c. Running di terminal atau CMD dengan perintah : **python DemoQCheckBox.py**
- d. Screenshot hasil keluaranya
- e. Push *source code* ke github!

Latihan 5

5. Buatlah program menggunakan CheckBox, dimana ketika CheckBox di check (✓) maka judul form akan muncul. Ketika di uncheck judul form akan hilang



6. Membuat ComboBox dengan PyQt

Langkah-langkah :

- Buat sebuah file dengan nama **DemoQComboBox.py**
- Ketikkan source code berikut pada file **DemoQComboBox.py**

import sys

```
from PyQt5.QtGui import *  
from PyQt5.QtCore import *  
from PyQt5.QtWidgets import *
```

```
class MainForm(QWidget):
```

```
    def __init__(self):  
        super().__init__()  
        self.setupUi()
```

```
    def setupUi(self):  
        self.resize(300, 100)  
        self.move(300, 300)  
        self.setWindowTitle('Demo QComboBox')
```

```
        self.combo = QComboBox()  
        for i in range(1,11):  
            self.combo.addItem('Item ke-%d' % i)  
  
        self.getTextButton = QPushButton('Ambil Teks')
```

```
        layout = QVBoxLayout()  
        layout.addWidget(self.combo)  
        layout.addWidget(self.getTextButton)
```

```

layout.addStretch()
self.setLayout(layout)

self.getTextButton.clicked.connect(self.getTextButtonClick)

def getTextButtonClick(self):
    QMessageBox.information(self, 'Informasi',
        'Anda memilih: ' + self.combo.currentText())

if __name__ == '__main__':
    a = QApplication(sys.argv)

    form = MainForm()
    form.show()

    a.exec_()

```

- c. Running di terminal atau CMD dengan perintah : **python DemoQComboBox.py**
- d. Screenshot hasil keluaranya
- e. Push *source code* ke github!

Latihan 6

6. Buatlah program menggunakan ComboBox, untuk memilih menu makanan dan minuman. Makanan yang disediakan adalah : Mendoan, Cireng, Gethuk, Tahu Bulat, Ketan Susu. Sedangkan minumannya adalah : Es Cincu, Milkshake, Chatime, Thaitea, Kopi Hitam

